IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 24, 2020, accepted July 8, 2020, date of publication July 16, 2020, date of current version July 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009849

Hybrid Deep Learning: An Efficient Reconnaissance and Surveillance Detection Mechanism in SDN

JAHANZAIB MALIK[1], ADNAN AKHUNZADA[2], IRAM BIBI[1], MUHAMMAD IMRAN[1], (Member, IEEE), ARSLAN MUSADDIQ[3], AND SUNG WON KIM[3]

[1]Computer Science Department, COMSATS University Islamabad, Islamabad 46000, Pakistan
[2]DTU Compute, Technical University of Denmark, 2800 Lyngby, Denmark
[3]Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, South Korea

Corresponding author: Sung Won Kim (swon@yu.ac.kr)

This research was supported in part by the Brain Korea 21 Plus Program (No. 22A20130012814) funded by the National Research Foundation of Korea (NRF), in part by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2020-2016-0-00313) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation), and in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2018R1D1A1A09082266).

ABSTRACT Software defined network (SDN) centralized control intelligence and network abstraction aims to facilitate applications, service deployment, programmability, innovation and ease in configuration management of the underlying networks. However, the centralized control intelligence and programmability is primarily a potential target for the evolving cyber threats and attacks to throw the entire network into chaos. The authors propose a control plane-based orchestration for varied sophisticated threats and attacks. The proposed mechanism comprises of a hybrid Cuda-enabled DL-driven architecture that utilizes the predictive power of Long short-term memory (LSTM) and Convolutional Neural Network (CNN) for an efficient and timely detection of multi-vector threats and attacks. A current state of the art dataset CICIDS2017 and standard performance evaluation metrics have been employed to thoroughly evaluate the proposed mechanism. We rigorously compared our proposed technique with our constructed hybrid DL-architectures and current benchmark algorithms. Our analysis shows that the proposed approach outperforms in terms of detection accuracy with a trivial trade-off speed efficiency. We also performed a 10-fold cross validation to explicitly show unbiased results.

INDEX TERMS Security, hybrid deep learning model, software defined networks, long short-term memory, convolutional neural network.

I. INTRODUCTION

THE traditional internet architecture is turned out to be a complex system because of decentralization with huge numbers of network devices such as routers, switches, middleboxes involving various layers, etc. Subsequently, the conventional network architecture also fails to respond dynamic nature of modern applications [1]. Software defined networks (SDNs) emerged as a promising network architecture with extraordinary adaptability in configuration and management, enabling scalability and deployment of network services. Being programmable innovative control plane of the SDN makes it a promising emerging next generation networking architecture [2]. On the contrary, the centralized control intelligence of SDN puts the control plane at a risk of being a single point of failure and potential primary target for varied adversarial sophisticated attacks.

The authors orchestrate control plane to propose a highly scalable deep learning (DL) based intrusion detection mechanism to combat multi-vector sophisticated cyber threats and attacks.

Additionally, the control plane can be extended to various underlying networks such as SDN-based fog computing, edge computing, and SDN-IoT, etc. [3], [4]. Further, the control plane-based orchestration for various underlying functions makes the SDN controller as the most potential part of the SDN architecture. On the contrary, the control plane is vulnerable and can be potentially targeted with varied evolving

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

sophisticated lethal cyber threats and attacks such as denial of service (DoS), distributed denial of service (DDoS), Brute Force, web attacks and other application level attacks such as SQL injections, cross site scripting etc., that can simply lead to compromise the accessibility and confidentiality of data, processes of application or can even disrupt the entire network [5]–[8]. The SDN networks beside leveraging huge benefits also present deviating security concerns and evolving threats. A timely and efficient Protection of the control plane from varied sophisticated multi-vector intrusions is a real challenge.

Besides, intrusion detection system (IDS) is a basic tool used to identify different cyber-attacks inside a system. An IDS screens and examines the system traffic going into or leaving from the system applications to raise cautions if an interruption or an abnormal activity is observed [9]. There are two types of IDS mainly named as signature-based detection and anomaly-based detection system. Signature based detection, extracts new information and compares it existing known intrusion signatures. Updated signatures are always needed to identify latest attacks. However, signature-based IDS fail to identify zero-day attacks [10]. Whereas, anomaly-based detection contrasts new information and models normal behaviour while denotes a significant deviation from this model as an anomaly [11], [12]. The literature is evident with ML-based approaches for varied detection mechanisms in SDNs [13]. On the other hand, the current literature shows tendency and a slight shift from ML to DL-based approaches [14], [15]. The reason behind is that DL does not experience extra processing for feature selection and are quite efficient. Besides, the advantage of DL-based anomaly detection approaches is having the capability to identify zero-day attacks.

Contributions: The principle contributions of this work are as follows:

- A control plane-based orchestration of a hybrid DL based intrusion detection mechanism that is highly scalable, flexible, and programmable to effectively identify multi-vector sophisticated evolving cyber threats and attacks. The proposed framework utilizes the predictive power of Long short-term memory (LSTM) and Convolutional Neural Network (CNN). Besides, the proposed work can be extended, customized and deployed to any commercial SDN controller such as OpenDaylight, Floodlight, NOX, POX etc.
- For practical experimentation, a current state of the art flow-based dataset (i.e., CICIDS2017) has been employed for a multi-class attack detection system.
- For comprehensive evaluation of the proposed system, standard and extended evaluation metrics have been utilized to gauge closely the actual performance of the system (i.e., accuracy, precision, recall, F1-score, TNR, NPV, FPR, FNR, FDR and MCC etc.).
- We have thoroughly compared our proposed technique with our constructed hybrid DL architectures and current benchmark algorithms. The proposed mechanism outperforms in terms of detection accuracy with a trivial trade off in speed efficiency.
- Additionally, we have employed 10-fold cross validation to ensure unbiased results.

Structure: The remaining work is structured in the following way; section II represents background and related work. Section III contains proposed methodology including the complete description of proposed framework, dataset description, pre-processing of dataset, basic architectural description of used deep learning algorithms, experimental setup and evaluation metrics. Section IV presents results and discussions. Section V concludes the work and defines future directions and recommendations.

II. BACKGROUND AND RELATED WORK

Software defined networking (SDN) is a promising next generation networking paradigm. The architecture of SDN is portrayed by the partition of the control plane from the data plane and application plane. An intelligently centralized logical entity defined as control plane gives an abstract view of the underlying network. Besides, the control plane is the central decision maker and controls data processing and forwarding capabilities of the entire network. The collection of basic forwarding devices and underlying SDN agents represents data plane. As SDN depends mainly on the control plan because it is programmable and have the capability to extend and implement various modules. Thus, SDNs are flexible and innovative networks. A comprehensive and detailed overview of the SDN architecture can be found in author's published work [8], [13], [16]. Every SDN controller has ability to extend varied modules. Consequently, the authors have implemented their detection module at the SDN control plane. Mostly, the design architecture of different SDN controller are same but varies in functionalities. Besides, the language of implementation differs from controller to controller. For instance, POX is written in Python, whereas; Floodlight is written in Java and so on. Consequently, our proposed detection module can simply be customized and extended to any commercial SDN controller using a programming language wrapper. Fig.1 shows the basic architectural description of SDN with our proposed detection module at control plane.

Moreover, the sequence diagram (see Fig.2) shows the communication between the OpenFlow switch, controller and proposed detection module. Sequence diagram specify types of APIs used to communicate with controller and the proposed detection module for SDN environment. The diagram shows the prime setup of the OpenFlow switch with the SDN controller. After initial negotiation of OpenFlow switch and SDN controller, some of the important messages are exchanged. Consequently, the sequence diagram also depicts some of the important message's (i.e., Feature Request, Feature Reply, Packet In, Packet Out, Flow-Mod and Flow-Removed) exchanged between controller and

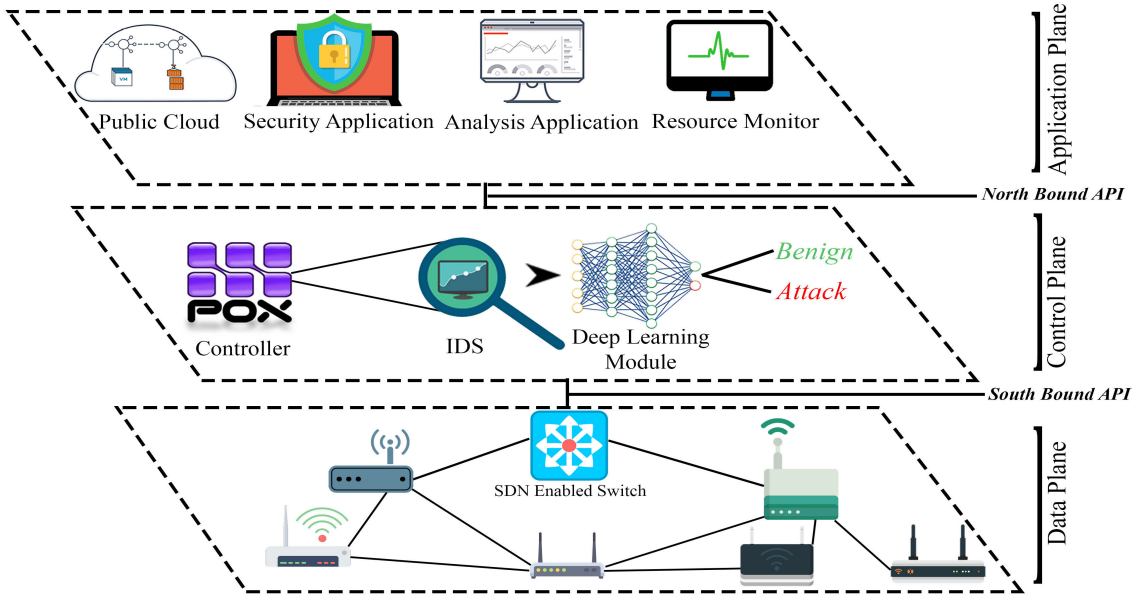


FIGURE 1. Architecture of SDN incorporated with proposed solution.

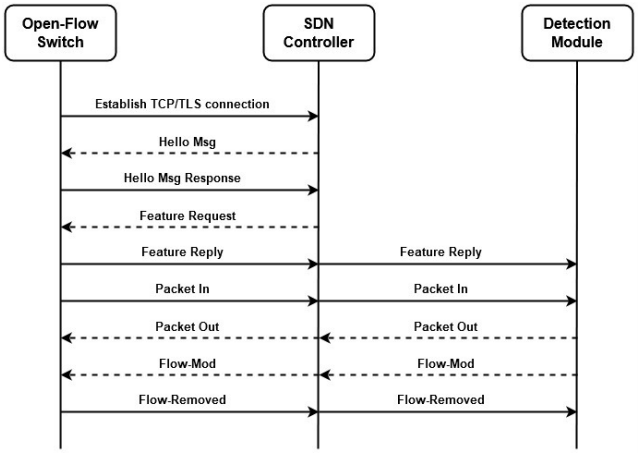


FIGURE 2. Sequence diagram of OpenFlow with SDN controller and proposed detection module.

detection module. The detection module receives the messages to be analysed for vital features analysis and drop the flow if the module finds it malicious. The key features that our detection module is looking for to identify a malicious attack are shown in Table 4.

Deep learning gathered significant interest and its applications are being explored inside many research areas, for example healthcare, automotive design and law implementation. There are likewise several existing works inside the area of NIDS in SDN [17], [18].

In [19] the authors proposed a deep learning-based intrusion detection framework employing Restricted Boltzmann Machine (RBM) in software defined networks (SDNs). For practical experimentation the technique utilized CMU based insider threat dataset and KDD99 dataset and achieved performance with 99.98% detection accuracy for binary classification. *Tang et al.* in [20] proposed a profound Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) based IDS utilizing NSL-KDD and CICIDS2017 dataset and achieved 89% detection accuracy for multi classification. However, the architecture of SDN Comprised on flows and NSL-KDD is not at all a flow-based dataset. Authors in [21], propose a deep learning system based on Multi-Layer Perceptron (MLP) for threat and attack detection for software defined networking. The proposed framework achieves 98.7% detection accuracy using CTU-13 dataset. Birkinshaw *et al.* [22] implement connection-based techniques as part of the intrusion detection and prevention system (IDPS), known as the Credit-Based Threshold Random Walk (CB-TRW) and Rate Limiting (RL) for port scan and DoS attacks detection in SDN. The data for the experimentation the network packets has been captured through five minutes of network traffic. For 10k packets per second the false positive rate is 0% while CPU usage is 97%.

Moreover, the authors in [23] proposed network intrusion detection framework using Recurrent Neural Network (RNN), Long short-term memory (LSTM) and Convolutional Neural Network (CNN). Particularly, DDoS attack detection using deep CNN ensemble framework has been published in our recent work [8]. The framework achieves 98% detection accuracy using ISCX2012 dataset. The author in [24] proposed a network intrusion detection system employing Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) using six basic features of NSL-KDD dataset.

TABLE 1. Comprehensive comparison of existing literature work.

Ref	Year	Attacks	Dataset	Algorithms	Detection approach	Limitations	Future work
[19]	2019	DDoS	CMU based insider threat dataset, KDD 99	RBM, SVM	Propose intrusion detection framework leveraging ensemble RBM and SVM with accuracy of 89% for multiclass.	Old, static data and not a flow-based dataset.	Need of real time anomaly detection.
[20]	2019	DoS, Probe, U2L, R2L	NSL-KDD, CICIDS2017	GRU-RNN	GRU-RNN classifier is used for anomaly detection and achieves an accuracy of 89% for multi-class.	NSL KDD is not a flow-based dataset.	Use diverse features to enhance performance of classifier and reduce the overhead on the controller.
[21]	2019	Botnet	CTU-13 ISOT	MLP	Intrusion detection mechanism based on MLP to detect botnet in SDN.	Not performs experiment on terminals that are infected with bots.	Explore network isolation can be performed on an infected host.
[22]	2019	Port-scan, DoS	Real traffic captured	CB-TRW, RL	The portscan and DoS detection and prevention scheme is proposed through CB-TRW and RL in software defined networking.	False positive rate and CPU usage is only defined to show the performance of detection and prevention mechanism.	Implementation of countermeasures for diverse attacks and protocols.
[23]	2018	DDoS	ISCX2012	RNN, LSTM, CNN	Feature Filtration has been performed and verification accuracy crossed 98%.	Computational complex and time overhead detection scheme.	N/A
[24]	2018	DoS, U2L, R2L	NSL-KDD	GRU-RNN	The proposed GRU-RNN technique achieve a detection accuracy of 89% using just 6 network features.	NSL-KDD is not flow based dataset.	Optimize the classifier and utilize more features to enhance the accuracy.
[25]	2018	Port Scan	Wireshark and tshark tool	MLP, CNN, LSTM	Solved the port scanning issue through Fast Gradient Sign Method (FGSM), JSMA reverse (JSMA-RE) and Jacobian-based Saliency Map Attack (JSMA) schemes.	Computational complex scheme.	Conduct more studies on adversarial attacks such as DDoS, U2L and R2L.
[26]	2018	TCP, UDP, ICMP	Barnyard	DNN	Flow-based anomaly detection worked on Snort with deep learning and achieved 85% detection accuracy.	Not achieved promising detection accuracy, Deployment in real-time environment has not been checked.	Improve DNN model performance by varying with rectification of training and DNN model.
[27]	2018	DoS, U2L, R2L	NSL-KDD	DT, ELM, NN, SVM, Ada-Boost	Presents an anomaly-based intrusion detection system in SDN achieving 80% detection accuracy.	The performance of classifier is not enough for its deployment in real environment.	Explore flow-based features that achieves a higher level of accuracy with lower false alarm rate.
[28]	2018	DoS, probe, U2L, R2L	NSL-KDD	GRU-LSTM	Hybrid framework leveraging GRU-LSTM is proposed while achieves 87% classification accuracy	The detection accuracy is not good enough to detect the intrusions in run time.	Plans to implement in a real Environment of SDN for real traffic.
[29]	2018	Botnet	Real traffic	MLP	Present Botnet attack detection mechanism employing MLP and achieves 98% detection accuracy.	The experiment is just evaluated on real traffic	Analysis of performance of the detection application for Big-Data techniques.
[30]	2017	DOS, U2R, Probe, R2L	NSL-KDD	RNN	The propose work achieved performance for multi-class classification with 81.29% accuracy.	Comparison has been performed with machine learning algorithm.	Reduction of training time.
[31]	2016	DDoS	Network traffic gathered from home wireless network	SAE	Feature extraction and detection has been performed through SAE and performance is 95.65% with 64 features.	Limits controller performance. Not SDN-based network traffic has been captured.	Extraction of features from raw bytes of packets.
[32]	2016	DDoS	Network traffic	BPNN	Propose DDoS detection and prevention mechanism while detecting and tracing back attack source in less than 1s.	Not evaluated on current benchmark dataset.	To unfold new anomaly detection technique in SDN.

The proposed methodology achieves 89% detection accuracy which is not good enough to detect real time evolving cyber threats and attacks. *Huang et al.* in [25] propose a deep learning-based anomaly detection system using Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN) and Long short-term memory (LSTM) for port scanning. The data for practical experimentation has been gathered via Wireshark and T-shark tool.

In [26], the authors propose a deep learning approach based on a deep neural network (DNN) for flow-based intrusion. The proposed framework achieved 85% detection accuracy using Barnyard with Snort a network-based intrusion detection system. Moreover, an intelligent IDS in SDN is proposed through using various machine learning classifiers (i.e., Decision Tree, Extreme Learning Machine, Support Vector Machine, Ada-Boost) and a deep learning model known as Artificial Neural Network (ANN) in [27]. The proposed framework show detection accuracy of 80% using NSL-KDD dataset. Similarly, in [28], a hybrid of Gated Recurrent Unit Long short-term memory (GRU-LSTM) is proposed for Flow-based anomaly detection in an SDN environment. ANOVA F-Test and Recursive feature Elimination (RFE) (ANOVA F-RFE) feature selection methods have been applied for enhancing the classifier performance. NSL-KDD is set as a benchmark dataset for test and experiments and obtains 87% detection accuracy.

The authors in [29] present a Botnet detection mechanism based on Multilayer Perceptron (MLP) in SDN. The real traffic data has been captured for experimentation and achieves 98% detection accuracy. *Yin et al.* in [30], Presented a intrusion detection framework using Recurrent Neural Networks (RNN-IDS) trained on NSL-KDD dataset. This model is evaluated for classification of network traffic with accuracy of 81.29% for multi-class classification. A DDoS detection framework in [31] is proposed leveraging stacked Autoencoder in SDN. For practical experimentation the network traffic has been collected from home wireless network and showed 95.65% detection accuracy with 64 basic features. To detect and mitigate the DDoS attacks in SDN the authors in [32] propose a deep learning mechanism Back Propagation Neural Network (BPNN). A feature set of malicious and benign traffic has been captured for dataset. The proposed framework can detect attack in less than 1 second and also trace the source switch.

The findings from our relevant literature, also defined in Table 1 have indicated that regardless of the high identification accuracy being accomplished, there is still opportunity to get better. Most of the proposed intrusion detection mechanisms are designed for binary classification only. Moreover, ensemble and hybrid models are still in its infantile stages for comprehensive evaluation. Henceforth, we also strongly belief the proposed hybrid architecture in this study make a valid contribution to the present pool of research.

III. METHODOLOGY

This section represents the complete proposed methodology of intrusion detection system including description of system architecture, pre-processing, dataset and employed deep learning algorithms.

A. PRE-PROCESSING

To increase effectiveness and performance of our proposed hybrid deep learning technique, the pre-processing of CICIDS2017 has been performed. Initially, the instances with missing, nan and infinity values has been dropped from dataset. Also, deep learning classifiers input numeric values only so the conversion of non-numeric value into numeric values has been performed. For improving the quality of dataset, the data normalization is also executed which includes scaling all value in range from 0-1 using MinMaxScalar function.

B. PROPOSED HYBRID DEEP LEARNING FRAMEWORK

For detection of sophisticated attacks, the proposed hybrid deep learning framework is designed employing Long short-term memory (LSTM) and Convolutional Neural Network (CNN). The schematic overview of our proposed hybrid (LSTM, CNN) architecture can be seen in Fig.3. The hybrid model postulated in this study is a combination of two deep learning algorithms (i.e., LSTM, CNN). Hybrid Models are next generation of artificial intelligence which consolidate at least two or more classifiers from same or distinct families to get higher accuracy and out performance from most of the conventional A.I classifiers. In our proposed hybrid framework, LSTM is employed as a method to achieve effective modelling for very long sequences from dataset, sustaining greatly for prevention of gradient vanishing problem also. Consequently, CNN is a well-known deep learning algorithm used for efficient extraction of features from raw data. To utterly get benefit from qualities of diverse deep learning classifiers at the same time, we have selected LSTM and CNN for improving overall results in terms of detection accuracy, precision, recall, F1-score and time efficiency. The training of algorithms is performed individually but the testing results are merged to achieve out-performance from hybrid framework. The detailed arrangement of proposed hybrid architecture is depicted in Table 2.

C. DATASET

Intrusion detection system (IDS) are considered most significant tool against advance and ever-growing cyber threats. However, lack in reliable and current state of the art dataset, anomaly-based intrusion detection system is experiencing accurate performance evaluation issues. To overcome the performance concern of IDS, we have designed our proposed system with CICIDS2017 [33], purely a flow-based dataset for software defined network (SDN). The dataset is consisted of benign and most up to date common attacks

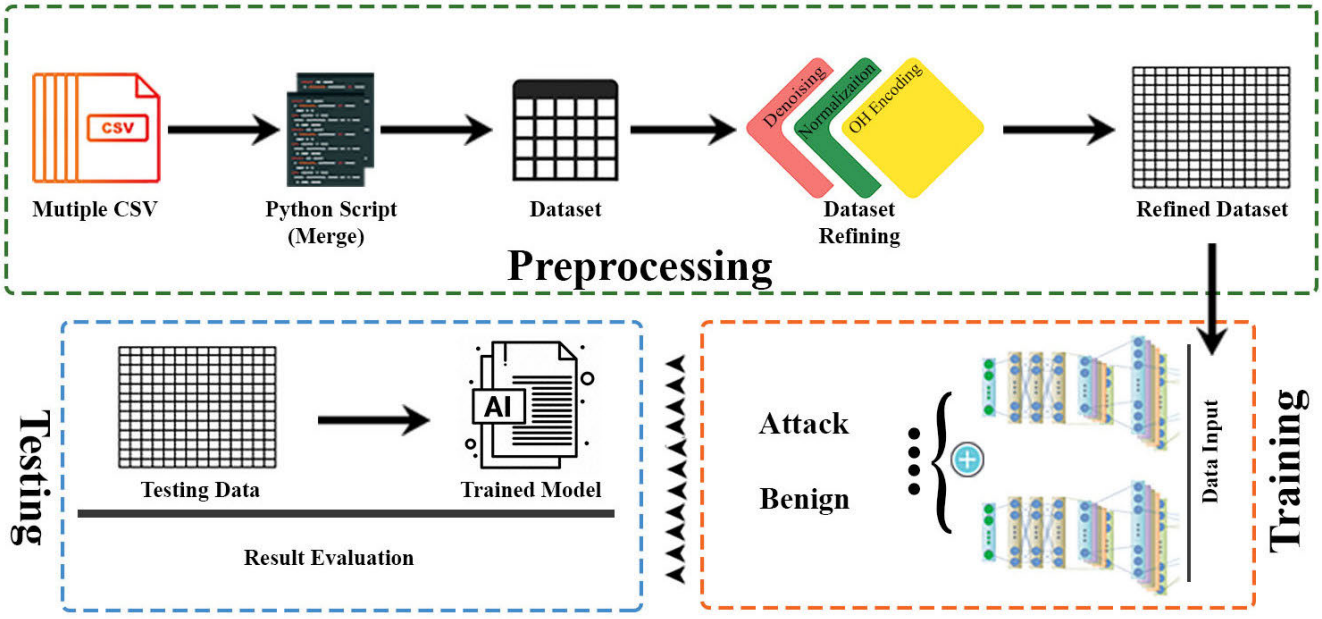


FIGURE 3. A simplified overview of proposed hybrid DL-driven architecture (LSTM, CNN).

TABLE 2. Detailed technical description of proposed and other comparative algorithms.

Algorithm	Layers	Neurons/Kernal	AF/ LF	Optimizer	Epochs	Batch-size
LSTM-CNN	*Conv Layer* (3)	(20, 15, 10)	*RelU/CC-E*	*Adam*	10	16
	LSTM(3)	(20, 15, 10)	-			
	Merge Layer		-			
	Dense Layer	40	-			
	Dense Layer	15	-			
	Output Layer	4	*softmax*			
LSTM-DNN	*Conv Layer* (3)	(20, 15, 10)	*RelU/CC-E*	*Adam*	10	16
	DNN(3)	(20, 15, 10)	-			
	Merge Layer		-			
	Dense Layer	40	-			
	Dense Layer	15	-			
	Output Layer	4	*softmax*			
LSTM-GRU	*Conv Layer* (3)	(20, 15, 10)	*RelU/CC-E*	*Adam*	10	16
	GRU(3)	(20, 15, 10)	-			
	Merge Layer		-			
	Dense Layer	40	-			
	Dense Layer	15	-			
	Output Layer	4	*softmax*			

AF = Activation Function. LF = Loss Function. CC-E = categorical cross-enotropy.

(i.e., Port Scan, Cross Site Scripting (XSS), Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet and DDoS). From total 15 label classes, in this study we are only concerned with 3 diverse application level attacks (i.e., Port Scan, Cross Site Scripting, Botnet) and one benign class. The complete distribution of utilized set of CICIDS2017 for proposed approach is depicted in Table 3.

Our reason for selection of particular classes (i.e., Bot, Port Scan and XSS) is to focus on the detection of attacks/mechanisms which are associated with reconnaissance which is basically an information gathering and surveillance technique to exploit systems. The proposed system is intentionally targeted to detect reconnaissance activities by the attacker in the earlier stage.

TABLE 3. Distribution table of CICIDS2017 Dataset for practical experimentation.

CLASSES	NUMBER OF INSTANCES
Benign	1,200
Botnet	1,966
Port Scan	1,309
Cross Site Scripting (XSS)	652
Total	**5,127**

TABLE 4. The list of features in CICIDS2017 Dataset.

Sr. No	Features Names	Sr. No	Features Names	Sr. No	Features
1	Source IP	**29**	Fwd IAT Max	**57**	Average Packet Size
2	Source Port	**30**	Fwd IAT Min	**58**	Avg Fwd Segment Size
3	Destination IP	**31**	Bwd IAT Total	**59**	Avg Bwd Segment Size
4	Destination Port	**32**	Bwd IAT Mean	**60**	Fwd Header Length
5	Protocol	**33**	Bwd IAT Std	**61**	Fwd Avg Bytes/Bulk
6	Timestamp	**34**	Bwd IAT Max	**62**	Fwd Avg Packets/Bulk
7	Flow Duration	**35**	Bwd IAT Min	**63**	Fwd Avg Bulk Rate
8	Total Fwd Packets	**36**	Fwd PSH Flags	**64**	Bwd Avg Bytes/Bulk
9	Total Backward Packets	**37**	Bwd PSH Flags	**65**	Bwd Avg Packets/Bulk
10	Total Length of Fwd Packets	**38**	Fwd URG Flags	**66**	Bwd Avg Bulk Rate
11	Total Length of Bwd Packets	**39**	Bwd URG Flags	**67**	Subflow Fwd Packets
12	Fwd Packet Length Max	**40**	Bwd Header Length	**68**	Subflow Fwd Bytes
13	Fwd Packet Length Min	**41**	Fwd Packets/s	**69**	Subflow Bwd Packets
14	Fwd Packet Length Mean	**42**	Bwd Packets/s	**70**	Subflow Bwd Bytes
15	Fwd Packet Length Std	**43**	Min Packet Length	**71**	Init_Win_bytes_forward
16	Bwd Packet Length Max	**44**	Max Packet Length	**72**	Init_Win_bytes_backward
17	Bwd Packet Length Min	**45**	Packet Length Mean	**73**	act_data_pkt_fwd
18	Bwd Packet Length Mean	**46**	Packet Length Std	**74**	min_seg_size_forward
19	Bwd Packet Length Std	**47**	Packet Length Variance	**75**	Active Mean
20	Flow Bytes/s	**48**	FIN Flag Count	**76**	Active Std
21	Flow Packets/s	**49**	SYN Flag Count	**77**	Active Max
22	Flow IAT Mean	**50**	RST Flag Count	**78**	Active Min
23	Flow IAT Std	**51**	PSH Flag Count	**79**	Idle Mean
24	Flow IAT Max	**52**	ACK Flag Count	**80**	Idle Std
25	Flow IAT Min	**53**	URG Flag Count	**81**	Idle Max
26	Fwd IAT Total	**54**	CWE Flag Count	**82**	Idle Min
27	Fwd IAT Mean	**55**	ECE Flag Count	**83**	Label
28	Fwd IAT Std	**56**	Down/Up Ratio		

D. ALGORITHMS

In this section we briefly explain deep learning algorithms like Long short-term memory (LSTM), Convolutional Neural Network (CNN) used in our experiment.

1) LONG-SHORT-TERM MEMORY (LSTM)

Long short-term memory (LSTM) is an expansion or variant of Recurrent Neural Network (RNN). The traditional RNN suffer from short term memory like if the sequence is long, the classifier finds difficulties to carry information from earlier time steps to later ones. So, recurrent neural network suffers from gradient vanishing problem and LSTM are designed as a solution. The basic architecture of LSTM considers the concept of utilizing the gates mechanism for units that regulate the flow of information. The gating mechanism contains three gates (i.e., input, output, and forget gate). These gates are responsible to decide which sequence of data is important to store or throw away. The input and output gates control the flow of input and output information of cell. Further, if the information is inappropriate; forget gate operates to reset their previous state information. The sigmoid is used as an activation function for gates and it squishes values between 0 and 1. The complete architectural description can be seen in [34].

2) CONVOLUTIONAL NEURAL NETWORK (CNN)

The great advancements in computer vision with deep learning for multitude of tasks such as Image & Video recognition, Image Analysis & Classification, Media Recreation, Recommendation Systems, Natural Language Processing has been constructed over one particular algorithm a Convolutional Neural Network (CNN). CNN is a collection of processing elements interconnected, aim to transform set of input to set of desired output. In CNN there are various operations performed on input data which include operations using convolution layer, pooling layer, flattening layer and padding and then the network relates to fully connected neural network. In CNN classification system, each feature element has one input node for its value as input to classifier, some hidden layers for learning and a single output node for the result. The complete architectural description can be seen in [35].

TABLE 5. Hardware and software specifications for experimentation setup.

CPU	**Processor**	**Model**	**Generation**
	Core-i9 (4.0Ghz)	9900K	Nineth(9th)
GPU	NVIDIA GTX-1080		
OS	Windows 10 - 64 Bit		
RAM	32GB - 3600 MHz		
Language	Python		
Libraries	Numpy, Tensorflow, Scikitlearn, Pandas		
IDE	Anaconda Spyder		

E. EXPERIMENTAL SETUP

The framework used to perform experimentation for this research work is Google's TensorFlow [36] framework. An efficient and user-friendly python library also known as Keras [37] is also utilized to design and implement

TABLE 6. 10 Fold of accuracy, precision, recall and F1-score for hybrid Cu-(LSTM & CNN), hybrid Cu-(LSTM & DNN) and hybrid Cu-(LSTM & GRU).

	Accuracy (%)			Recall (%)			Precision (%)			F1-Score (%)		
Folds	@@	«»	++	@@	«»	++	@@	«»	++	@@	«»	++
1	98.55	98.55	98.67	99.19	99.39	99.49	98.39	98.19	98.29	99.19	99.39	99.49
2	97.65	98.55	98.49	98.48	99.19	99.19	97.60	98.40	98.30	98.48	99.19	99.19
3	98.55	**98.85**	98.61	99.49	99.39	99.39	98.10	**98.69**	98.29	99.49	99.39	99.39
4	**98.97**	98.43	98.49	99.59	99.19	99.19	**98.69**	98.19	98.29	99.59	99.19	99.19
5	98.31	98.37	98.43	99.08	99.49	99.09	98.09	97.80	98.29	99.08	99.49	99.09
6	98.73	98.67	98.25	99.59	99.29	**99.69**	98.30	98.49	97.40	99.59	99.29	**99.69**
7	98.79	98.43	98.73	**99.69**	99.49	99.29	98.29	97.89	98.59	**99.69**	99.49	99.29
8	98.73	98.49	98.25	99.39	**99.69**	99.59	98.49	97.79	97.49	99.39	**99.69**	99.59
9	98.79	98.31	**98.97**	99.59	98.89	99.59	98.39	98.29	**98.69**	99.59	98.89	99.59
10	98.97	98.49	98.61	98.90	98.99	98.99	99.39	98.49	98.69	98.90	98.99	98.99

Abbreviation Terms: @@ Hybrid Cu-(LSTM & CNN), «» Hybrid Cu-(LSTM & DNN), ++ Hybrid Cu-(LSTM & GRU)

our proposed hybrid intrusion detection model for intrusion detection. The practical experiment is performed using mention settings for hardware and software in Table 5.

F. EVALUATION METRICS

For comprehensive performance measurement, we have used standard evaluation parameters known as Accuracy, Precision, Recall and f1-score. Further, True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) values are calculated from Confusion Matrix. Basic mathematical formulas to calculate standard evaluation metrics are defined below:

1) ACCURACY

Accuracy shows the percentage of correctly labelled records. The detection accuracy also shows the ability of an algorithm.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

2) POSITIVE PREDICTIVE VALUE (PPV) OR PRECISION

PPV is also called Precision that shows the number of correctly predicted records.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

3) TRUE POSITIVE RATE (TPR) OR RECALL

The TPR which is also called Recall shows the number of correctly predicted records out of total records.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

4) F1-SCORE

It presents harmonic mean between recall and precision.

$$F1 - score = \frac{2 * TP}{2 * TP + FP + FN} \tag{4}$$

5) AU-ROC CURVE

ROC curve shows the Plotting of TP rate and FP rate in 2D. The area under cover is the overall performance of system,

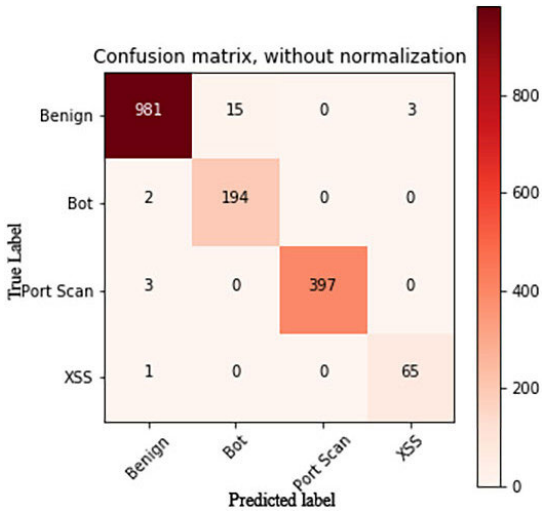


FIGURE 4. Confusion matrix of proposed hybrid model.

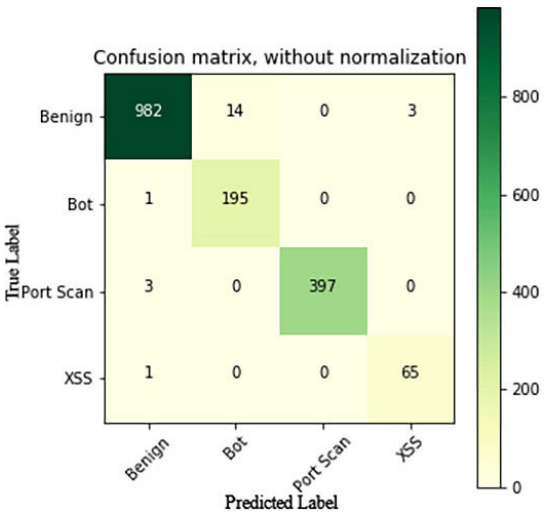


FIGURE 5. Confusion matrix of proposed hybrid model.

more area under curve higher the performance and efficiency of model.

For better evaluation of our proposed work, we have further calculated other performance evaluation metrics known

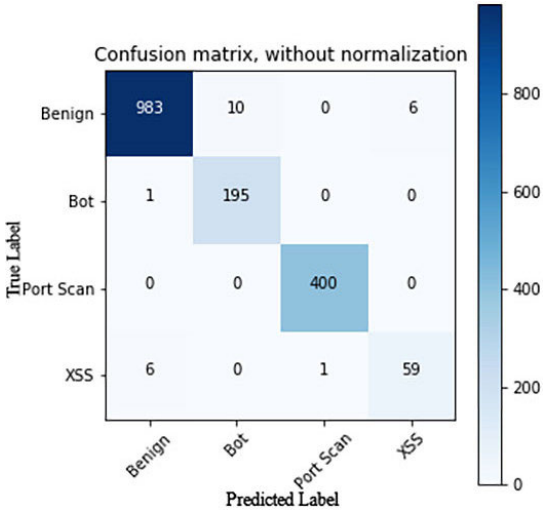


FIGURE 6. Confusion matrix of proposed hybrid model.

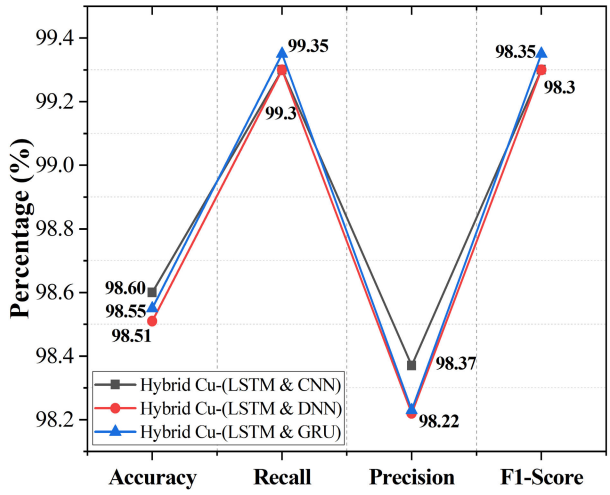


FIGURE 7. Accuracy, recall, precision and F1-score values of proposed Cuda-enabled hybrid model.

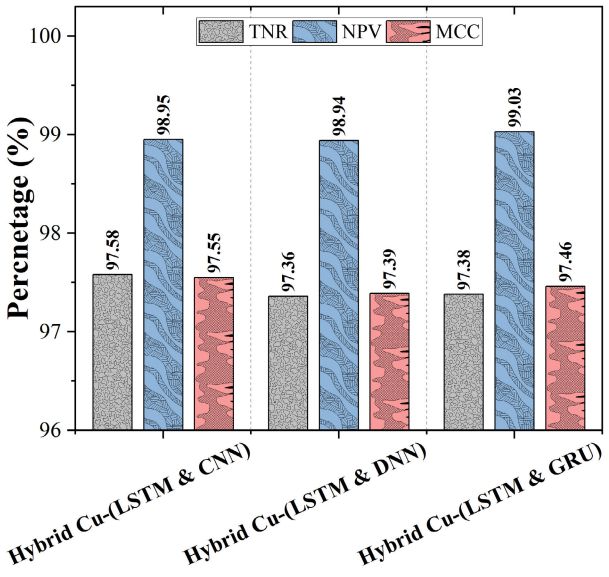


FIGURE 8. TNR, NPV and MCC rate of our proposed Cuda-enabled hybrid model.

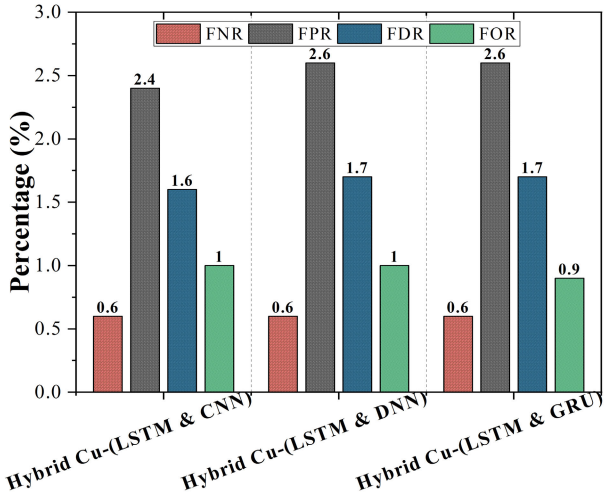


FIGURE 9. FNR, FPR, FDR and FOR rate of our proposed Cuda-enabled hybrid model.

as True Negative Rate (TNR), Negative Predictive Value (NPV) and Matthews Correlation Coefficient (MCC), Bookmarker Informadness (BM), Markedness (MK), False Discovery Rate (FDR), False Positive Rate (FPR), False Negative Rate (FNR), and False Omission Rate (FOR). The detailed description of standard performance evaluation metric can be visualized in [38].

IV. RESULTS AND DISCUSSION

To show the performance of our proposed framework, we used 10-fold cross validation technique. Fig. (4-6) shows the performance of our proposed hybrid DL-driven technique through confusion matrix. Our proposed technique identifies three different classes of attacks (i.e., Cross Site Scripting (XSS), Botnet, Port Scan) correctly with very low false rate.

To expressly show unbiased results we also employed 10-fold cross validation technique. The outcome of each fold for the referenced measurements like Detection Accuracy, Precision, Recall and F1-Score can be found in Table 6. Detection accuracy is considered as the prime evaluation metric to rigorously evaluate the proposed system. The average values of 10 folds for detection accuracy, precision, recall and F-measure are defined in Fig.7. The detection accuracy of 98.6% of hybrid Cuda-enabled deep learning framework (LSTM, CNN) shows the out performance of system as compared to other constructed hybrid frameworks including (LSTM, DNN) and (LSTM, GRU).

For better evaluation of proposed work, we also observed True Negative Rate (TNR), Negative Predictive Value (NPV) and Matthews Correlation Coefficient (MCC) values and presented in Fig.8. The proposed Cuda-enabled hybrid deep learning framework is showing outstanding results with

TABLE 7. Comparison of proposed GPU-empowered GRU technique with other contemporary existing state-of-the-art solutions for android Malware detection.

Schemes	Algorithms	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	D.Time
Proposed	Cu(LSTM-CNN)	CICIDS2017	98.60	99.37	99.35	99.35	296(ms)
[20]	GRU-RNN	CICIDS2017	89.00	99.00	99.00	99.00	-
[28]	GRU-LSTM	NSL-KDD	87.90	83.50	77.90	80.60	-
[23]	RNN,LSTM,CNN	ISCX2012	98.00	-	-	-	-

Abbreviation Terms: GRU-CNN – Gated Recurrent Unit- Convolutional Neural Network, DNN – Deep Neural Network, LSTM – Long short-term memory, D.Time– Detection Time.

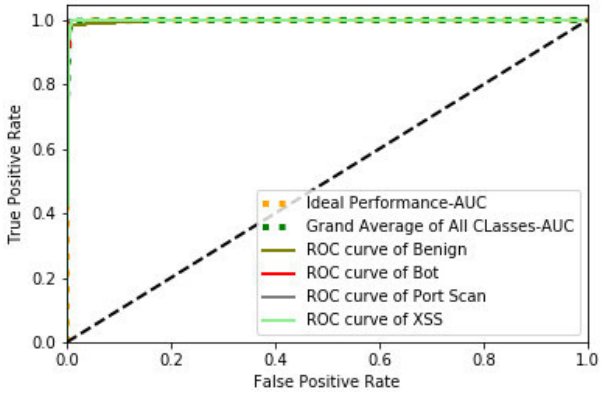


FIGURE 10. ROC Curve of proposed Cuda-enabled hybrid model.

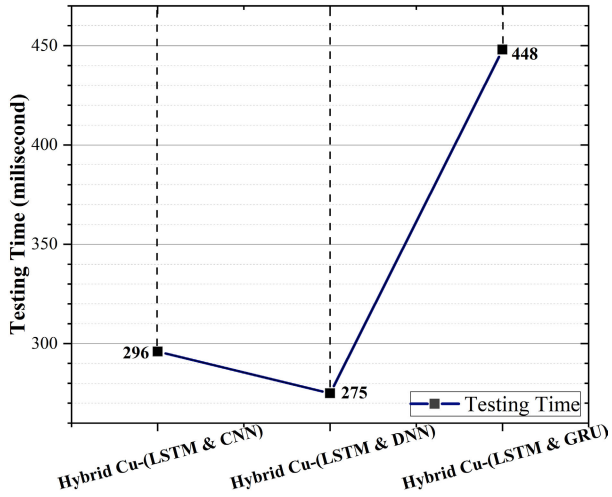


FIGURE 11. Testing time of our proposed Cuda-enabled hybrid model.

97.58%, 98.95%, and 97.55% for TNR, NPV and MCC values respectively.

Other results such as False Negative Rate (FNR), False Positive Rate (FPR), False Discovery Rate (FDR) and False Omission Rate (FOR) are also presented in Fig.9. The low rate of proposed framework shows better performance of our proposed hybrid classifier. Our hybrid of LSTM and CNN achieve 0.6%, 2.4%, 1.6 % and 1% for FNR, FPR, FDR and FOR respectively. We have also compared the achieved results of our proposed Cuda-empowered hybrid of LSTM and CNN model against the existing state of the art works in Table 7. The comparison show that our achieved results are promising when contrasted with other deep learning-based strategies for intrusion detection in SDN.

The graph plotted between true positive rate and false positive rate is called AU-ROC, the curve defines more area under curve efficient the performance of the framework. The ROC curve for hybrid framework is depicted in Fig.10 shows the efficient performance of our proposed technique.

Testing time of proposed system is also calculated and shown in Fig.11. The time consumption for proposed framework is 296 milliseconds. It can be clearly seen that time efficiency of hybrid (LSTM, DNN) is 275. There is a trivial trade-off between proposed hybrid deep learning model (LSTM, CNN) and (LSTM, DNN).

V. CONCLUSION

The hybrid Cuda-enabled deep learning-based intrusion detection system has strong ability to detect varied sophisticated threats and attacks. Further, the control plane-based orchestration is highly flexible, innovative and cost effective that can be extended to any underlying network and customized to any commercial SDN controller. This framework shows high accuracy for detection of benign and evolving cyber threats and attacks. The performance of our model obtains performance with 98.6% detection accuracy. As part of our future work, we plan to implement various other deep learning models to efficiently and timely detect evolving cyber threats. We also plan to orchestrate the control plane for various other promising functions such as timely prevention and mitigation of the underlying large-scale distributed systems and IoT.

REFERENCES

[1] R. Amin, M. Reisslein, and N. Shah, "Hybrid SDN networks: A survey of existing approaches," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3259–3306, 4th Quart., 2018.

[2] A. Akhunzada and M. K. Khan, "Toward secure software defined vehicular networks: Taxonomy, requirements, and open issues," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 110–118, Jul. 2017.

[3] A. Carrega, M. Repetto, F. Risso, S. Covaci, A. Zafeiropoulos, T. Giannetsos, and O. Toscano, "Situational awareness in virtual networks: The ASTRID approach," in *Proc. IEEE 7th Int. Conf. Cloud Netw. (CloudNet)*, Tokyo, Japan, Oct. 2018, pp. 1–6.

[4] N. Banerjee, T. Giannetsos, E. Panaousis, and C. C. Took, "Unsupervised learning for trustworthy IoT," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Jul. 2018, pp. 1–8.

[5] A. Shaghaghi, M. A. Kaafar, R. Buyya, and S. Jha, "Software-defined network (SDN) data plane security: Issues, solutions, and future directions," in *Handbook of Computer Networks and Cyber Security*. Springer, 2020, pp. 341–387.

[6] H. Zhang, Z. Cai, Q. Liu, Q. Xiao, Y. Li, and C. F. Cheang, "A survey on security-aware measurement in SDN," *Secur. Commun. Netw.*, vol. 2018, pp. 1–14, Oct. 2018.

[7] R. Swami, M. Dave, and V. Ranga, "Software-defined networking-based DDoS defense mechanisms," *ACM Comput. Surv.*, vol. 52, no. 2, pp. 1–36, May 2019.

[8] S. Haider, A. Akhunzada, I. Mustafa, T. B. Patel, A. Fernandez, K.-K. R. Choo, and J. Iqbal, "A deep cnn ensemble framework for efficient ddos attack detection in software defined networks," *IEEE Access*, vol. 8, pp. 53972–53983, 2020.

[9] M. A. Ferrag, L. Maglaras, H. Janicke, and R. Smith, "Deep learning techniques for cyber security intrusion detection : A detailed analysis," in *Proc. SCADA*, Sep. 2019, pp. 126–136.

[10] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, p. 20, Dec. 2019.

[11] N. Shone, T. Nguyen Ngoc, V. Dinh Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 41–50, Feb. 2018.

[12] S. Potluri and C. Diedrich, "Accelerated deep neural networks for enhanced intrusion detection system," in *Proc. IEEE 21st Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2016, pp. 1–8.

[13] A. Akhunzada, A. Gani, N. B. Anuar, A. Abdelaziz, M. K. Khan, A. Hayat, and S. U. Khan, "Secure and dependable software defined networks," *J. Netw. Comput. Appl.*, vol. 61, pp. 199–221, 2016.

[14] M. Abadi, U. Erlingsson, I. Goodfellow, H. B. Mcmahan, I. Mironov, N. Papernot, K. Talwar, and L. Zhang, "On the protection of private information in machine learning systems: Two recent approches," in *Proc. IEEE 30th Comput. Secur. Found. Symp. (CSF)*, Aug. 2017, pp. 1–6.

[15] M. Rahman Minar and J. Naher, "Recent advances in deep learning: An overview," 2018, *arXiv:1807.08169*. [Online]. Available: http://arxiv.org/abs/1807.08169

[16] A. Akhunzada, E. Ahmed, A. Gani, M. K. Khan, M. Imran, and S. Guizani, "Securing software defined networks: Taxonomy, requirements, and open issues," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 36–44, Apr. 2015.

[17] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis, and G. Loukas, "A taxonomy and survey of attacks against machine learning," *Comput. Sci. Rev.*, vol. 34, Nov. 2019, Art. no. 100199.

[18] J. Cui, M. Wang, Y. Luo, and H. Zhong, "DDoS detection and defense mechanism based on cognitive-inspired computing in SDN," *Future Gener. Comput. Syst.*, vol. 97, pp. 275–283, Aug. 2019.

[19] S. Garg, K. Kaur, N. Kumar, and J. J. P. C. Rodrigues, "Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in SDN: A social multimedia perspective," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 566–578, Mar. 2019.

[20] T. A. Tang, D. McLernon, L. Mhamdi, S. A. R. Zaidi, and M. Ghogho, "Intrusion detection in sdn-based networks: Deep recurrent neural network approach," in *Deep Learning Applications for Cyber Security*. Springer, 2019, pp. 175–195.

[21] S. Maeda, A. Kanai, S. Tanimoto, T. Hatashima, and K. Ohkubo, "A botnet detection method on SDN using deep learning," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–6.

[22] C. Birkinshaw, E. Rouka, and V. G. Vassilakis, "Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks," *J. Netw. Comput. Appl.*, vol. 136, pp. 71–85, Jun. 2019.

[23] C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang, X. Li, and L. Gong, "Detection and defense of DDoS attack-based on deep learning in OpenFlow-based SDN," *Int. J. Commun. Syst.*, vol. 31, no. 5, p. e3497, Mar. 2018.

[24] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep recurrent neural network for intrusion detection in SDN-based networks," in *Proc. 4th IEEE Conf. Netw. Softwarization Workshops (NetSoft)*, Jun. 2018, pp. 202–206.

[25] C.-H. Huang, T.-H. Lee, L.-H. Chang, J.-R. Lin, and G. Horng, "Adversarial attacks on SDN-based deep learning ids system," in *Proc. Int. Conf. Mobile Wireless Technol.* Springer, 2018, pp. 181–191.

[26] R. M. A. Ujjan, Z. Pervez, and K. Dahal, "Suspicious traffic detection in SDN with collaborative techniques of snort and deep neural networks," in *Proc. IEEE 20th Int. Conf. High Perform. Comput. Commun.*, Jun. 2018, pp. 915–920.

[27] M. Latah and L. Toker, "Towards an efficient anomaly-based intrusion detection for software-defined networks," *IET Netw.*, vol. 7, no. 6, pp. 453–459, Nov. 2018.

[28] S. K. Dey and M. M. Rahman, "Flow based anomaly detection in software defined networking: A deep learning approach with feature selection method," in *Proc. 4th Int. Conf. Electr. Eng. Inf. Commun. Technol. (iCEE-iCT)*, Sep. 2018, pp. 630–635.

[29] I. Letteri, M. Del Rosso, P. Caianiello, and D. Cassioli, "Performance of botnet detection by neural networks in software-defined networks," in *Proc. ITASEC*, 2018, pp. 1–8.

[30] 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.

[31] Q. Niyaz, W. Sun, and A. Y Javaid, "A deep learning based DDoS detection system in software-defined networking (SDN)," 2016, *arXiv:1611.07400*. [Online]. Available: http://arxiv.org/abs/1611.07400

[32] Y. Cui, L. Yan, S. Li, H. Xing, W. Pan, J. Zhu, and X. Zheng, "SD-Anti-DDoS: Fast and efficient DDoS defense in software-defined networks," *J. Netw. Comput. Appl.*, vol. 68, pp. 65–79, Jun. 2016.

[33] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, 2018, pp. 108–116.

[34] W. Xia, W. Zhu, B. Liao, M. Chen, L. Cai, and L. Huang, "Novel architecture for long short-term memory used in question classification," *Neurocomputing*, vol. 299, pp. 20–31, 2018.

[35] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," 2018, *arXiv:1808.03314*. [Online]. Available: http://arxiv.org/abs/1808.03314

[36] M. Abadi, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th Symp. Oper. Syst. Des. Implement.*, 2016, pp. 265–283.

[37] A. Gulli and S. Pal, *Deep Learning With Keras*. Birmingham, U.K.: Packt, 2017.

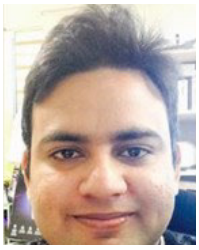
[38] A. Tharwat, "Classification assessment methods," *Appl. Comput. Informat.*, Aug. 2018.

JAHANZAIB MALIK received the B.S. degree (Hons.) in software engineering from the National University of Modern Language Islamabad, Pakistan, and the M.Sc. degree in information security from COMSATS University Islamabad, Islamabad. He is currently a Researcher with the National Cyber Security Auditing and Evaluation Laboratory (NCSAEL), National University of Science and Technology (NUST), Islamabad. His research interests include, but are not limited to software defined networking, smart devices security, threat detection and intelligence, malware analysis and detection, application of deep learning and machine learning in cyber defence, distributed computing, and big data.

ADNAN AKHUNZADA was an Enthusiastic and a Dedicated Professional with extensive 12 years of research and development experience in ICT industry and academia. His experience as an Educator and a Researcher are diverse. He was a Lecturer, a Senior Lecturer, a year Tutor, an Occasional Lecturer (other engineering departments), and an Assistant Professor with COMSATS University Islamabad (CUI). He was a Senior Researcher with the RISE SICS Vasteras, Sweden, and a Research Fellow and a Scientific Lead with the DTU Compute, Technical University of Denmark (DTU). He was also a Visiting Professor having mentorship of graduate students and supervision of academic and research and development projects at UG and PG levels. He has demonstrated history and a proven track record of high-impact published research, such as patents, journals, transactions, commercial products, book chapters, reputable magazines, conferences, and conference proceedings. He was involved with the international accreditation, such as the Accreditation Board for Engineering and Technology (ABET) and the Curriculum Development according to the guidelines of ACM/IEEE. He is involved with various EU and the Swedish funded projects of cyber security. His main research interests include cyber security, machine learning, deep learning, reinforcement learning, artificial intelligence, blockchain and data mining, information systems, large-scale distributed systems, such as edge, fog and cloud, and SDNs, the IoT, industry 4.0, and the Internet of Everything (IoE). He is a member of the technical program committee of varied reputable conferences and editorial boards. He serves as an Associate Editor for IEEE Access.

IRAM BIBI received the B.S. degree (Hons.) in software engineering from the National University of Modern Languages (NUML) Islamabad, Pakistan, in 2017, and the Master of Science degree in information security from Comsats University Islamabad, Islamabad, in 2020. She is currently with ProSanct, for a period of one year, as a Research Assistant. Her research interests include, but not limited to analysis and detection of network based on cyber threat and attacks for android, the Internet of Things, and software defined networking.

MUHAMMAD IMRAN (Member, IEEE) graduated in software engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2006. He received the master's degree in software engineering and the Ph.D. degree in computer science from the University of Southampton, U.K., in 2009 and 2015, respectively. He was a Lecturer with CIIT, from 2007 to 2008, Islamabad, Pakistan. He is currently an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad (CUI), Islamabad. His research interests include social network analysis, artificial intelligence, and semantic web. He received the Faculty Development Scholarship from CIIT.



ARSLAN MUSADDIQ received the B.S. degree in electrical engineering (telecommunication) from Bahria University, Islamabad, Pakistan, in 2011, and the M.S. degree in communication and network engineering from University Putra Malaysia, in 2015. He is currently pursuing the Ph.D. degree with the Department of Information and Communication Engineering, College of Engineering, Yeungnam University, Gyeongsan, South Korea. His research interests include wireless networking, the Internet of Things, wireless resource management, routing protocols, and ad hoc networks. He was a recipient of the Outstanding Dissertation (M.S. Level) Award from the IEEE Malaysia Communication Society and the Vehicular Technology Society Joint Chapter, in 2015.

SUNG WON KIM received the B.S. and M.S. degrees from the Department of Control and Instrumentation Engineering, Seoul National University, South Korea, in 1990 and 1992, respectively, and the Ph.D. degree from the School of Electrical Engineering and Computer Sciences, Seoul National University, in 2002. From 1992 to 2001, he was a Researcher with the Research and Development Center, LG Electronics, South Korea. From 2001 to 2003, he was a Researcher with the Research and Development Center, AL Tech, South Korea. From 2003 to 2005, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. In 2005, he joined the Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, South Korea, where he is currently a Professor. His research interests include resource management, wireless networks, mobile networks, performance evaluation, and embedded systems.