Journal of Information Assurance and Security.
ISSN 1554-1010 Volume 16 (2021) pp. 010-023
© MIR Labs, www.mirlabs.net/jias/index.html

Artificial Neural Network for Cybersecurity: A Comprehensive Review

Prajoy Podder [1], Subrato Bharati [2], M. Rubaiyat Hossain Mondal [3], Pinto Kumar Paul [4], Utku Kose [5]

[1,2,3] Institute of Information and Communication Technology, Bangladesh University of Engineering and Technology, Dhaka-1205, Bangladesh
[4]Ranada Prasad Shaha University, Narayanganj, Bangladesh
[5]Suleyman Demirel University, Isparta, Turkey

[1] *prajoypodder@gmail.com,* [2] *subratobharati1@gmail.com,* [3] *rubaiyat97@iict.buet.ac.bd,* [4] *pinto.kumar07@gmail.com,*
[5]*utkukose@sdu.edu.tr*

Abstract: **Cybersecurity is a very emerging field that protects systems, networks, and data from digital attacks. With the increase in the scale of the Internet and the evolution of cyber attacks, developing novel cybersecurity tools has become important, particularly for Internet of things (IoT) networks. This paper provides a systematic review of the application of deep learning (DL) approaches for cybersecurity. This paper provides a short description of DL methods which is used in cybersecurity, including deep belief networks, generative adversarial networks, recurrent neural networks, and others. Next, we illustrate the differences between shallow learning and DL. Moreover, a discussion is provided on the currently prevailing cyber-attacks in IoT and other networks, and the effectiveness of DL methods to manage these attacks. Besides, this paper describes studies that highlight the DL technique, cybersecurity applications, and the source of datasets. Next, a discussion is provided on the feasibility of DL systems for malware detection and classification, intrusion detection, and other frequent cyber-attacks, including identifying file type, spam, and network traffic. Our review indicates that high classification accuracy of 99.72% is obtained by restricted Boltzmann machine (RBM) when applied to a custom dataset, while long short-term memory (LSTM) achieves an accuracy of 99.80% for KDD Cup 99 dataset. Finally, this article discusses the importance of cybersecurity for reliable and practicable IoT-driven healthcare systems.**

Keywords: Deep learning; cyber analytics; autoencoders; convolutional neural networks (CNN), deep belief networks (DBN)

I. Introduction

Cybersecurity is the complete package of all techniques and technologies responsible for defending networks, software, and data from attacks [1, 2]. The mechanism of cyber defense is available at the network, data level, host and application. Some cybersecurity tools like firewalls, the system of intrusion detection, the system of intrusion protection etc., are always active at each end to identify security breaches and stop attacks [3, 4]. Nevertheless, with the increasing number of systems having Internet-connection, the risk of attacks is increasing day by day. With the realization of Internet of things (IoT) networks, cybersecurity is becoming more important than ever. Computer networks including IoT are vulnerable to many security threats. Some attacks are of known pattern can be easily managed. However, attackers are developing zero-day exploits, where the attack takes places as soon as a weakness in the system is detected. Such an attack has no previous record and the attack can damange the computer system before the problem is solved. Moreover, the system must be defended not only from external threats but also need to be protected from insider threats, such as misuse of the authorized access, which can be an individual or mean to be a part of the organization.

The main challenge is finding out the compromising system's indicators from the attack's lifecycle, which may have meaningful signs of a future attack. However, this could be a difficult job because of massive quantities of data-generating continuously from lots of cyber-enabled devices. Data Science uses the extensive range of data made by the cyber defense system, including the security information and event management (SIEM) scheme, sometimes overflowing the specialist in security with the event warnings, identifying patterns, related events, and detecting abnormal behaviour to improve cybersecurity.

Hybrid detection in security amalgamates anomaly and misuse detection. This system is mainly used to decrease the rate of false-positive value of anonymous attacks and enhance the rate of detection of recognized intrusions. Maximum DL approaches are hybrid methods [5, 6].Previous reviews, i.e., those in [7-9] have illustrated applications of machine learning (ML) for the solution of cyber-related problems, but deep learning (DL) methods have not been focused on those papers. Some works illustrate DL approaches for cybersecurity. These approaches have some limitations in the applications on cybersecurity [10, 11].

This paper reviews cybersecurity using DL. Moreover, DL methods in cybersecurity and the difference between DL and shallow learning are broadly discussed, and the results of different DL methods are reported. The rest of the paper is organized as follows. Section II discusses the differences between DL and ML, Section III introduces different DL methods in the context of cybersecurity. DL and shallow learning are compared in Section IV. The performance results of different DL methods are reported in Section V. Finally, the paper concludes in Section VI.

II. DL and ML

Both ML and DL are subsets of artificial intelligence (AI). The differences between ML and DL include the following:

a) Dependencies of data: The performance of DL models are not comparatively better than traditional ML models for small-scale data volumes. The reason behind this is DL models need a large portion of data to comprehend the data flawlessly. On the other hand, traditional ML algorithms use the established rules [14].

b) Hardware dependencies: Graphics Processing Unit (GPU) can be considered essential hardware for training the DL models properly. The GPU is mainly applied to optimize matrix processes effectively since DL models require a lot of matrix operations. On the other hand, traditional ML algorithms do not usually require high-performance machines with GPUs [18].

c) Processing in feature: The procedure of driving domain knowledge into a feature extractor in order to decrease the complexity of data is termed feature processing. Patterns are usually generated in feature processing, and therefore, ML and DL algorithms work better. However, this stage is time-consuming, and specialized knowledge is required in this case. The performance of most ML models rely on the features accuracy (i.e., pixel values, textures, shapes, locations, etc.) extracted. Attempting to derive high-level features openly from personal data is a main difference between traditional ML and DL algorithms [17]. Accordingly, DL decreases the designing effort to an extracting features for every problem.

d) Execution time: Large execution time is needed to train a DL model owing to its having various parameters. The training step also takes longer. On the contrary, less execution time (only seconds to few hours) is needed to train a ML model. Nevertheless, the time required in testing stage is just the contrast. DL models need very short testing time compared with some ML models.

III. DL Approaches in Cyber Security

This section illustrates different types of DL methods used in cyber security.

A. *Deep Belief Networks*

Deep Belief Networks (DBNs) is brought in a seminal paper by Geoffrey Hinton. DBNs are a class of Deep Neural Networks (DNNs). A DBN is composed of several layers of hidden casual variables. Besides, there are connections exists between the layers and no connections between units within each layer [12]. It is the combination of probability and statistics with ML and neural networks. Figure 1 shows different types of DBN.

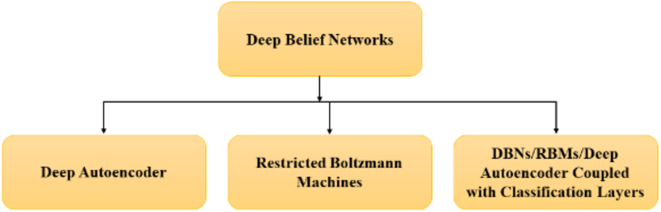

Deep Belief Networks
Deep Autoencoder
Restricted Boltzmann Machines
DBNs/RBMs/Deep Autoencoder Coupled with Classification Layers

Figure 1. Classification of DBN

B. *Autoencoder*

An unsupervised method is an autoencoder where the input is given as a vector. The network attempts to match and the output is the same as the input vector. One can generate a lower or higher dimensionality illustration of the data by getting the input and varying the recreating the input with its dimensionality. Data encoding operation (i.e., feature compression) is executed in the network with a small dimension of hidden layers. A denoising autoencoder can play an important role in order to eliminate the noise and reconstruct the original input from the noisy input. Figure 2 illustrates a basic autoencoder.

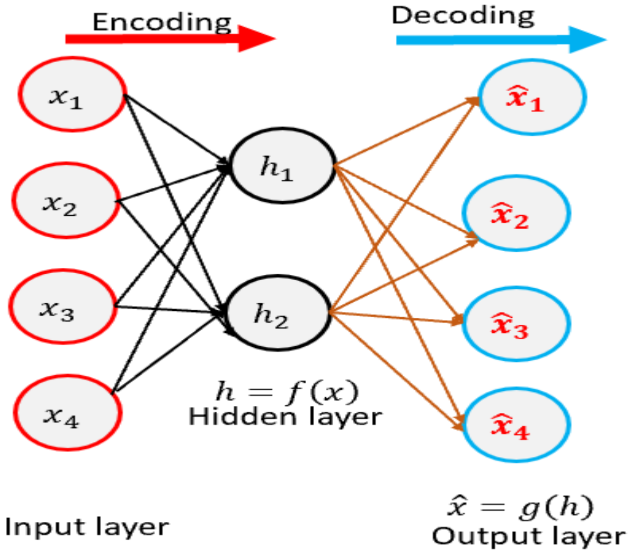


Figure 2. Autoencoder

C. *Recurrent Neural Network*

A recurrent neural network (RNN), a subset of neural networks, which is connected between nodes and form a directed graph as shown in Figure 3. This makes the network in its internal state. It permits to show dynamic sequential behavior. They use their internal memory to process arbitrary sequences of input and the signal travels both forward and backward by creating loops in the network [13-15].

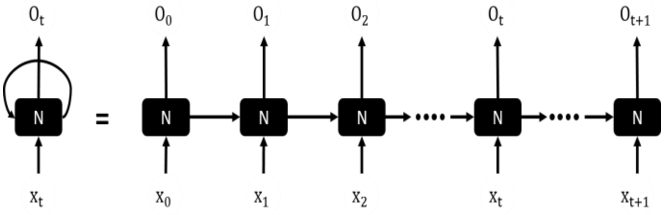


Figure 3. Recurrent Neural Network

Typically, it is more complex to train RNNs due to the disappearance of the gradients. However, the improvements in architecture and training have formed various RNNs. This model is simpler to train. The long short-term memory (LSTM), an improved system of RNN, was first brought by Hochreiter and Schmidhuber in 1997 [16]. LSTM is making a major change in speech recognition and set a revolutionary record on some traditional models in certain speech applications. It is introduced to solve RNNs short term memory problem. LSTM units connect to the situation in the following time stage. The configuration of the units that accumulate information is called a memory cell.

D. Convolutional Neural Network

Convolutional neural network (CNN) is a portion of deep NN that processes as well as analyze visual imagery input. If a colored or grayscale image is considered as input, then the image will be stored in pixels like 2D array. In addition, CNNs are also applied for managing audio spectrograms with 2D arrays. However, the model of CNN contains three kinds of layers, including classification layers, pooling layers and convolution layers [15, 17]. An illustration of CNN is shown in Figure 4.

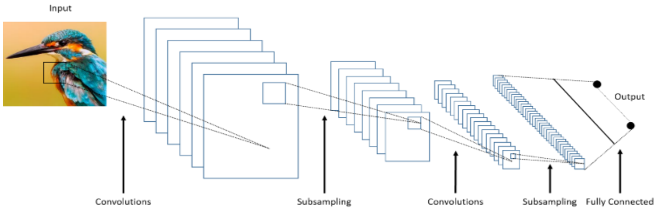


Figure 4. Convolutional Neural Network

E. Generative Adversarial Networks

GANs are deployed in unsupervised ML, where 2 neural networks contest against one another in a game of zero-sum to overcome one another. It is introduced by the work of Goodfellow. Figure 5 shows the block diagram of GAN. The generator produces output data using the similar features as real time data by using input data. Then, the discriminator analyze the real data, whether the input is real or fake [18]. There is a wide range of applications in GAN system, including optical flow estimation [98], caption generation [97], image enhancement [96], and DCGAN for Facebook [99].

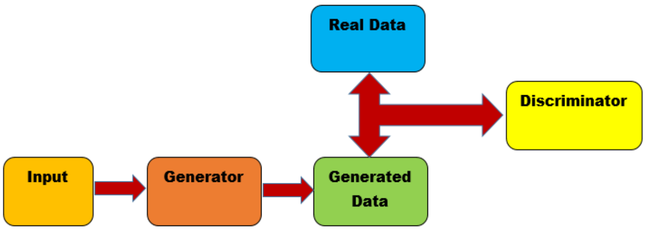


Figure 5. Generative Adversarial Networks

F. Recursive Neural Network

Recursive neural networks relate a number of weights recursively. It has a number of inputs. At first, the primary 2 inputs are nurtured in the model as one. A node output is then considered as an input for the following node. Many natural language processing and image segmentation use this type of model.

IV. Comparison between Shallow Learning and DL

This section provides a brief comparison between DL and shallow learning algorithms. DL has multiple layers, as shown in Figure 6. Besides, in DL, a deep network has several hidden layers, while shallow neural networks typically have 1-hidden layer. The neuron layers are linked with adaptive weights, besides the neighbor network layers are generally staying associated. However, there are two kinds of shallow network architecture: supervised and unsupervised. In supervised learning, the labels remain known to learn a work. Moreover, feature extraction is achieved individually.

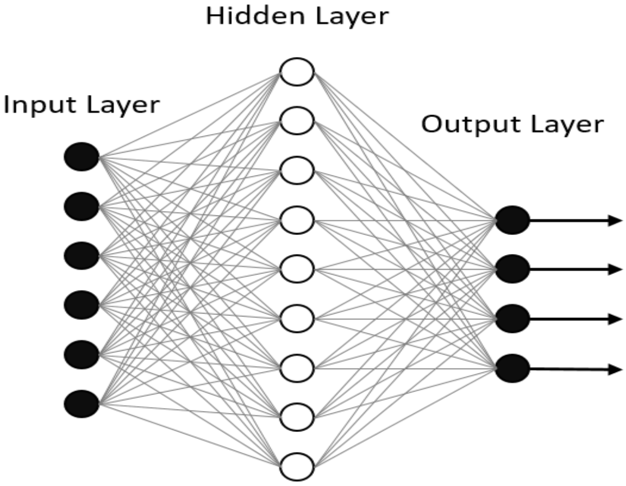


Figure 6. Shallow Neural Network

This forms of DL model derives higher-level features from the raw input with the help of its multiple hidden layers. Figure 7 illustrates a deep neural network. There are several levels between the input layer and output layer; the output layer is considered as higher level, and input layer are considered as lower level. From the lower-level concepts, higher-level concepts are defined. Although feature extraction can be obtained from the few initial layers of DL network. The DL architecture is of three types: unsupervised, hybrid and supervised. Advance feature extraction in shallow neural networks is performed separately because they have only one hidden layer. However, deep networks are capable of learning. However, with great computational power, several GPUs are needed for DL methods and it costs too much time to train DL models [19].

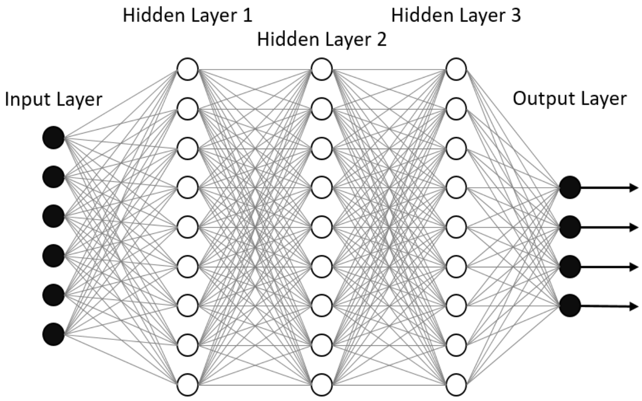


Figure 7. Deep Neural Network

Cyber Attack	**DL Method**	**Dataset**	**Research Paper**
Malware Detection and Classification	CNN	Microsoft Malware Classification Challenge [108]	[41]
	Autoencoder	Comodo Cloud Security Center [119]	[33]
	Autoencoder	Dataset of call sequence in Public malware API [130]	[42]
	CNN	Genome Project [120], McAfee Labs	[32]
	CNN RNN	Maltrieve Private, Virus Share [121]	[24]
	CNN RNN	Unknown	[25]
	CNN (Dynamic)	Unknown	[39]
	DNN	Private data of Jotti commercial	[29]
	DNN	Internal Microsoft dataset	[40]
	DNN	DREBIN [107]	[45]
	DNN	Microsoft corporation provided dataset	[44]
	Autoencoders (Denoising)	C4 Security dataset	[43]
	RNN	Internal Microsoft dataset	[23]
	RNN	Virus Total [117], Alexa [60]	[36]
	RBM	Contagio [118] Google Play Store [116]	[21]
	RBM	Contagio [118], Google Play Store [116], Genome Project [120]	[22]
	RBM	Comodo Cloud [119], Security Center	[26]
	RBM	Virus share [121],Google play store [116]	[35]
	RBM	Self-generated dataset	[34]
	RBM	VirusTotal [117], DREBIN [107], Google Play [116], Genome Project [120]	[27]
	RBM	Comodo Cloud [119], Security Center	[28]
	RBM	Unknown	[38]
	RBM	Unknown	[31]
	Autoencoder	Challenge of Classification of Microsoft Malware[108],NSL-KDD [115]	[46]
	DNN	Malware of traffic data, VirusShare, Kaspersky, MalShare, Malware sample	[37]

Table 1. DL methods for Malware Detection and Classification

Cyber Attack	**DL Method**	**Dataset**	**Research Paper**
Intrusion Detection	Autoencoder	KDDCUP 1999 [114]	[56]
	Autoencoder	KDDCUP 1999 [114]	[62]
	Autoencoder	KDDCUP 1999 [114]	[63]
	Autoencoder	NSL-KDD [115]	[61]
	Autoencoder	Network Experiments and Open-Car-Test-bed	[60]
	Autoencoder	NSL-KDD [115]	[70]
	Autoencoder	CIDDS-001 [127]	[54]
	Ladder Networks (Autoencoder)	KDD 1999 [114]	[71]
	Autoencoder RBM	KDD 1999 [114]	[53]
	Autoencoder	Custom	[55]
	CNN	CTU-13 [124], IXIA [131]	[56]
	CNN (dilated) Autoencoder	CTU-UNB [124, 125], Contagio-CTU-UNB [125]	[20]
	DNN	KDDCUP 1999 [114]	[58]
	DNN	simulator of Cooja network	[59]

	DNN	NSL-KDD [115]	[69]
	RBM	KDDCUP 1999 [114]	[49]
	RNN	KDDCUP 1999 [114]	[67]
	RNN	UNM, ADFA-LD, KDDCUP-1998 [114] data sets	[65]
	RNN	KDDCUP-1999 and additional [114], unique data	[66]
	RNN	Custom	[46]
	RNN	KDD 1999 [114]	[68]
	RNN	KDD 1999 [114]	[64]
	RNN	NSL-KDD [115]	[57]
	RBM	NSL-KDD [115]	[47]
	RBM	KDD 1999 [114]	[50]
	RBM	NSL-KDD [115]	[51]
	RBM	NSL-KDD [115] KDDCUP 1999 [114] UNSW-NB15	[48]
	RBM Autoencoder	KDD 1999[114]	[52]
	RBM RNN	CTU-13[124]	[158]
	Autoencoder	Challenge of Microsoft Classification of Malware [108], NSL-KDD [115]	[46]

Table 2. DL methods for Intrusion Detection

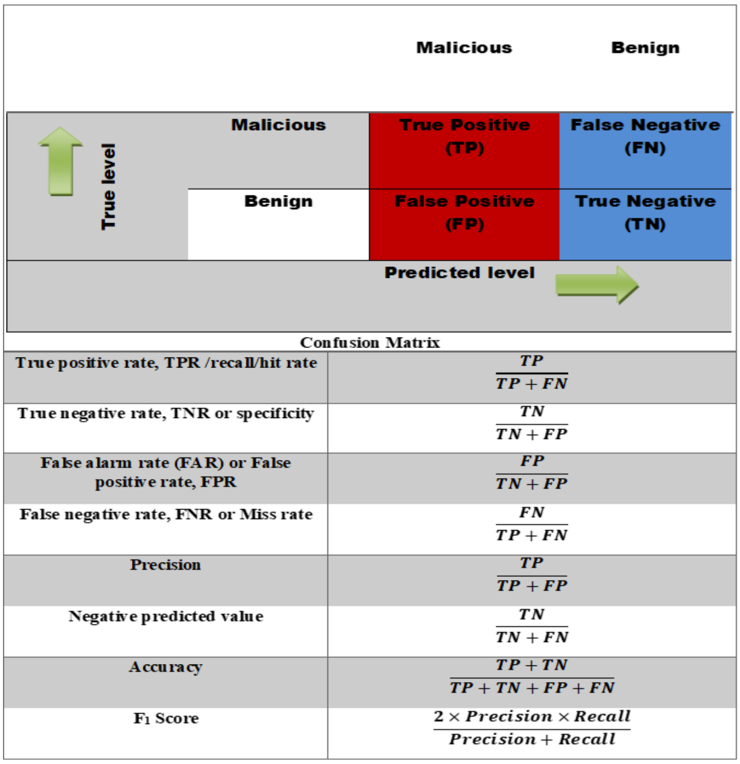
Cyber Attack	**DL Method**	**Dataset**	**Research Paper**
File Type Identification	Autoencoder	Internal Dataset	[81]
Identification of Network Traffic	Autoencoder	Honeypot dataset resulted internally	[83]
	Autoencoder	ISCX-VPN-nonVPN-traffic dataset[164]	[82]
Spam Identification	Autoencoder	EnronSpam [126], PU1, PU2, PU3, PU4	[84]
	RBM	EnronSpam [126] SpamAssassin [127] LingSpam [128]	[85]
Impersonation Attacks	Autoencoder	AWID[156]	[86]
User Authentication	Autoencoder	Custom	[87]
DGA	CNN	Alexa [104], Private Dataset	[88]
	GAN	Alexa [104]	[89]
	RNN	Alexa [104], OSINT [105]	[90]
	RNN	Alexa [104], DGArchive [106], OSINT [105]	[91]
	CNN RNN	Alexa [104], OSINT [105]	[92]
	CNN RNN	Alexa [104], DGArchive [106]	[93]
	RNN	Alexa [104], OSINT [105]	[94]
	RNN	Malware Capture Facility Project Dataset	[95]
Attack of Drive-by Download	CNN	KDD 1999 [114]	[119]
	CNN	honeypot setup, Malware Bytes,Malware domain list, Alexa [104]	[118]
Traffic Identification	CNN	ISCX VPN-nonVPN traffic dataset [164]	[163]
Insider Threat	DNN RNN	CERT Dataset v6.2 [123]	[166]
Anomaly Detection	RNN	Custom	[169]
Keystroke Verification	RNN	Custom	[170]
False Data Ejection	RBM	Custom	[172]

Table 3. DL methods for Other Frequent Cyber Attack

Dataset Name	**Features**	**Reference**
KDD Cup 99	Intrusion Detection, R2L, DoS, Probing	[114]
NSL-KDD	Network Intrusion Detection	[115]
CTU-13	Scenarios of thirteen captures of various samples in botnet are included in this dataset. Every scenario is taken in a file like pcap. That file consists of every packets of 3 kinds of traffic.	[59]
Alexa	Alexa offers us access to a set of web sites. Alexa can be expected legitimate.	[104]
AWID	Comprehensive WiFi network benchmark dataset. Detect impersonation attacks	[100]
DGA	38 classes with 168,900 samples	[106]
CTU	Consists of different, real botnet attacks and labels.	[103]
OSINT	OSINT DGA feeds from Bambenek Consulting. DGA (Domain Generation Algorithm) are used by malware	[105]
VirusShare	A repository of 38,005,488 malware samples.	[121]
DREBIN	1,20,000 android applications are contained in this dataset.	[107]

Microsoft Malware Classification Challenge	Every malware has a unique value of 20-character hash recognizing the file and a unique Id, and a Class. An integer value is represented names of nine-family. This malware can belong.	[108]
CERT Insider Threat Dataset v6.2	System logs spanning 516 days are included. This dataset contains over 130 million events. Approximately 400 of them are malicious.	[109, 110]
EnronSpam	The total number of emails (legitimate and spam) is 5975. The ratio of spam to legitimate rate is 1:3.	[111]
SpamAssassin	It can differentiate effectively between non-spam and spam in between cases of 95% to 100%, relying on the kind of mail taken and Bayesian filter with training.	[112]
Malware Training Sets	Zeus:2014 samples, Crypto:2024 samples, Locker:434 samples, APT1:292 samples.	[123]
CIDDS-001	The dataset consists a large number of traffic instances. 153026 instances are collected from External Server. 172839 instances are collected from OpenStack Server.	[127]
Public malware API call sequence dataset	API call sequences (Malware): 42,797, API call sequences (goodware): 1,079. Each API call sequence: First 100 API calls that are consecutively non-repeated	[130]
ISCX VPN-nonVPN traffic dataset	It has a categories of 14-traffic: P2P, VOIP, VPN-P2P,VPN-VOIP, etc. Wireshark and tcpdump were used for capturing the traffic generated about 28GB of data.	[131]

Table 4. Explanation of Cyber Security Datasets in DL



		Malicious	Benign
True level	Malicious	True Positive (TP)	False Negative (FN)
	Benign	False Positive (FP)	True Negative (TN)
		Predicted level	

Confusion Matrix

True positive rate, TPR /recall/hit rate	$\frac{TP}{TP+FN}$
True negative rate, TNR or specificity	$\frac{TN}{TN+FP}$
False alarm rate (FAR) or False positive rate, FPR	$\frac{FP}{TN+FP}$
False negative rate, FNR or Miss rate	$\frac{FN}{TP+FN}$
Precision	$\frac{TP}{TP+FP}$
Negative predicted value	$\frac{TN}{TN+FN}$
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
F1 Score	$\frac{2\times Precision \times Recall}{Precision+Recall}$

Figure 8. Several performance metrics

However, DL takes too much time to analyze and extract relevant information from the huge amount of data and the data is not formed properly.

Table 1 summarizes various DL methods applied by researchers for malware detection and classification. Most researchers use restricted Boltzmann machine (RBM) method. Table 2 summarizes various DL methods applied for intrusion detection. Most researchers use autoencoder and RNN method. Table 3 summarizes the DL method used in order to detect other type of cyber-attacks.

KDD Cup 99 dataset formed for the challenge of KDD in 1999 is one of the most commonly used datasets in order to detect the various type of intrusions. KDD means Knowledge Discovery. About more than four million network traffic records exist in this dataset. Twenty two different types of attacks are contained in this dataset that can be categorized into four families such as denial-of-service (DoS), R2L, for example, predicting the password, U2R, and probing. The other datasets used in various research papers for the classification of various threats have been described in Table 4 with short details. Several performance metrics are depicted in Figure 8.

V. PERFORMANCE ANALYSIS OF DL METHODS APPLIED IN VARIOUS RESEARCH PAPERS

DL models have shown significant improvements over traditional ML-based solutions, signature-based methods and rule-based methods in order to address cybersecurity problems. Table 5 illustrates the performance results achieved adopting different DL models. The results are reported in terms of precision, false negative rate (FNR), classification accuracy, F1-score, true positive rate (TPR), etc. We have reviewed 85 papers. From the review, it can be seen that most researchers have focused on malware classification and detection of various types of intrusion in the network. Cyber-physical autonomous systems which is not only sensor-based but also communication-enabled (e.g., automotive systems), biometrics behavioral (i.e., signature dynamics) are considered as increasing areas for DL applications of security.

As we become more reliant on network-connected devices, we will see an increase in the number of cyber-physical systems and computational systems, each having its own set of attack vectors owing to its unique baseline. For malware and intrusion detection, RBMs were the most often utilized DL technique. RNNs were another popular solution for tackling the largest range of cyber security challenges feasible (i.e., network intrusions, cyber-physical intrusions, malware, host intrusions and names of malicious domain).

The large use of RBMs and autoencoders, around 50%, is most likely owing to a scarcity of labeled data, and unlabeled data is pre-trained and fine-tuned using a little quantity of labeled data. RNNs are likely popular because many cyber security jobs or data may be treated as a time series problem. This is beneficial to RNNs.

Conclusions on the success of any approach are difficult to make since various studies utilize various datasets and measurements. Certain tendencies, however, are remarkable. The performance of various areas of the security business varied greatly. Domains constructed employing a variety of techniques seem to have the most consistent DGA-produced hazardous domains, with TPRs ranging from 1% to 1.5 % and accuracy values ranging from 0.9959 to 0.9969, equivalent to 96.01 to 99.86%. Network intrusion detection techniques, on the other hand, have a performance range of 92.33 to 100 percent with a TPR of 1.58 to 2.3 percent and an accuracy range of 44 to 99 percent. A high classification accuracy of 99.72% is reported for RBM when applied to a custom dataset [34], while accuracy of 99.80% is achieved by LSTM for KDD Cup 99 dataset [66]. Historically, the capacity to detect network intrusions has significantly been reliant on the kind and quantity of attacks carried out. Another crucial component influencing overall performance was the training set's relationship between benign and dangerous data. This quandary stems from the difficulties of getting legally harmful materials. Because authentic data might be difficult to get, data is often generated using viral simulations and reverse engineering.

Methods	Data used	Paper	Precision	FNR	Accuracy	F1-Score	TPR
DBN	KDD CUP 99	[47]	92.33%		93.49%		
LR-DBN	10% KDDCUP'99	[49]	97.9%	2.47%			
DBN	Netflow	[31]			96.7%		
DL RBM	NSL-KDD, KDD CUP 1999, UNSW-NB15	[48]	81.95%, 94.43%, 83.40%		90.99%, 97.11%, 95.84%		77.48%, 92.77%, 79.19%
RBM	[118], [120], [116]	[22]	95.77%		96.76%		97.84%
DBN	40% NSL-KDD	[50]			97.5%		
RNN	NSL-KDD	[57]			83.28%		
Autoencoder	KDD CUP 99	[71]			99.18%		
DBN based PNN	KDD CUP 99	[80]	93.25%	0.62%	99.14%		
CNN and RNN (LSTM and Softmax layer)	Virus Share [121], Maltrieve Private	[24]	85.6%		89.4%		89.4%
DBN		[31]			96.7%		
CNN	Genome Project, McAfee Labs (malware samples: 2475 and benign samples: 3627) (benign: 9268 and malware: 9902)	[32]	99.0%, 27%		98.0%, 80%	97.0%, 78%	95.0%, 85%

Autoencoder	Comodo Cloud Security Center	[33]			95.64%		
RBM	-	[26]			96.6%		
RBM	Custom	[34]			99.72%		90.1%
RBM	EnronSpam	[85]			93.4%		
RSTNN	Custom	[36]	97.6%			96.9%	96.2%
DNN	-	[37]	97.1%				100%
Recurrent SVM	Alexa, OSINT	[92]	92.06%		99.69%	92.60%	93.14%
Bidirectional LSTM	Alexa, OSINT	[92]	92.32%		99.64%	92.70%	93.09%
Fusion of CNN and LSTM	Alexa, OSINT	[92]	91.59%		99.59%	92.06%	92.53%
CNN	Alexa, DGArchive	[93]			99.18%		72.89%
LSTM	Alexa, DGArchive	[93]			98.96%		74.05%
DNN		[99]			97%		
CNN	Malimg Dataset, Microsoft Malware Dataset	[41]			98.52%, 98.99%, 99.57%		
DNN	DREBIN	[45]		6.37%, 3.96%	95.93%, 98.35%		
RNN	UNM, ADFA-LD, KDDCUP 1998 [114] data sets	[65]	99.31%	4.62%	96%	97.31%	95.38%
RNN	KDD Cup 99	[67]			96.93%	-	98.88%
RNN	KDD Cup 99	[68]	84.6%		77.55%	-	73%
LSTM	10% KDD Cup 99	[64]	-		93.85%	-	-
LSTM	10% KDD Cup 99	[72]	98.8%		96.93%	-	-
LSTM	KDD Cup 99	[66]	-		99.8%	-	-
LSTM	KDD Cup 99	[73]	98.95%		97.54%	-	-
GRU	Netflow	[74]	-		84.15%	-	-
CNN	CTU-UNB datasets	[20]	98.44%		-	-	98%
CNN	Netflow	[75]	93%		-	-	92%
CNN	Netflow	[76]	-		92%		-
1D-CNN	ISCX dataset	[77]	97.3%		-	-	96%
CNN	Netflow	[78]	-		99.41%		-
Stacked Autoencoder	AWID	[101]	86.15%		92.18%		
DBN, RNN	CTU dataset	[102]	81.26%, 68.63%				99.34%, 70.35%
CNN	Microsoft Malware Classification Challenge (2015)	[129]			99.24%		100%

Table 5. Performance Analysis of various DL methods

When employing DL-driven security technology, some difficulties may arise. The model's accuracy can be viewed as a significant impediment.

The use of any new tool, especially DL tools, is universally frowned upon because they are ultimately black boxes. As a result, when errors occur, determining the cause is impossible, and unlike DL applications such as the marketing sector, larger costs and hazards are associated with cybersecurity missteps. A cybersecurity analyst may waste time analyzing false alarms, or an automated response to intrusion detection may erroneously restrict access to critical services. Furthermore, a DL tool can completely ignore a cyber-attack. Another barrier to adoption is that many of the currently available systems focus on a specific hazard, such as virus detection. Researchers should investigate methods for generalizing or combining multiple DL approaches in order to cover a broader range of attack vectors and provide a more comprehensive solution. Multiple DL detection techniques must be used concurrently, and information gathered by various techniques may also be used to improve local performance.

Cybersecurity has become an important issue for IoT since IoT can contribute to managing pandemics, particularly the novel coronavirus disease (COVID-19). One example of the use of IoT for COVID-19 is to mitigate the causative virus from being spreading. This can be done by the screening of temperature, tracing the contacts, and several other ways. Detecting early cases of the infection, tracing, and then isolating the suspected patients can be done with IoT. Note that IoT-driven healthcare systems and IoT-driven COVID-19 diagnosis systems are emerging techniques that can be useful to patients and doctors. Another example is facilitating the new lifestyle during COVID-19, including home-office, distant learning, fitness training at home, etc. These activities enable the running of businesses, educational institutions, government offices without risking the people's health. Another use case of IoT is to resolve machinery issues for controlling medical inventory, tracking tagged nebulizers, oxygen cylinders, and other medical equipment. For tackling a pandemic, IoT can be used along with other techniques such as near field communication, radio frequency identification, WiFi, light fidelity, sensor networks, etc. These technologies require small portable devices that have low computation power and low battery life. As a result, ensuring cybersecurity for small IoT devices is a more challenging task compared to traditional computers, server, smartphones and laptops. Cyber attacks evolve rapidly, so it is difficult to incorporate security measures in IoT devices quickly. Unless the cyber attacks are mitigated, IoT cannot be effectively used in controlling pandemics. Security threats such as phishing, spamming, ransomware, Distributed DoS [137-143] may affect the reliability of IoT-driven healthcare and COVID-19 diagnosis [132-136] systems. Hence, understanding the possible security threats and finding appropriate mitigation techniques is essential in the context of IoT and other networking scenarios.

VI. Conclusions

This paper focuses on the use of DL in improving the security system. As attacks of malicious against cyber system networks are advancing, the cyber defender needs to be more advanced. Cybersecurity personnel should have the capability to remark and employ original signatures to identify original attacks. DL approaches to cybersecurity applications offer a smart opportunity to identify novel malware variants and attacks of zero-day. In this review, we have described the applications of DL systems to different types of cybersecurity attack types. These attacks are mainly application software, targeted networks, data and host system. Likewise, this paper illustrates that the standard datasets are very important to advancing DL in the cybersecurity domain. The paper aims to draw a complete review of DL methods, the needs of DL in cybersecurity, and to encourage future research of DL in cybersecurity. Finally, this article discusses the use case scenarios of IoT in the context of COVID-19, and highlights the importance of cybersecurity for IoT devices.

References

[1] Buczak, L.; Guven, E. A Survey of Data Mining and Machine Learning Methods for Cyber Security. IEEE Commun. Surv. Tutor. 2016, 18, 1153–1176.

[2] A. Milenkoski, M. Vieira, S. Kounev, A. Avritzer, and B. D. Payne. Evaluating computer intrusion detection systems: A survey of common practices. ACM Comput. Surv., vol. 48, no. 1, pp. 1-41, 2015.

[3] C. N. Modi and K. Acha. Virtualization layer security challenges and intrusion detection/prevention systems in cloud computing: A comprehensive review. J. Supercomput., vol. 73, no. 3, pp. 1192-1234, 2017.

[4] Nguyen, T.T.T.; Armitage, G. A survey of techniques for internet traffic classification using machine learning. IEEE Commun. Surv. Tutor. 2008, 10, 56–76.

[5] E. Viegas, A. O. Santin, A. França, R. Jasinski, V. A. Pedroni, and L. S. Oliveira. Towards an energy-efficient anomaly-based intrusion detection engine for embedded systems," IEEE Trans. Comput., vol. 66, no. 1, pp. 163-177, Jan. 2017.

[6] A. Patcha and J.-M. Park, ``An overview of anomaly detection techniques: Existing solutions and latest technological trends," Comput. Netw., vol. 51, no. 12, pp. 34483470, Aug. 2007.

[7] Sperotto, A.; Schaffrath, G.; Sadre, R.; Morariu, C.; Pras, A.; Stiller, B. An overview of IP flow-based intrusion detection. IEEE Commun. Surv. Tutor. 2010, 12, 343–356.

[8] Wu, S.X.; Banzhaf, W. The use of computational intelligence in intrusion detection systems: A review. Appl. Soft Comput. 2010, 10, 1–35. [CrossRef]

[9] Torres, J.M.; Comesaña, C.I.; García-Nieto, P.J. Machine learning techniques applied to cybersecurity. Int. J. Mach. Learn. Cybern. 2019, 1–14.

[10] Xin, Y.; Kong, L.; Liu, Z.; Chen, Y.; Li, Y.; Zhu, H.; Gao, M.; Hou, H.;Wang, C. Machine Learning and Deep Learning Methods for Cybersecurity. IEEE Access 2018, 6, 35365–35381.

[11] Bharati, S.; Podder, P.; Mondal, M.; Robel, M. and Alam, R.; Threats and countermeasures of cyber security in direct and remote vehicle communication systems. Journal of Information Assurance & Security. 2020, 15(4), 153-164. 2020.

[12] Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.; Du, X.; Guizani, M. A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security. arXiv 2018, arXiv:1807.11023.

[13] El Hihi, S.; Bengio, Y. Hierarchical recurrent neural networks for long-term dependencies. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 1996; pp. 493–499.

[14] Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2014; pp. 3104–3112.

[15] Berman, D.S.; Buczak, A.L.; Chavis, J.S.; Corbett, C.L. A Survey of Deep Learning Methods for Cyber Security. Information 2019, 10, 122.

[16] Hochreiter, S.; Schmidhuber, J. Long short-term memory. Neural Comput. 1997, 9, 1735–1780.

[17] Sainath, T.N.; Mohamed, A.R.; Kingsbury, B.; Ramabhadran, B. Deep convolutional neural networks for LVCSR. In Proceedings of the 2013 IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 8614–8618.

[18] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.

[19] Deng, L.; Yu, D. Deep learning: Methods and applications. Found. Trends Signal Process. 2014, 7, 197–387.

[20] Yu, Y.; Long, J.; Cai, Z. Network intrusion detection through stacking dilated convolutional autoencoders. Secur. Commun. Netw. 2017, 2017, 4184196.

[21] Yuan, Z.; Lu, Y.; Wang, Z.; Xue, Y. Droid-sec: Deep learning in android malware detection. ACM SIGCOMM Comput. Commun. Rev. 2014, 44, 371–372.

[22] Yuan, Z.; Lu, Y.; Xue, Y. Droiddetector: Android malware characterization and detection using deep learning. Tsinghua Sci. Technol. 2016, 21, 114–123.

[23] Pascanu, R.; Stokes, J.W.; Sanossian, H.; Marinescu, M.; Thomas, A. Malware classification with recurrent networks. In Proceedings of the 2015 IEEE International Conference Acoustics, Speech and Signal Process, (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 1916–1920.

[24] Kolosnjaji, B.; Zarras, A.; Webster, G.; Eckert, C. Deep learning for classification of malware system call sequences. In Proceedings of the Australasian Joint Conf. on Artificial Intelligence, Hobart, Australia, 5–8 December 2016; pp. 137–149.

[25] Tobiyama, S.; Yamaguchi, Y.; Shimada, H.; Ikuse, T.; Yagi, T. Malware detection with deep neural network using process behavior. In Proceedings of the IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Atlanta, GA, USA, 10–14 June 2016; Volume 2, pp. 577–582.

[26] Hou, S.; Saas, A.; Ye, Y.; Chen, L. Droiddelver: An android malware detection system using deep belief network based on API call blocks. In Proceedings of the International Conference Web-Age Information Manage, Nanchang, China, 3–5 June 2016; pp. 54–66.

[27] Zhu, D.; Jin, H.; Yang, Y.;Wu, D.; Chen,W. DeepFlow: Deep learning-based malware detection by mining Android application for abnormal usage of sensitive data. In Proceedings of the 2017 IEEE Symposium Computers and Communications (ISCC), Heraklion, Greece, 3–6 July 2017; pp. 438–443.

[28] Ye, Y.; Chen, L.; Hou, S.; Hardy,W.; Li, X. DeepAM: A heterogeneous deep learning framework for intelligent malware detection. Knowl. Inf. Syst. 2018, 54, 265–285.

[29] Saxe, J.; Berlin, K. Deep neural network based malware detection using two dimensional binary program features. In Proceedings of the 10th International Conference Malicious and Unwanted Software (MALWARE),Washington, DC, USA, 20–22 October 2015; pp. 11–20.

[30] Weber, M.; Schmid, M.; Schatz, M.; Geyer, D. A toolkit for detecting and analyzing malicious software. In Proceedings of the 18th Annual Computer Security Applications Conference, Las Vegas, NV, USA, 9–13 December 2002; pp. 423–431.

[31] Ding, Y.; Chen, S.; Xu, J. Application of Deep Belief Networks for opcode based malware detection. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 3901–3908.

[32] McLaughlin, N.; del Rincon, J.M.; Kang, B.; Yerima, S.; Miller, P.; Sezer, S.; Safaei, Y.; Trickel, E.; Zhao, Z.; Doupe, A.; et al. Deep android malware detection. In Proceedings of the 7th ACM on Conference on Data and Application Security and Privacy, Scottsdale, AZ, USA, 22–24 March 2017; pp. 301–308.

[33] Hardy, W.; Chen, L.; Hou, S.; Ye, Y.; Li, X. DL4MD: A deep learning framework for intelligent malware detection. In Proceedings of the International Conference Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; p. 61.

[34] Benchea, R.; Gavrilu¸t, D.T. Combining restricted Boltzmann machine and one side perceptron for malware detection. In Proceedings of the International Conference on Conceptual Structures, Iasi, Romania, 27–30 July 2014; pp. 93–103.

[35] Xu, L.; Zhang, D.; Jayasena, N.; Cavazos, J. HADM: Hybrid analysis for detection of malware. In Proceedings of the SAI Intelligent Systems Conference, London, UK, 21–22 September 2016; pp. 702–724.

[36] Shibahara, T.; Yagi, T.; Akiyama, M.; Chiba, D.; Yada, T. Efficient dynamic malware analysis based on network behavior using deep learning. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM),Washington, DC, USA, 4–8 December 2016; pp. 1–7.

[37] Mizuno, S.; Hatada, M.; Mori, T.; Goto, S. Bot Detector: A robust and scalable approach toward detecting malware-infected devices. In Proceedings of the 2017 IEEE International Conference Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–7.

[38] Chen, Y.; Zhang, Y.; Maharjan, S. Deep learning for secure mobile edge computing. arXiv 2017, arXiv:1709.08025.

[39] Hill, G.D.; Bellekens, X.J.A. Deep learning based cryptographic primitive classification. arXiv 2017, arXiv:1709.08385.

[40] Dahl, G.E.; Stokes, J.W.; Deng, L.; Yu, D. Large-scale malware classification using random projections and neural networks. In Proceedings of the 2013 IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 3422–3426.

[41] M. Kalash, M. Rochan, N. Mohammed, N. D. B. Bruce, Y. Wang and F. Iqbal, "Malware Classification with Deep Convolutional Neural Networks," 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2018, pp. 1-5, doi: 10.1109/NTMS.2018.8328749.

[42] Wang, X.; Yiu, S.M. A multi-task learning model for malware classification with useful file access pattern from API call sequence. arXiv 2016, arXiv:1610.05945.

[43] David, O.E.; Netanyahu, N.S. Deepsign: Deep learning for automatic malware signature generation and classification. In Proceedings of the 2015 International Joint Conference Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015; pp. 1–8.

[44] Huang, W.; Stokes, J.W. MtNet: A multi-task neural network for dynamic malware classification. In Proceedings of the International Conference Detection of Intrusions and Malware, and Vulnerability Assessment, Donostia-San Sebastián, Spain, 7–8 July 2016; pp. 399–418.

[45] Grosse, K.; Papernot, N.; Manoharan, P.; Backes, M.; McDaniel, P. Adversarial perturbations against deep neural networks for malware classification. arXiv 2016, arXiv:1606.04435.

[46] McDermott, C.D.; Majdani, F.; Petrovski, A. Botnet detection in the Internet of things using deep learning approaches. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.

[47] Gao, N.; Gao, L.; Gao, Q.; Wang, H. An intrusion detection model based on deep belief networks. In Proceedings of the 2014 2nd International Conference Advanced Cloud and Big Data (CBD), Huangshan, China, 20–22 November 2014; pp. 247–252.

[48] Nguyen, K.K.; Hoang, D.T.; Niyato, D.; Wang, P.; Nguyen, P.; Dutkiewicz, E. Cyberattack detection in mobile cloud computing: A deep learning approach. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6.

[49] Alrawashdeh, K.; Purdy, C. Toward an online anomaly intrusion detection system based on deep learning. In Proceedings of the 15th IEEE International Conference Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 195–200.

[50] Alom, M.Z.; Bontupalli, V.; Taha, T.M. Intrusion detection using deep belief networks. In Proceedings of the 2015 National

Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, 15–19 June 2015; pp. 339–344.

[51] Dong, B.; Wang, X. Comparison deep learning method to traditional methods using for network intrusion detection. In Proceedings of the 8th IEEE International Conference Communication Software and Networks (ICCSN), Beijing, China, 4–6 June 2016; pp. 581–585.

[52] Li, Y.; Ma, R.; Jiao, R. A hybrid malicious code detection method based on deep learning. Methods 2015, 9, 205–216.

[53] Alom, M.Z.; Taha, T.M. Network intrusion detection for cyber security using unsupervised deep learning approaches. In Proceedings of the 2017 IEEE National Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, 27–30 June 2017; pp. 63–69.

[54] Abdulhammed, R.; Faezipour, M.; Abuzneid, A.; AbuMallouh, A. Deep and machine learning approaches, for anomaly-based intrusion detection of imbalanced network traffic. IEEE Sens. Lett. 2018.

[55] Mirsky, Y.; Doitshman, T.; Elovici, Y.; Shabtai, A. Kitsune: An ensemble of autoencoders for online network intrusion detection. arXiv 2018, arXiv:1802.09089.

[56] Wang, W.; Zhu, M.; Zeng, X.; Ye, X.; Sheng, Y. Malware traffic classification using convolutional neural network for representation learning. In Proceedings of the IEEE 2017 International Conference on Information Networking (ICOIN), Da Nang, Vietnam, 11–13 January 2017; pp. 712–717.

[57] Yin, C.L.; Zhu, Y.F.; Fei, J.L.; He, X.Z. A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access 2017, 5, 21954–21961.

[58] Roy, S.S.; Mallik, A.; Gulati, R.; Obaidat, M.S.; Krishna, P.V. A Deep Learning Based Artificial Neural Network Approach for Intrusion Detection. In Proceedings of the International Conference Mathematics and Computing, Haldia, India, 17–21 January 2017; pp. 44–53.

[59] Tang, T.A.; Mhamdi, L.; McLernon, D.; Zaidi, S.A.R.; Ghogho, M. Deep learning approach for network intrusion detection in software defined networking. In Proceedings of the 2016 International Conference Wireless Networks and Mobile Communication (WINCOM), Fez, Morocco, 26–29 October 2016; pp. 258–263.

[60] Chawla, S. Deep Learning Based Intrusion Detection System for Internet of Things; University of Washington: Seattle, WA, USA, 2017.

[61] Diro, A.A.; Chilamkurti, N. Deep learning: The frontier for distributed attack detection in Fog-to-Things computing. IEEE Commun. Mag. 2018, 56, 169–175.

[62] Ma, T.; Wang, F.; Cheng, J.; Yu, Y.; Chen, X. A Hybrid Spectral Clustering and Deep Neural Network Ensemble Algorithm for Intrusion Detection in Sensor Networks. Sensors 2016, 16, 1701.

[63] Aminanto, M.E.; Kim, K. Deep Learning-Based Feature Selection for Intrusion Detection System in Transport Layer. Proceedings of the Korea Institutes of Information Security and Cryptology Conference. 2016, 740-743.

[64] Staudemeyer, R.C. Applying long short-term memory recurrent neural networks to intrusion detection. S. Afr. Comput. J. 2015, 56, 136–154.

[65] Kim, J.; Kim, H. Applying recurrent neural network to intrusion detection with hessian free optimization. In Proceedings of the International Conference on Information Security Applications, Jeju Island, Korea, 20–22 August 2015; pp. 357–369.

[66] Kim, G.; Yi, H.; Lee, J.; Paek, Y.; Yoon, S. LSTM-Based System-Call Language Modeling and Robust Ensemble Method for Designing Host-Based Intrusion Detection Systems. arXiv 2016, arXiv:1611.01726.

[67] Kim, J.; Kim, J.; Thu, H.L.T.; Kim, H. Long Short Term Memory Recurrent Neural Network Classifier for Intrusion Detection. In Proceedings of the 2016 International Conference Platform Technology and Service (PlatCon), Jeju, Korea, 15–17 February 2016; pp. 1–5.

[68] Krishnan, R.B.; Raajan, N.R. An intellectual intrusion detection system model for attacks classification using RNN. Int. J. Pharm. Technol. 2016, 8, 23157–23164.

[69] Diro, A.A.; Chilamkurti, N. Distributed attack detection scheme using deep learning approach for Internet of things. Future Gener. Comput. Syst. 2018, 82, 761–768.

[70] Diro, A.A.; Chilamkurti, N. Leveraging LSTM Networks for Attack Detection in Fog-to-Things Communications. IEEE Commun. Mag. 2018, 56, 124–130.

[71] Nadeem, M.; Marshall, O.; Singh, S.; Fang, X.; Yuan, X. Semi-Supervised Deep Neural Network for Network Intrusion Detection.

[72] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, ``Long short term memory recurrent neural network classifier for intrusion detection," in Proc. Int. Conf. Platform Technol. Service, 2016, pp. 1-5.

[73] T.-T.-H. Le, J. Kim, and H. Kim, ``An effective intrusion detection classifier using long short-term memory with gradient descent optimization," in Proc. Int. Conf. Platform Technol. Service, 2017, pp. 1-6.

[74] A. F. Agarap. (2017). ``A neural network architecture combining gated recurrent unit (GRU) and support vector machine (SVM) for intrusion detection in network traffic data." [Online]. Available: https://arxiv.org/abs/1709.03082

[75] B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, "Deep learning for classification of malware system call sequences," in AI 2016: Advances in Artificial Intelligence, 2016, pp. 137-149.

[76] J. Saxe and K. Berlin. (2017). ``eXpose: A character-level convolutional neural network with embeddings for detecting malicious urls, file paths and registry keys." [Online]. Available: https://arxiv.org/abs/1702.08568

[77] Wang, W.; Zhu, M.; Wang, J.; Zeng, X.; Yang, Z. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In Proceedings of the 2017 IEEE International Conference Intelligence and Security Informatics (ISI), Beijing, China, 22–24 July 2017; pp. 43–48.

[78] 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., 2017, pp. 712-717.

[79] Q. Tan, W. Huang, and Q. Li, ``An intrusion detection method based on DBN in ad hoc networks," in Proc. Int. Conf. Wireless Commun. Sensor Netw., 2016, pp. 477-485.

[80] G. Zhao, C. Zhang, and L. Zheng, ``Intrusion detection using deep belief network and probabilistic neural network," in Proc. IEEE Int. Conf. Comput. Sci. Eng., vol. 1, Jul. 2017, pp. 639-642.

[81] Cox, J.A.; James, C.D.; Aimone, J.B. A signal processing approach for cyber data classification with deep neural networks. Procedia Comput. Sci. 2015, 61, 349–354.

[82] Lotfollahi, M.; Shirali, R.; Siavoshani, M.J.; Saberian, M. Deep Packet: A Novel Approach for Encrypted Traffic Classification Using Deep Learning. arXiv 2017, arXiv:1709.02656

[83] Wang, Z. The Applications of Deep Learning on Traffic Identification; BlackHat: Washington, DC, USA, 2015.

[84] Mi, G.; Gao, Y.; Tan, Y. Apply stacked auto-encoder to spam detection. In Proceedings of the International Conference in Swarm Intelligence, Beijing, China, 26–29 June 2015; pp. 3–15.

[85] Tzortzis, G.; Likas, A. Deep Belief Networks for Spam Filtering. in Tools with Artificial Intelligence. In Proceedings of the 2007 19th IEEE International Conference on ICTAI, Patras, Greece, 29–31 October 2007; Volume 2, pp. 306–309.

[86] Loukas, G.; Vuong, T.; Heartfield, R.; Sakellari, G.; Yoon, Y.; Gan, D. Cloud-based cyber-physical intrusion detection for vehicles using Deep Learning. IEEE Access 2018, 6, 3491–3508.

[87] Kobojek, P.; Saeed, K. Application of recurrent neural networks for user verification based on keystroke dynamics. J. Telecommun. Inf. Technol. 2016, 3, 80–90.

[88] Zeng, F.; Chang, S.;Wan, X. Classification for DGA-Based Malicious Domain Names with Deep Learning Architectures. Int. J. Intell. Inf. Syst. 2017, 6, 67–71.

[89] Anderson, H.S.; Woodbridge, J.; Filar, B. DeepDGA: Adversarially-tuned domain generation and detection. In Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security, Vienna, Austria, 28 October 2016; pp. 13–21.

[90] Woodbridge, J.; Anderson, H.S.; Ahuja, A.; Grant, D. Predicting domain generation algorithms with long short-term memory networks. arXiv 2016, arXiv:1611.00791.

[91] Lison, P.; Mavroeidis, V. Automatic Detection of Malware-Generated Domains with Recurrent Neural Models. arXiv 2017, arXiv:1709.07102.

[92] Mac, H.; Tran, D.; Tong, V.; Nguyen, L.G.; Tran, H.A. DGA Botnet Detection Using Supervised Learning Methods. In Proceedings of the 8th International Symposium on Information and Communication Technology, Nhatrang, Vietnam, 7–8 December 2017; pp. 211–218.

[93] Yu, B.; Gray, D.L.; Pan, J.; de Cock, M.; Nascimento, A.C.A. Inline DGA detection with deep networks. In Proceedings of the 2017 IEEE International Conference Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017; pp. 683–692.

[94] Tran, D.; Mac, H.; Tong, V.; Tran, H.A.; Nguyen, L.G. A LSTM based framework for handling multiclass imbalance in DGA botnet detection. Neurocomputing 2018, 275, 2401–2413.

[95] Torres, P.; Catania, C.; Garcia, S.; Garino, C.G. An Analysis of Recurrent Neural Networks for Botnet Detection Behavior. In Proceedings of the 2016 IEEE Biennial Congress of Argentina (ARGENCON), Buenos Aires, Argentina, 15–17 June 2016; pp. 1–6.

[96] Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. arXiv 2016, arXiv:1609.04802.

[97] Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; Lee, H. Generative adversarial text to image synthesis. arXiv 2016, arXiv:1605.05396.

[98] Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; van der Smagt, P.; Cremers, D.; Brox, T. Flownet: Learning optical flow with convolutional networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 2758–2766.

[99] Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv 2015, arXiv:1511.06434.

[100] Kolias, C.; Kambourakis, G.; Stavrou, A.; Gritzalis, S. Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset. IEEE Commun. Surv. Tutor. 2015, 18, 184–208.

[101] Aminanto, M.E.; Kim, K. Improving detection of Wi-Fi impersonation by fully unsupervised deep learning. In Proceedings of the International Conference on Information Security Applications, Jeju Island, Korea, 24–26 August 2017; pp. 212–223.

[102] Maimó, L.F.; Gómez, A.L.P.; Clemente, F.J.G.; Pérez, M.G. A self-adaptive deep learning-based system for anomaly detection in 5G networks. IEEE Access 2018, 6, 7700–7712.

[103] Garcia, S.; Grill, M.; Stiborek, J.; Zunino, A. An empirical comparison of botnet detection methods. Comput. Secur. 2014, 45, 100–123.

[104] Alexa Top Sites. Available online: https://aws.amazon.com/alexa-top-sites/ (accessed on 23 April 2020).

[105] Bambenek Consulting—Master Feeds. Available online: http://osint.bambenekconsulting.com/feeds/ (accessed on 25 April 2021).

[106] DGArchive. Available online: https://dgarchive.caad.fkie.fraunhofer.de/site/ (accessed on 23 April 2020).

[107] Arp, D.; Spreitzenbarth, M.; Hubner, M.; Gascon, H.; Rieck, K.; Siemens, C.E.R.T. DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket. NDSS 2014, 14, 23–26.

[108] Microsoft Malware Classification (BIG 2015). Available online: https://www.kaggle.com/c/malware-classification (accessed on 23 April 2020).

[109] Lindauer, B.; Glasser, J.; Rosen, M.; Wallnau, K.C.; ExactData, L. Generating Test Data for Insider Threat Detectors. JoWUA 2014, 5, 80–94.

[110] Glasser, J.; Lindauer, B. Bridging the gap: A pragmatic approach to generating insider threat data. In Proceedings of the 2013 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 23–24 May 2013; pp. 98–104.

[111] EnronSpam. Available online: https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/enron-spam/ (accessed on 23 April 2020).

[112] SpamAssassin. Available online: https://www.kaggle.com/beatoa/spamassassin-public-corpus (accessed on 27 April 2020).

[113] LingSpam. Available online: https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/lingspam_public.tar.gz (accessed on 23 February 2019).

[114] KDD Cup 99. Available online: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (accessed on 23 April 2020).

[115] Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), Ottawa, ON, Canada, 8–10 July 2009; pp. 1–6.

[116] Google Play Store. Available online: https://play.google.com/store (accessed on 23 April 2020).

[117] VirusTotal. Available online: https://virustotal.com (accessed on 23 April 2020).

[118] Contagio. Available online: http://contagiodump.blogspot.com/ (accessed on 23 April 2020).

[119] Comodo. Available online: https://www.comodo.com/home/internet-security/updates/vdp/database.php (accessed on 23 April 2020).

[120] Zhou, Y.; Jiang, X. Dissecting android malware: Characterization and evolution. In Proceedings of the 2012 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 20–23 May 2012; pp. 95–109.

[121] VirusShare. Available online: http://virusshare.com/ (accessed on 25 April 2021).

[122] Arp, D.; Spreitzenbarth, M.; Hubner, M.; Gascon, H.; Rieck, K.; Siemens, C.E.R.T. DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket. NDSS 2014, 14, 23–26.

[123] https://marcoramilli.com/2016/12/16/malware-training-sets-a-machine-learning-dataset-for-everyone/ (accessed on 25 April 2021).

[124] The CTU-13 Dataset. Available online: https://stratosphereips.org/category/dataset (accessed on 25 April 2021).

[125] The UNB ISCX 2012 Intrusion Detection Evaluation Dataset. Available online: http://www.unb.ca/cic/research/datasets/ids.html (accessed on 25 April 2021).

[126] EnronSpam. Available online: https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/enron-spam/ (accessed on 25 April 2021).

[127] SpamAssassin. Available online: http://www.spamassassin.org/publiccorpus (accessed on 23 April 2020).

[128] LingSpam. Available online: https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/lingspam_public.tar.gz (accessed on 23 April 2020).

[129] Chen, J., 2020, November. A Malware Classification Method Based on Basic Block and CNN. In International Conference on Neural Information Processing (pp. 275-283). Springer, Cham.

[130] Kim, G.; Yi, H.; Lee, J.; Paek, Y.; Yoon, S. LSTM-Based System-Call Language Modeling and Robust Ensemble Method for Designing Host-Based Intrusion Detection Systems. arXiv 2016, arXiv:1611.01726.

[131] ISCX VPN-nonVPN Encrypted Network Traffic Dataset. 2017. Available online: http://www.unb.ca/cic/ research/datasets/vpn.html (accessed on 23 February 2019).

[132] Mondal, M.R.H., Bharati, S., Podder, P. and Podder, P., 2020. "Data analytics for novel coronavirus disease," Informatics in Medicine Unlocked, 20, p.100374.

[133] Bharati, S., Podder, P. and Mondal, M.R.H., 2020. "Hybrid deep learning for detecting lung diseases from X-ray images," Informatics in Medicine Unlocked, 20, p.100391.

[134] Bharati, S., Podder, P., Mondal, M.R.H. and Paul, P.K., 2021. "Applications and Challenges of Cloud Integrated IoMT," In Cognitive Internet of Medical Things for Smart Healthcare (pp. 67-85). Springer, Cham.

[135] Parvez, A.S., Robel, M.R.A., Rouf, M.A., Podder, P. and Bharati, S., 2019. "Effect of fault tolerance in the field of cloud computing," In International Conference on Inventive Computation Technologies (pp. 297-305). Springer, Cham.

[136] Robel, M.R.A., Bharati, S., Podder, P., Raihan-Al-Masud, M. and Mandal, S., 2019. "Fault tolerance in cloud computing-an algorithmic approach," In International Conference on Innovations in Bio-Inspired Computing and Applications (pp. 307-316). Springer, Cham.

[137] Podder, P., Mondal, M., Bharati, S. and Paul, P.K., 2021. "Review on the security threats of internet of things," International Journal of Computer Applications, 176(41), pp. 37-45.

[138] Bharati, S., Podder, P., Mondal, M.R.H. and Gandhi, N., 2020. "Optimized NASNet for Diagnosis of COVID-19 from Lung CT Images," In International Conference on Intelligent Systems Design and Applications (pp. 647-656). Springer, Cham.

[139] Robel, M.R.A., Bharati, S., Podder, P. and Mondal, M.R.H., 2020. "IoT Driven Healthcare Monitoring System. Fog, Edge, and Pervasive Computing in Intelligent IoT Driven Applications," pp.161-176.

[140] Podder, P., Khamparia, A., Mondal, M.R.H., Rahman, M.A. and Bharati, S., 2021. "Forecasting the Spread of COVID-19 and ICU Requirements," International Journal of Online and Biomedical Engineering (iJOE), 17(5), pp. 81-99. https://doi.org/10.3991/ijoe.v17i05.20009

[141] Podder, P., Bharati, S., Mondal, M.R.H. and Kose, U., 2021. "Application of Machine Learning for the Diagnosis of COVID-19. In Data Science for COVID-19," Academic Press, pp. 175-194.

[142] Bharati, S., Podder, P. and Mondal, M.R.H., 2020. "Artificial neural network based breast cancer screening: a comprehensive review," International Journal of Computer Information Systems and Industrial Management Applications., 12, pp.125-137.

[143] Podder, P. and Mondal, M.R.H., 2020. "Machine Learning to Predict COVID-19 and ICU Requirement," In 2020 11th International Conference on Electrical and Computer Engineering (ICECE) (pp. 483-486). IEEE.

Author Biographies

Prajoy Podder worked as a Lecturer in the Department of Electrical and Electronic Engineering in Ranada Prasad Shaha University, Narayanganj-1400, Bangladesh. He completed B.Sc. (Engg.) degree in Electronics and Communication Engineering from Khulna University of Engineering & Technology, Khulna-9203, Bangladesh. He is currently pursuing M.Sc. (Engg.) degree in Institute of Information and Communication Technology from Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh. He is a researcher in the Institute of Information and Communication Technology, Bangladesh University of Engineering & Technology, Dhaka-1000, Bangladesh. He is regular reviewer of Data in Brief, Elsevier and Frontiers of Information Technology and Electronic Engineering, Springer, ARRAY, Elsevier. He is the lead guest editor of Special Issue on Development of Advanced Wireless Communications, Networks and Sensors in American Journal of Networks and Communications. His research interest includes machine learning, pattern recognition, neural networks, computer networking, distributed sensor networks, parallel and distributed computing, VLSI system design, image processing, embedded system design, data analytics. He published several IEEE conference papers, journals and Springer Book Chapters.

Subrato Bharati received his B.S. degree in Electrical and Electronic Engineering from Ranada Prasad Shaha University, Narayanganj-1400, Bangladesh. He is currently working as a research assistant in the Institute of Information and Communication Technology(IICT), Bangladesh University of Engineering and Technology, Dhaka, Bangladesh. He is a regular reviewer of several Wiley, Elsevier and Springer International Journals. He is an associate editor of Journal of the International Academy for Case Studies. He is a member of scientific and technical program committee in some conferences including CECNet 2021, ICONCS, ICCRDA-2020, ICICCR 2021, etc. His research interest includes bioinformatics, medical image processing, pattern recognition, deep learning, wireless communications, data analytics, machine learning, neural networks, and feature selection. He published several journals paper, and also published several IEEE, Springer reputed conference papers. He published Springer and Elsevier, De Gruyter, CRC Press and Wiley Book chapters as well.

M. Rubaiyat Hossain Mondal, PhD is currently working as a faculty member in the Institute of Information and Communication Technology (IICT) at Bangladesh University of Engineering and Technology (BUET), Bangladesh. He received his Bachelor's degree and Master's degree in Electrical and Electronic Engineering from BUET. He joined IICT, BUET as a faculty member in 2005. From 2010 to 2014 he was with the Department of Electrical and Computer Systems Engineering (ECSE) of Monash University, Australia from where he obtained his PhD in 2014. He has authored a number of articles in reputed journals, conferences and book chapters. He is an active reviewer of several journals published by IEEE, Elsevier and Springer. He was a member of the Technical Committee of different IEEE International conferences. His research interest includes artificial intelligence, bioinformatics, image processing, wireless communication and optical wireless communication.

Pinto Kumar Paul received his B.Sc degree in CSE from Daffodil International University, Dhaka, Bangladesh. He currently working as a Lecturer in the Department of CSE, Ranada Prasad Shaha University, Narayanganj-1400, Bangladesh. His research interest includes NLP, Image Processing and Internet of Things.

Utku Kose, PhD received the B.S. degree in 2008 from computer education of Gazi University, Turkey as a faculty valedictorian. He received M.S. degree in 2010 from Afyon Kocatepe University, Turkey in the field of computer and D.S. / Ph. D. degree in 2017 from Selcuk University, Turkey in the field of computer engineering. Between 2009 and 2011, he has worked as a Research Assistant in Afyon Kocatepe University. Following, he has also worked as a Lecturer and Vocational School - Vice Director in Afyon Kocatepe

University between 2011 and 2012, as a Lecturer and Research Center Director in Usak University between 2012 and 2017, and as an Assistant Professor in Suleyman Demirel University between 2017 and 2019. Currently, he is an Associate Professor in Suleyman Demirel University, Turkey. He has more than 100 publications including articles, authored and edited books, proceedings, and reports. He is also in editorial boards of many scientific journals and serves as one of the editors of the Biomedical and Robotics Healthcare book series by CRC Press. His research interest includes artificial intelligence, machine ethics, artificial intelligence safety, optimization, the chaos theory, distance education, e-learning, computer education, and computer science.