Multilevel Deep Neural Network Approach for Enhanced Distributed Denial-of-Service Attack Detection and Classification in Software-Defined Internet of Things Networks

Abid, Yawar Abbas; Wu, Jinsong; Xu, Guangquan; Fu, Shihui; Waqas, Muhammad

DOI
10.1109/JIOT.2024.3376578

Publication date
2024

Document Version
Final published version

Published in
IEEE Internet of Things Journal

Citation (APA)
Abid, Y. A., Wu, J., Xu, G., Fu, S., & Waqas, M. (2024). Multilevel Deep Neural Network Approach for Enhanced Distributed Denial-of-Service Attack Detection and Classification in Software-Defined Internet of Things Networks. *IEEE Internet of Things Journal*, *11*(14), 24715-24725. https://doi.org/10.1109/JIOT.2024.3376578

Multilevel Deep Neural Network Approach for Enhanced Distributed Denial-of-Service Attack Detection and Classification in Software-Defined Internet of Things Networks

Yawar Abbas Abid, Jinsong Wu, *Senior Member, IEEE*, Guangquan Xu, *Member, IEEE*,
Shihui Fu, and Muhammad Waqas

***Abstract*—With the increasing rates of interconnected Internet of Things (IoT) devices within software-defined networking (SDN) environments, Distributed Denial-of-Service (DDoS) attacks have become increasingly common. As a result of this challenge, novel detection and classification methods must be developed based on the unique characteristics of SDN-supported IoT networks. This article proposes a novel approach to detecting and categorizing DDoS attacks that have been optimized specifically for such environments. As part of our methodology, we integrate convolutional neural networks (CNNs) and long-short-term memory (LSTM) models into a multilevel deep neural network architecture. With this hybrid architecture, complex spatial and temporal patterns can be automatically extracted from raw network traffic data to facilitate comprehensive analysis and accurate identification of DDoS attacks. We validate the efficacy and superiority of our proposed approach over traditional machine learning algorithms by conducting rigorous experiments on real-world data sets. Our findings underscore the potential of the multilevel deep neural network approach as a robust and scalable solution for mitigating DDoS attacks in SDN-supported IoT networks. By improving network security and resilience to evolving threats, our methodology contributes to safeguarding critical infrastructures in the era of interconnected IoT ecosystems.**

***Index Terms*—CIC-DDoS 2019, classification, convolutional neural network (CNN), Distributed Denial of Service (DDoS), long-short-term memory (LSTM), recurrent neural network (RNN), software-defined networking (SDN).**

I. Introduction

THE DISTINCT nature of Internet of Things (IoT) networks, characterized by heterogeneous nodes and devices limited by resources, requires a unique security approach that differentiates them from traditional networks. Given the varied vulnerabilities in IoT networks, various types of Distributed Denial-of-Service (DDoS) attacks can be executed using different methods, with varying impacts [1], [2]. With the substantial growth of the Internet and its excessive exploitation as a commercial platform, numerous network attacks have emerged [2]. One of the most common attacks is DDoS [3]. These attacks have targeted major websites, such as Amazon, eBay, and Yahoo, attracting much public attention, as reported in [4]. These attacks affect network services by the excessive number of requests, which causes rejection for an actual network user. If multiple systems are used to launch such attacks, it is known as a DDoS attack [5]. Detecting and managing these attacks can be a challenging task [6].

Very innovative machine learning (ML) technologies, such as deep learning that integrates software-defined networking (SDN) and sophisticated deep learning algorithms, are a persistently used security innovation [6]. SDN consists of the current generation of networks, which makes them more competent than previous networks, as it ensures various simplifications in the task of network management and troubleshooting and, consequently, takes an essential position to be implemented. Blending the implications of convolutional neural networks (CNNs) with RL-based techniques and other ML strategies in SDNs infrastructure can be purposefully utilized to administer real devices. The addition of these two strategies can enhance the detection efficiency of DDoS attacks to ensure the security of IoT networks [7] Whereas, intelligent deep learning methods would also improve the safety of SDN-IoT networks by giving them strong defenses against DDoS threats. In this sense, a network made up of IoT objects includes a

Manuscript received 29 October 2023; revised 18 February 2024; accepted 26 February 2024. Date of publication 12 March 2024; date of current version 9 July 2024. This work was supported in part by the China Guangxi Science and Technology Plan Project (Guangxi Science and Technology Base and Talent Special Project) under Grant AD23026096 and Grant 2022AC20001; in part by Chile CONICYT FONDECYT Regular under Grant 1181809; and in part by Chile CONICYT FONDEF under Grant ID16I10466. *(Corresponding authors: Jinsong Wu; Guangquan Xu.)*

Yawar Abbas Abid is with the School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China, and also with the Department of Computers Science, COMSATS University Islamabad, Islamabad 57000, Pakistan (e-mail: yawar.ravian@gmail.com).

Jinsong Wu is with the School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin 510004, China, and also with the Department of Electrical Engineering, University of Chile, Santiago 8370451, Chile (e-mail: wujs@ieee.org).

Guangquan Xu is with the Tianjin Key Laboratory of Advanced Networking, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China (e-mail: losin@tju.edu.cn).

Shihui Fu is with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: shihui.fu@tudelft.nl).

Muhammad Waqas is with the FAST School of Computing, National University of Computer and Emerging Science, Karachi 75050, Pakistan (e-mail: waqas.sheikh@nu.edu.pk).

Digital Object Identifier 10.1109/JIOT.2024.3376578

variety of physical objects, including sensors, communication components, software, and other technologies to provide the connection for the exchange and sharing/processing of data among users, the cloud, and systems over the Internet [8].

SDN is a streamlined concept of network systems that fulfills the unfulfilled desires of obsolete hardware-based network systems. SDN builds a flexible, soft, and programmable framework that covers packets and dataflow, applications, and control planes [9]. The control plane in the SDN gives access to the data plane for configuration and management, which is the place where routers, switches, modern devices, or any other equipment such as that is placed. The data plane devices get the information from the control plane where the controller programs the rules that are responsible for a packet to be transmitted. The controller links the processing units of the application alongside the networks through which they are transmitted. In SDN, these planes are exploited to break the plug where the decision making and forwarding functions are separated from each other.

Despite the many advantages of SDN, there are drawbacks, including scalability, dependability, and security issues. One of the security risks is its susceptibility to DDoS attacks, as it uses a centralized controller. When a switch receives a packet from an unmatched Internet protocol (IP) (from the flow table), it is automatically sent to the controller. The controller then sends a flow rule for this IP to the device, usually a switch. If an attacker uses the default behavior and directs most packets from various IP addresses, these packets will be transmitted to the controller. This traffic will occupy all the controller's resources, making it impossible for legitimate users to access the system. These issues make SDNs vulnerable to DDoS attacks. However, many ML algorithms have been proposed to secure SDN networks. These ML algorithms provide more efficient, intelligent, and dynamic solutions for the optimization, management, and security of SDN networks [10]. The elucidating cybersecurity-promulgated malware taxonomy (ECMT) framework is widely used to enhance cybersecurity in IoT environments, preventing intrusion, protecting information, deterring cybercrime, and reducing energy consumption. Using established criteria or thresholds applied to aspects of network traffic is one method to detect DDoS attacks [11], [12], [13]. These rules help to spot unusual activity that can point to an attack, including sending out an alert when the volume of incoming traffic from a certain IP address exceeds a predetermined threshold in a short amount of time. These are modest to understand and implement, but may not be able to detect complex or dynamic DDoS attack patterns. Furthermore, the high number of false positives or false negatives produced by these technologies can be challenging regardless of whether there are valid traffic changes or complicated assault tactics meant to avoid detection.

Several studies on DDoS detection have been carried out utilizing custom characteristics derived from network traffic data and conventional ML techniques [14], [15], [16], [17]. These attributes consist mostly of payload properties, statistical data obtained from network flows, or information from packet headers. Although these methods can yield respectable results in some situations, they are not always suitable for capturing the intricate patterns and dependencies in network traffic data. Furthermore, the quality and applicability of the features created manually that may not always accurately reflect the subtleties of DDoS assault behavior have a significant impact on how effective these algorithms are.

CNNs and shallow neural networks [18], [19], [20], [21], deep learning algorithms have shown potential to identify spatial patterns, but they have difficulty handling encrypted communication, evasion tactics, and dynamic traffic that attackers use in distributed DDoS operations. Consequently, to increase detection accuracy and decrease false positives for a reliable DDoS detection method, this research suggests combining long-short-term memory (LSTM) and CNN techniques. The main contribution of this research is to examine an accurate and effective DDoS detection method in SDN that can recognize attacks instantly.

The LSTM and CNNs are combined in the LSTM-CNN hybrid technique, which takes advantage of CNN's superior spatial pattern recognition skills and LSTMs' temporal analytic capabilities. The model can now automatically identify relevant elements from unprocessed network traffic data, capturing the intricate temporal and spatial correlations typical of DDoS attacks. Compared to conventional methodologies, the hybrid LSTM-CNN strategy is resistant to changing attack tactics. It is capable of processing massive amounts of data quickly and effectively. Additionally, by continuously learning from the incoming data and utilizing deep learning techniques, the MLDNN methodology reduces false positives and improves overall detection accuracy. The following are the main contributions of this article.

1) This article presents a technique for detecting DDoS attacks and classifying them according to different protocols. The technique uses a combination of deep CNNs and LSTM models, which provide temporal and spatial analysis capabilities, thus strengthening the effectiveness of the detection.
2) The hybrid approach LSTM-CNN is proposed to automatically learn features from raw network traffic, adapt in real time, and effectively identify attack vectors, reducing false alarms and minimizing disruptions to network operations.

The suggested approach has a number of benefits. For example, the hybrid method provides real-time detection capabilities that minimize the impact of DDoS attacks on network infrastructure and services. Additionally, the proposed approach does not rely on a single type of information, making it an exceptional tool for detecting DDoS attacks in ever-changing network environments. Its adaptability helps it learn from incoming data and adjust to subtle changes in traffic patterns.

The remaining article is organized as follows. Section II discusses the latest relevant research works in the chosen area. Section III describes the overall proposed approach of the applied methods with the model flow structure and parametric values. Section V provides the relevant experimental results and discussions. Section VI concludes this article.

II. Related Works

The identification and categorization of DDoS attacks within SDNs have become crucial research areas due to the rising threat levels. Recent studies have significantly contributed by introducing innovative methods and approaches. This section offers a concise overview of DDoS attack detection and classification in SDNs, utilizing diverse strategies. These methods utilize traffic attributes within SDNs to effectively detect DDoS attacks, improving accuracy by analyzing traffic patterns and distinguishing between normal and malicious activity. Additionally, employing deep learning techniques for DDoS attack detection and classification in SDNs has proven highly effective, accurately identifying various attack types. Notably, one study [12] focused on utilizing machine learning (ML) techniques to detect anomalies associated with DDoS attacks in SDNs, thereby enhancing detection capabilities by recognizing abnormal network behavior. This approach integrates flow-based, protocol-based, and behavioral-based detection methods to precisely identify and mitigate DDoS attacks within SDNs.

Through research [16], it has been shown that deep learning models are efficient. This approach improves detection accuracy by utilizing SDN's dynamic network control capabilities and machine learning algorithms. Deep learning can correctly classify different categories of DDoS attacks. The study [22] examined how ML techniques can be adapted for DDoS attack anomaly detection. Spotting out the SDN that will be most helpful to increasing the detection capabilities of SDNs. Unconventional network activities that may be Denial-of-Service (DoS) attacks. DDoS attacks can be detected and mitigated quickly and accurately through SDNs.

The DDoS attack detection methods via SDN extends detection capabilities by identifying abnormal network activities related to DDoS attacks. First, to correctly identify and contain DDoS attacks in SDNs, Abdulqadder et al. [17] proposed a multilayered detection architecture that utilizes flow-based detection, protocol-based detection, and behavior-based detection. For the sake of enhancement of detection accuracy, the research [21] created a hybrid attack detection framework based on ML algorithms and SDN's dynamic network control features. Additionally, on the other hand, the reinforcement learning which [23] illustrates how is very capable of making the quick adaptations to the ever-changing attack circumstances and making the timely mitigation decisions. And, it provided a very deep reinforcement learning method to detect and decrease DDoS attacks in the SDN nets. A DDoS-resistant SDN architecture has been proposed by [24] that combines the random forest (RF) and also flow entropy algorithms. The method uses flow-level information to derive the entropy and furthermore utilizes an RF classifier [25]. A deep-learning-based scheme for the DDoS attacks detection in SDNs was adopted by us with the aim of achieving a very high detection accuracy rate. This route employs a deep learning model that is very powerful and utilizes the network traffic data and then distinguishes between the normal traffic and an attack.

Although a DDoS attack detection approach employing traffic behavior analysis that was reported in [13] is already available, more research is required to make networks like SDNs more resilient to DDoS attacks. ML algorithms along with traffic behavior pattern monitoring are employed to detect irregular traffic immediately which is the flagging of DDoS attacks through the software-defined network approach contributed to leveraging SDN to detect DDoS attacks by suggesting a software-defined network method that uses traffic features, to analyze traffic and identify features that are specifically related to DDoS attacks and thereby resulting in highly advanced detection. Since it can detect these pollutants and mitigate them adequately and timely. Bahashwan et al. [26] proposed a software-defined network-based DDoS attack detection approach using traffic characteristics, the proposed model achieved an accuracy of 94% in accurately identifying and classifying DDoS attacks. Alanazi et al. [27] presented a hybrid deep learning model along with its ability to take feature selection into consideration; their ultimate goal was to improve detection accuracy and yet reduce computational cost. Along the same lines, Jiang et al. [28] discussed the idea of multidimensional network traffic detection (MNTD). They proposed an algorithm with a multigranularity level for detecting abnormal network traffic which uses the multi-instance learning technique put forward by Waqas et al. [29], [30], [31]. One key benefit is the deployment of this way when it is hard to annotate in an appropriate manner and saves the need for supervision.

Table I provides an overview of earlier studies on the use of a DL model for attack detection. The deep learning model, application domain, and data set used in these works are compared. A summary of the evaluation findings of each study is also provided, and Table II offers a comparative analysis of various assault detection techniques. The features, benefits, and downsides of each approach are the basis for this comparison.

III. Methodology

The proposed approach consists of several stages. In the first stage, we preprocess the data to facilitate smooth training. In this stage, we clean the traffic data by filtering, normalizing, and removing any noise or outliers to ensure consistency. We also convert the data into a suitable format for training and testing the models. In the second stage, the training process is initiated. The training process of the proposed approach is a two-stage learning process. In the first stage, we train the CNN and recurrent neural network (RNN) modules of the network. These two modules are trained separately to minimize as much loss as possible. After that, the learned features of the models are combined to obtain better performance. In this section, we first present a brief overview of CNN and LSTM, followed by the hybrid learning approach approach as shown in Figure 1.

A. Data Preprocessing

Socket Features Removal: The basic socket features, such as IP addresses of the source and the destination, the ports of the source and the destination, flow ID, and the timestamp, will be removed. Since these features differ for each network, packet characterization is required to train the model directly.

TABLE I
OVERVIEWS SEVERAL STUDIES ON USING DL MODELS IN DDOS ATTACK DETECTION

Study	Method Used	Data set	Description
[32]	CNN, LSTM	CICDDoS2019 [33]	An IIoT-based CNN-LSTM-based model to identify and categorize DDoS attacks.
[34]	A signature-based attack detection approach (LSTM), DL	CICDDoS2019	A DDoS attack detection and prevention system for public cloud networks using an LSTM-CLOUD architecture
[35]	AE,MLP	CICDDoS2019	AE-MLP is a hybrid method for identifying and categorizing DDoS attacks.
[36]	RF, LGBM, XGBoost and Ada Boos	CICDDoS2019[33], Slowloris[37]	A machine learning-based DDoS attack detection system for D2D communications
[38]	IG, RF, DL,LSTM and Autoencoder	InSDN[39]CICIDS2017, CICIDS2018	IG and RF feature selection techniques in SDNs are the foundation of this DDoS detection strategy.
[40]	DNN	CICIDS2017[41]	Real-time SDN DDoS assault detection.
[42]	FS-WOA-DNN	CICIDS2017	Method for detecting DDoS attacks on cloud storage services
[43]	Information entropy detection method, DL and CNN	CICIDS2017	A dual-tier DDoS assault identification technique grounded in information entropy and deep learning
[44]	CNN, ROA	KDD cup 99[45]	IU-ROA model for DDoS attack detection and mitigation
[46]	A threshold tuning method and feature-based classification method	Mirai [47], SYN-flooding LowRate [48]	The SAFE system's ID for DDoS attack flows
[49]	RNN, AE	CICDDoS2019	RNN and AE can be combined to improve accuracy.
[50]	LSTM, DL, binary cross-entropy, a Mini-batch gradient descent (GD) algorithm	Hogailla [51]	A Fog network defense strategy against DDoS attacks using DL.

TABLE II
ADVANTAGES AND DISADVANTAGES OF VARIOUS ASSAULT DETECTION STRATEGIES

Approach	Description	Advantages	Disadvantages
Statistics-based	It processes data using advanced statistical techniques and examines network traffic.	-Easy yet lacking in precision.	-Needs a deep understanding of statistics.
Pattern-based	It looks for patterns, forms, and other features in the data.	- Simple to put into practice.	- For identification, one may utilize a hash function.
Rule-based	It uses an attack ”signature” to identify possible attacks on suspicious network traffic.	-Rules necessitate pattern matching. Therefore, rule-based systems may incur significant computing expenses.	-A vast number of regulations are needed to identify every possible risk. -Minimal FPR. -High detection rate.
State-based	It looks at a series of events to look for any potential intrusion.	-Probabilistic and autodidactic.	-A low rate of false positives.
Heuristic-based	identifies any unusual or aberrant behaviour.	-Reduces the number of false negatives and positives. -High precision in detecting.	-The detection thresholds determine the accuracy. -High cost of calculation.
ML-based approach	Machine learning models use rules, procedures, or ”transfer functions” to identify patterns and predict behaviours.	-Swiftly recognizes trends in traffic. -Excellent precision in identifying.	-Issue with feature engineering. -Lengthy training period. -Improved accuracy requires a larger dataset.
DL-based approach	Feature extraction and classification modules use supervised and unsupervised learning.	-Capacity to acquire features with high dimensions. -Adapting flexibly to new issues. -Superior capacity for learning layer features. -The capacity to handle raw data directly.	-Problem with generalization. -Leads to overfitting occasionally. -Computational overhead. -Improved accuracy requires a larger dataset.

In addition to that, the regular user and the intruder can both have the same IP address. Hence, an overfitting problem can arise if the model is trained based on socket information as this information can make the model biased.

1) *Data Cleaning:* A huge number of missing (NaN) values are present in the data set, which are necessarily removed from the data set.
2) *Input Data Normalization:* The features in the data set have numerous numerical values. Classification errors may occur while training the model on original data and it will also be time consuming to train the model. Therefore, it is appropriate to normalize the data features. There are many options available for normalizing the data such as min–max and z-score normalization, the minimum value assigned to any feature is 0 whereas the maximum value is 1 for a feature having maximum values.
3) *Labeled Data Encoding:* The proposed model is trained using binary classification where each traffic input is classified either as malicious or normal. The classes of DDoS are placed into the attack category. Finally, the string value will be encoded to binary values (0 and 1); 0 for the normal label and 1 for the attack label.

B. Working of CNN

CNNs are commonly used in image recognition and computer vision tasks, such as image classification, anomaly detection, object detection, and many others [52], [53], [54]. The key operations in a CNN are convolution, pooling, and fully connected layers, as defined in Algorithm 1. Here are the resultant equations (1) and (2) for these operations.

Convolution is the core operation in CNNs, where a filter is applied to an input image to extract features. The output feature map is obtained by convolving the filter with the input image, element-wise multiplication, and summation.

Resultant Equation (1):

$$V[c,d] = \sum k_1 \sum k_2 X[c+k_1,\ d+k_2] \cdot F[k_1,k_2] + b_i \quad (1)$$

Algorithm 1 Working of CNN

```
Let $X$ be the input dataset of network traffic data.
Split dataset $X$ into training $X_{train}$ and testing $X_{test}$.
Train a CNN model on the training set $X_{train}$ to learn feature representations from the network traffic data.
Define the architecture of the CNN model.
Define the weights and biases of the CNN layers as $W_c$ and $b_c$ respectively.
Perform forward propagation:
Convolution: Compute the convolution operation of the input $X_{train}$ with the weights $W_c$ and biases $b_c$.
Activation: Apply a nonlinear ReLU activation function into the convolved outputs.
Pooling: Perform pooling operations (e.g., max pooling) to downsample the feature maps.
Calculate the loss function $L_c$ (cross-entropy) among the predicted output and the true labels.
Update CNN weights and biases $W_c$ and $b_c$ using back-propagation and gradient descent.
Repeat the training process for multiple epochs until convergence.
```

Algorithm 2 Working of LSTM

```
Preprocess the network traffic data into sequences, considering a specific time window or time steps.
Let $S$ represent the sequences of network traffic data.
Train an LSTM model on the training sequences S to capture temporal dependencies and patterns.
Define the architecture of the LSTM model, including LSTM layers and possibly additional layers like dropout or batch normalization.
Define the weights and biases of the LSTM layers as WL and bL respectively.
Perform forward propagation:
LSTM Cell Equations:
    a) $\mathrm{V_t} = \sigma(\mathrm{W_1}[\mathrm{h_t} - 1, \mathrm{x_t}] + \mathrm{b_1})$
    b) $\mathrm{I_t} = \sigma(\mathrm{W_2}[\mathrm{h_t} - 1\mathrm{x_t}] + \mathrm{b_2})$
    c) $\mathrm{C_{t'}} = \tanh\big(\mathrm{W_3} \cdot \big[\mathrm{h_{t-1}}, \mathrm{x_t}\big] + \mathrm{b_3}\big)$
    d) $\mathrm{Cs} = \mathrm{fs} \odot \mathrm{C_s} - 1 + \mathrm{I_s} \odot \mathrm{C_s}'$
    e) $\mathrm{O_{s_s}} = \sigma(\mathrm{W_4} \cdot [\mathrm{h_t} - 1\mathrm{X_s}] + \mathrm{b_4})$
    f) $\mathrm{hs} = \mathrm{O_s} \odot \tanh(\mathrm{C_s})$
Calculate the loss function $L_l$ and update the LSTM weights and biases $W_L$ and $b_L$ using back propagation and gradient descent.
Repeat the training process for multiple epochs.
```

where $V[c, d]$ is the value of the output feature map at (c, d), X indicates the input value, F is the filter/kernel, and b_i is the bias term. Pooling is used to decrease the spatial sizes of the feature maps, reducing computational complexity and extracting dominant features.

Resultant Equation (2):

$$V[c, d] = \max_m \max_n X[c.s + k_1,\ d.s + k_2] \tag{2}$$

where $V[c, d]$ is the value of the output feature map at (c, d), X is the input feature map, and s indicates stride, which determines the step size for the pooling window. CNNs are used with fully connected layers to categorize or predict the output based on the extracted information. Each neuron in a layer that is fully connected is linked to every neuron in the layer above it, enabling extensive information exchange across the network.

Resultant Equation (3):

$$OV = \text{ fun } (W \cdot X + b_i) \tag{3}$$

where OV indicates the output vector, *fun* is an activation function, W is the weight matrix connecting the previous layer to this layer, X is the input vector from the previous layer, and *bi* is the bias term. Note that the activation function fun can vary based on the specific task and network architecture; commonly used activation functions include ReLU, sigmoid, and tanh. These equations represent the basic operations in a CNN. However, the architecture and variations in CNNs can be more complex, incorporating additional layers, skip connections, batch normalization, and other techniques to improve performance.

C. Working of LSTM

To use LSTM for DDoS attack detection, the model can be trained on time-series data of network traffic features. Here are the resultant equations for LSTM in the context of DDoS attack detection as given in Algorithm 2. The input to the LSTM model consists of a sequence of network traffic features at each time step. The input features can be represented as a vector, such as packet count, packet size, source IP address, destination IP address, and so on. The LSTM cell equations describe the computations performed at each time step within the LSTM unit. In the context of DDoS attack detection, the LSTM cell (4)–(9) can be written as follows.

1) *Forget Gate:*

$$V_t = \sigma\big(W_1\big[h_{t-1}x_t\big] + b_1\big). \tag{4}$$

2) *Input Gate:*

$$I_t = \sigma\big(W_2\big[h_{t-1}x_t\big] + b_2\big) \tag{5}$$

$$C_{t'} = \tanh\big(W_3 \cdot \big[h_{t-1}, x_t\big] + b_3\big). \tag{6}$$

3) *Update Cell State:*

$$C_s = f_s \odot C_s - 1 + I_s \odot C_s{}'. \tag{7}$$

4) *Output Gate:*

$$O_s = \sigma\big(W_4 \cdot \big[h_t - 1, X_{ss}\big] + b_4\big) \tag{8}$$

$$h_s = O_s \odot \tanh(C_s) \tag{9}$$

where X_s represents the input feature vector, $\mathrm{h_{t-1}}$ is the hidden state, W_1, W_2, W_3, and W_4 are weight matrices, b_1, b_2, b_3, and b_4 are bias terms, σ represents the sigmoid function, tanh indicates the hyperbolic tangent activation function, and $\odot$ indicates element wise multiplication.

After processing the entire sequence, the final hidden state h_t of the last time step can be fed into a classification layer to predict the DDoS attack class. The classification layer can be a fully connected layer followed by a softmax activation function to obtain class probabilities. Each sample in the

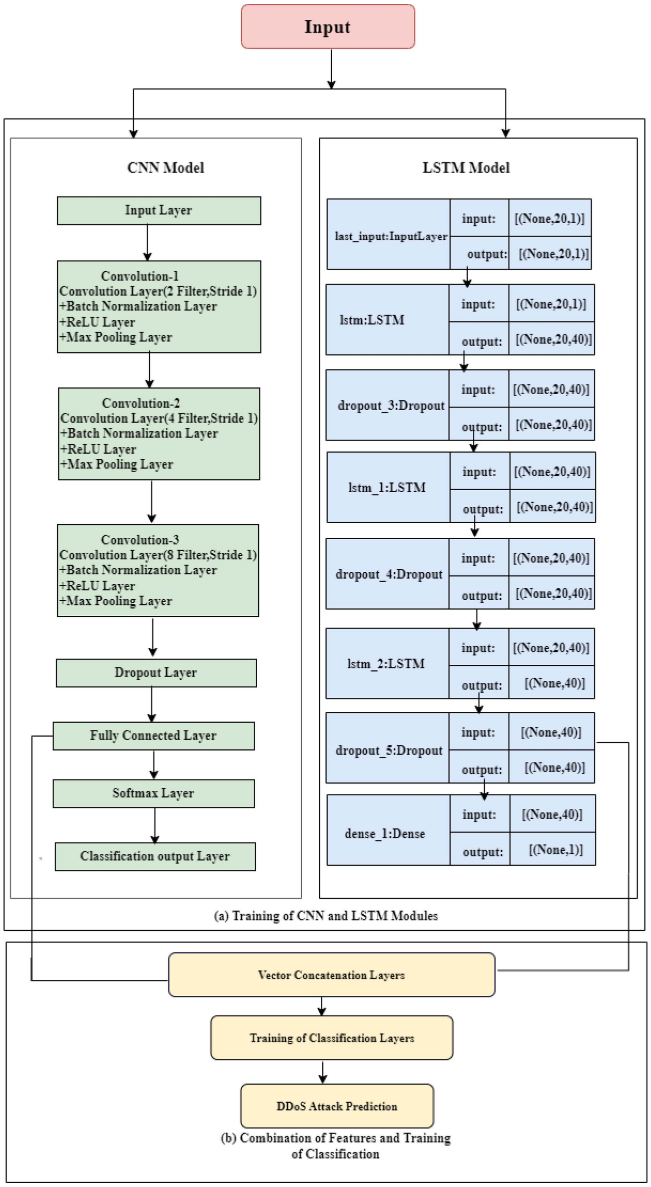

Fig. 1. Structure of the proposed MLDNN framework. (a) shows the Stage 1 training process for CNN and LSTM, while (b) illustrates the combination of feature vectors and training of classification layers.

time-series data has a matching attack class label, and this labeled data is used to train the LSTM model. In order to reduce the classification loss, the model parameters (weights and biases) are improved using methods like gradient descent and backpropagation through time (BPTT). It is significant to remember that the particular architecture and configuration of the LSTM model may change based on the demands and features of the task of detecting and classifying DDoS attacks. Although there may be more factors and optimizations to take into account during actual implementation, the aforementioned equations offer a general foundation for using LSTM in this situation.

D. Hybrid CNN and LSTM Learning Approach

During the hybrid training stage, we trained the CNN and LSTM modules separately. The goal is to minimize the loss as much as possible using cross-entropy loss. Each model processes different input information. The CNN module performs 1-D convolution on a batch of individual network traffic data instances, while the LSTM module uses timestamp-based input. We group input instances from the same time frame in a batch and provide them as input to the LSTM block. These models consider different aspects of the data. For example, the CNN module focuses on spatial information, while the LSTM module extracts dependencies between similar or close timeframe instances. Once both models converge, we combine their output to obtain a reliable identification of DDoS attacks. The cross-entropy loss is given by

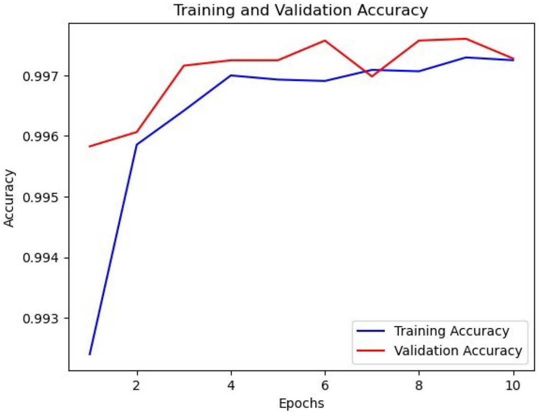

Fig. 2. Representation of training and validation accuracy.

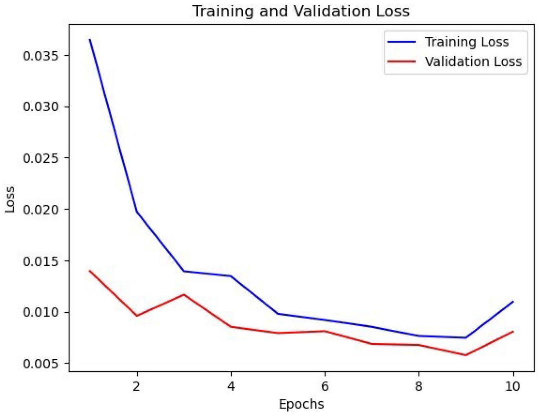

Fig. 3. Representation of training and validation loss.

$$CE = -\sum_{i}^{C} t_i \log(s_i) \tag{10}$$

where t_i and s_i are the ground truth and the CNN score for each class $_i$ in C. Usually, an activation function (Sigmoid/Softmax) is applied to the scores before the CE Loss computation, we write $f(s_i)$ to refer to the activations.

During the second stage of training, we combine the outputs of the two pretrained networks. Let $\boldsymbol{x}_i$ represent a data instance related to network traffic, and let $\boldsymbol{o}_i^c$ and $\boldsymbol{o}_l^c$ represent the corresponding output embeddings of the CNN and LSTM networks, respectively, after removing their classification layers. The embedding for $\boldsymbol{x}_i$ is obtained by merging these two outputs and creating a new vector, $x_i' = [\boldsymbol{o}_i^c, \boldsymbol{o}_l^c]$. This embedding vector,

TABLE III
CICDDoS2019 Data Set Distribution

Training Instances	Testing Instances
87619, 77, 1	3756,2

$\boldsymbol{x}_i'$ is then fed as input to the network classification layers, represented by $f(:)$. The embedded vector $\boldsymbol{x}_i'$ combines spatial and temporal dependencies for more meaningful information.

Furthermore, the hyperparameters of both modules, such as the number of layers and filters, are determined through a cross-validation process during network training.

IV. Experimental Setup

In this section, you will find information on the experimental design, data set details, evaluation measures, as well as hardware and software specifications that were utilized in the research. We used 5.4.0-6 Ubuntu and the system will be equipped with 32 GB of random access memory (RAM) and 1 graphics processing unit (GPU) with 4 GB of graphics RAM. The TensorFlow framework is utilized to implement the deep neural network [33].

A. Data Set

In this work, CICDDoS2019 has been used to apply the models, which consist of a large number of various DDoS attacks that can be executed by protocols of the application layer using connection-oriented and connectionless protocols such as TCP/UDP. In the data set, the classification of attacks is done in terms of reflection-based attacks and exploitation-based attacks. To evaluate the data set for training and testing, the data set collections consist of 18 classes. The training data set consisted of different types of DDoS attacks, each saved in a separate file as shown in Table III. Several kinds of attacks are present in the data set, such as SYN, TFTP, DrDoS_NTP, Benign, Portmap, LDAP, UDP, UDP_lag, DrDoS_DNS, MSSQL, DrDoS_UDP, DrDoS_MSSQL, NetBIOS, DrDoS_NetBIOS, DrDoS_LDAP, DrDoS_SNMP, UDPLag, and WebDDoS [55].

B. Evaluation Metrics

This section provides a detailed analysis of the network performance of our model in terms of the evaluation metrics for SDN to detect the DDoS attack. The efficiency and performance of DDoS detection systems in a network can be measured using parameters accuracy (A), recall (R), precision (P), and F1-measure (F1). All these metrics are based on four measures: 1) true positive; 2) false positive; 3) true negative; and 4) false negative, as given in (11)–(16). Additionally, the results are obtained by performing fivefold cross-validation

$$\text{Classification Accuracy} = \frac{\text{Correctly Predicted Samples}}{\text{Number of Test Samples}} \times 100\% \tag{11}$$

$$\text{Classification Error} = \frac{\text{Incorrectly Predicted Samples}}{\text{Number of Testing Samples}} \times 100\% \tag{12}$$

$$\text{A} = \frac{\text{Accurately classified records}}{\text{Total Record}} \times 100\% \tag{13}$$

TABLE IV
Obtained Results Using the CNN Module

Class Label	CNN model			
	P	R	F1	A(%)
0	0.98	0.98	0.98	97.5
1	0.94	0.96	0.95	
2	0.95	0.91	0.93	
3	0.92	0.91	0.91	
4	0.99	0.90	0.95	
5	0.89	0.91	0.91	
6	0.85	0.86	0.85	
7	0.93	0.85	0.91	
9	0.89	0.85	0.86	
10	0.96	0.84	0.88	
11	0.91	0.81	0.86	
12	0.92	0.95	0.93	
13	0.90	0.93	0.91	
14	0.97	0.95	0.96	
15	0.84	0.91	0.87	
16	0.87	0.84	0.86	
17	0.90	0.94	0.92	

$$\text{P} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \times 100\% \tag{14}$$

$$\text{R} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \times 100\% \tag{15}$$

$$F1 = \frac{2 \times P \times R}{P + R} \times 100\%. \tag{16}$$

V. Result Analysis and Discussion

In this section, we present the results and performance comparison between the proposed MLDNN approach and several state-of-the-art ML techniques. First, we compare each module (CNN and LSTM) individually. Later, we will be able to present their combined performance.

The results obtained by the CNN module of the proposed MLDNN approach are shown in Table IV. The obtained results show that the proposed approach is able to obtain good precision and recall rates in all classes and an average accuracy of 97.5%. Moreover, the training and validation loss and accuracy graphical explanations are also shown in Figures 2 and 3.

The results obtained from the individual LSTM module are shown in Table V. The outcomes of the LSTM model for identifying and classifying DDoS attacks in SDN demonstrate promising results across all categories. However, the model's effectiveness varies across different classifications, unlike the CNN module. Therefore, the output of both separately trained modules is combined for better performance.

Table VI presents the performance of the combined (LSTM + CNN) module for each class. The precision and recall outcomes reveal that the integration of both modules results in superior performance compared to using just one. The proposed MLDNN combination of both CNN and LSTM obtained 99.4% average accuracy. This highlights the robustness of the combined module over any individual module.

The results presented in Tables IV–VI show that combining CNN and LSTM models for DDoS attack detection provides a synergistic strategy that takes advantage of the benefits of both architectures. The hybrid model includes a network "for the sake of traffic" that uses the spatial pattern recognition abilities of CCMs and the temporal modeling knowledge of LSTMs.

TABLE V
OBTAINED RESULTS USING THE LSTM MODULE

Class Label	LSTM model			
	P	R	F1	A(%)
0	0.98	0.98	0.98	98.3
1	0.91	0.96	0.93	
2	0.95	0.97	0.95	
3	0.93	0.99	0.96	
4	0.99	0.99	0.99	
5	0.95	0.94	0.94	
6	0.95	0.99	0.98	
7	0.97	0.95	0.98	
9	0.99	0.80	0.92	
10	0.99	0.94	0.94	
11	0.91	0.99	0.95	
12	0.92	0.95	0.93	
13	0.70	0.83	0.81	
14	0.77	0.95	0.86	
15	0.94	0.95	0.94	
16	0.87	0.84	0.86	
17	0.85	0.84	0.83	

TABLE VI
OBTAINED RESULTS USING THE PROPOSED COMBINATION OF CNN + LSTM MODULE

Class Label	CNN + LSTM model			
	P	R	F1	A(%)
0	0.98	0.98	0.98	99.4
1	0.95	1.0	0.98	
2	1.0	0.97	0.99	
3	0.99	1.0	0.99	
4	0.99	0.99	0.99	
5	1.0	1.0	1.0	
6	0.97	1.0	0.99	
7	1.0	1.0	1.0	
9	0.99	0.95	0.98	
10	0.99	1.0	1.0	
11	0.98	0.98	0.98	
12	0.99	0.99	0.99	
13	1.0	1.0	1.0	
14	0.95	0.99	0.98	
15	0.99	0.99	0.99	
16	1.0	1.0	1.0	
17	0.95	0.99	0.98	

The combo of these two makes the result more accurate by identifying not only the immediate and the extended links found in DDoS attacks but also achieving a successful detection report that would be consistent and general in nature. The main stakeholders will be the general public, local government officials, and social service agencies. The public will play a crucial role in determining the community educational outcomes. The hybrid method which is highly interpretative leads network managers to understand threats as they are able to see those threats up-close and make informed decisions of what to do next. A conclusion of the CNN and LSTM models fusion is a very powerful and flexible technique to develop the DDoS attacks detection, the latter surpassing the efficiency and performance of individual models.

A. Performance Comparison With Other ML-Based Method

Furthermore, a comparative analysis is conducted between the results generated by various algorithms. The comparison's specifics are shown in Table VII. When compared to the RF and support vector machine (SVM) methods, the MLDNN algorithm performs better. The method that is suggested offers a comprehensive analysis that takes into account relationships that are both permanent and transient. However, SVMs and RFs rely on manually created features that are extracted from the data. This means that they may not be as effective at identifying complex patterns and may require specialist knowledge in the area. The CNN-LSTM hybrid model performs better when handling large data sets, avoiding false positives, and adapting to evolving attack tactics. However, SVMs and RFs are known for being simple, understandable, and efficient in terms of computation, which makes them more appealing in situations where resource constraints or interpretability are important. Although the CNN-LSTM hybrid strategy has the potential to provide greater accuracy and durability, the choice of different approaches for detecting DDoS attacks depends on the specific needs and constraints of the application in question.

The CNN-LSTM hybrid method for DDoS attack detection has advantages over Logistic Regression as well as K-Nearest Neighbors and Ensemble Learning methodologies, which include RFs and AdaBoost. The CNN-LSTM hybrid model applied deep learning architectures each to figure out the elaborate spatial and temporal patterns without external processing of network traffic data. This way, all analysis inputs are accessible and the detection is far much accurate. Logistic Regression and KNN follow linear or nearest neighbor approach, respectively, which may not be suitable for the data sets with high dimensions and can be less accurate in solving problems with a vast number of other attributes. Ensemble learning methods, such as RFs and gradient boosting machines, integrate many models to enhance the robustness and scalability [56]. Nevertheless, they might be demanding in terms of processing resources and necessitate meticulous feature engineering to enhance results. Logistic Regression, KNN, and Ensemble Learning are suitable options for resource-limited environments or applications that prioritize transparency and interpretability. The CNN-LSTM hybrid technique is extremely versatile in adapting to changing assault tactics and has significantly decreased the occurrence of false positives.

The comparison between the CNN-LSTM hybrid technique and the combination of SVM with self-organizing maps (SOM) for DDoS attack detection [57] reveals clear advantages and disadvantages. The CNN-LSTM model independently acquires complex spatial and temporal patterns from unprocessed network data, providing thorough analysis and achieving high precision. On the contrary, SVM with SOM transformed guided learning to unguided learning which helps it arranged the input space before in morphing it into categories. Interpretability can be improved in SVM with SOM yet may have problems tracking of detailed patterns when compared to CNN-LSTM compact hybrid. However, the CNN-LSTM model additionally points to such outstanding versatility, the number of errors is reduced, and the system can work in case of the system overloads, which makes it a very good fit for any network conditions. Exceedingly, the selection depends on the specific requirements of the detection task and it furnishes the CNN-LSTM technique which is one of the strongest and adaptable answers.

TABLE VII
PERFORMANCE COMPARISON WITH OTHER ML-BASED TECHNIQUES

Model	Accuracy	Recall (R)	Precision (P)	F I-Score
Logistic Regression	0.837	0.835	0.825	0.827
KNN	0.953	0.932	0.958	0.949
SVC	0.858	0.841	0.858	0.861
Random Forest	0.942	0.952	0.965	0.953
Ensemble Classifier	0.975	0.947	0.953	0.965
ANN	0.979	0.971	0.974	0.972
SVM and Random Forest [57]	0.986	0.969	0.991	0.988
SVM and SOM [58]	0.976	0.966	0.984	0.975
Ensemble of Decision Tree and SVM [60]	0.985	0.978	0.986	0.984
Feature Selection(Genetic algorithm) [59]	0.961	0.952	0.970	0.964
Proposed (LSTM + CNN)	**0.994**	**0.995**	**0.989**	**0.993**

On the other hand, the proposed CNN-LSTM model, contrary to the genetic algorithm (GA) feature selection technique [58], can explore raw network data by identifying patterns and feed back, and hence reduce the need for manual feature selection. But gprox feature selection selects subsets of features that improve the accuracy of classification, while it may not be applicable for very large network system. This CNN-LSTM hybrid model offers a distinctive feature because it is multipurpose, reduces the number of wrong identifications, and is capable of dealing with the huge and ever-changing networks. Thus, it is definitely an appropriate option.

The CNN-LSTM hybrid system is the best performing solution to intrusion detection as compared to the rest of the approaches discussed previously. This is due to its power to learn sophisticated patterns from the unprocessed networks data right from the analysis up to and including highly accurate conclusions. On the contrary, the SVM with SOM or GA feature selection approaches are meant to provide the interpretability and flexibility that the CNN-LSTM hybrid model may be struggling with in terms of adaptation and scalability. In the case of selection of the detection method, the method that will be chosen should be in accordance with the specific needs of the problem to be solved. However, the LSTM-CNN compound is a resilient and adaptable solution that is capable of effectively solving the problem of detection of DDoS attacks in the dynamic network environment.

VI. CONCLUSION

This article introduces a new hybrid model that combines CNN and LSTM to detect DDoS attacks. The main objective of our research is to evaluate the effectiveness of this hybrid model compared to the traditional and sophisticated methods used for detecting DDoS attacks. Our analysis provides a thorough understanding of the powers and limitations of each technique, which could be very beneficial in real-world scenarios. This method has the ability to acquire complex spatial and temporal patterns from unprocessed network traffic data. The model's extraordinary capacity to thoroughly analyze data and accurately detect DDoS attacks highlights its effectiveness in addressing the ever-changing network security risks. SVMs using SOM or GA feature selection approaches are alternative methods that offer interpretability and flexibility. However, they may not exhibit the same level of resilience to detect and adapt to attack patterns as the CNN-LSTM hybrid model. The SVM with SOM algorithm combines elements of supervised and unsupervised learning, although it may encounter difficulties in accurately representing the intricate nature of DDoS assault patterns. Similarly, GA feature selection techniques can uncover useful features, but may not fully utilize the abundant information present in the raw network data.

In terms of future prospects, there exist numerous opportunities for additional research and development in the realm of DDoS attack detection. Optimizing the parameters of the CNN-LSTM hybrid model and investigating ensemble methods that include various detection techniques could improve the resilience and effectiveness of detection systems. Moreover, it is important to prioritize endeavors aimed at enhancing the comprehensibility and openness of deep learning models, as this will play a pivotal role in establishing confidence and comprehension in the decision-making procedure.

In conclusion, the CNN-LSTM hybrid approach has shown encouraging results and offers a flexible solution for identifying DDoS attacks. It is capable of effectively tackling the complex challenges presented by contemporary network security threats. The CNN-LSTM hybrid model will be crucial in protecting network infrastructures and ensuring the reliability and accessibility of critical services as the threat landscape evolves.

REFERENCES

[1] M. Cherian and S. L. Varma, "Secure SDN–IoT framework for DDoS attack detection using deep learning and counter based approach," *J. Netw. Syst. Manag.*, vol. 31, no. 3, p. 54, 2023.

[2] O. Jullian, B. Otero, E. Rodriguez, N. Gutierrez, H. Antona, and R. Canal, "Deep-learning based detection for cyber-attacks in iot networks: A distributed attack detection framework," *J. Netw. Syst. Manag.*, vol. 31, no. 2, p. 33, 2023.

[3] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 3, pp. 38–47, 2001.

[4] L. Garber, “Denial-of-service attacks rip the Internet,” *Computer*, vol. 33, no. 4, pp. 12–17, Apr. 2000.

[5] J. D. Howard, *An Analysis of Security Incidents on the Internet 1989-1995*. Pittsburgh, PA, USA: Carnegie Mellon Univ., 1997.

[6] F. Lau, S. H. Rubin, M. H. Smith, and L. Trajkovic, “Distributed denial of service attacks,” in *Proc. SMC Conf., IEEE Int. Conf. Syst., Man Cybern., Cybern. Evol. Syst., Humans, Org. Complex Interact.*, vol. 3, 2000, pp. 2275–2280.

[7] M. Ge, X. Fu, N. Syed, Z. Baig, G. Teo, and A. Robles-Kelly, “Deep learning-based intrusion detection for IoT networks,” in *Proc. IEEE 24th Pac. Rim Int. Symp. Dependable Comput. (PRDC)*, 2019, pp. 256–25 609.

[8] M. Mittal, K. Kumar, and S. Behal, “Deep learning approaches for detecting DDoS attacks: A systematic review,” *Soft Comput.*, vol. 27, no. 18, pp. 13039–13075, 2023.

[9] P. Winter, E. Hermann, and M. Zeilinger, “Inductive intrusion detection in flow-based network data using one-class support vector machines,” in *Proc. 4th IFIP Int. Conf. New Technol., Mobile Secur.*, 2011, pp. 1–5.

[10] I. S. Thaseen, B. Poorva, and P. S. Ushasree, “Network intrusion detection using machine learning techniques,” in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng.*, 2020, pp. 1–7.

[11] A. AlEroud and I. Alsmadi, “Identifying cyber-attacks on software defined networks: An inference-based intrusion detection approach,” *J. Netw. Comput. Appl.*, vol. 80, pp. 152–164, Feb. 2017.

[12] R. Mohammadi, C. Lal, M. Conti, and L. Sharma, “Software defined network-based HTTP flooding attack defender,” *Comput. Electr. Eng.*, vol. 101, Jul. 2022, Art. no. 108019.

[13] S. Guozi, W. Jiang, G. Yu, R. Danni, and L. Huakang, “DDoS attacks and flash event detection based on flow characteristics in SDN,” in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Surveill. (AVSS)*, 2018, pp. 1–6.

[14] R. Kokila, S. T. Selvi, and K. Govindarajan, “DDoS detection and analysis in SDN-based environment using support vector machine classifier,” in *Proc. 6th Int. Conf. Adv. Comput. (ICoAC)*, 2014, pp. 205–210.

[15] T. V. Phan, T. Van Toan, D. Van Tuyen, T. T. Huong, and N. H. Thanh, “OpenFlowSIA: An optimized protection scheme for software-defined networks from flooding attacks,” in *Proc. IEEE 6th Int. Conf. Commun. Electron. (ICCE)*, 2016, pp. 13–18.

[16] S. Kranthi, M. Kanchana, and M. Suneetha, “A study of IDS-based software-defined networking by using machine learning concept,” in *Proc. ICDIS*, 2022, p. 65.

[17] I. H. Abdulqadder, S. Zhou, D. Zou, I. T. Aziz, and S. M. A. Akber, “Multi-layered intrusion detection and prevention in the SDN/NFV enabled cloud of 5G networks using AI-based defense mechanisms,” *Comput. Netw.*, vol. 179, Oct. 2020, Art. no. 107364.

[18] A. Rahim, Y. Zhong, T. Ahmad, S. Ahmad, P. Pławiak, and M. Hammad, “Enhancing smart home security: Anomaly detection and face recognition in smart home IoT devices using logit-boosted CNN models,” *Sensors*, vol. 23, no. 15, p. 6979, 2023.

[19] A. Rahim, Y. Zhong, T. Ahmad, S. Ahmad, and M. A. ElAffendi, “Hyper-tuned convolutional neural networks for authorship verification in digital forensic investigations,” *Comput., Mater. Continua*, vol. 76, no. 2, p. 6979, 2023.

[20] T. Ahmad and J. Wu, “SDIGRU: Spatial and deep features integration using multilayer gated recurrent unit for human activity recognition,” *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 1, pp. 973–985, Feb. 2024.

[21] D. Javeed, T. Gao, M. T. Khan, and I. Ahmad, “A hybrid deep learning-driven SDN enabled mechanism for secure communication in Internet of Things (IoT),” *Sensors*, vol. 21, no. 14, p. 4884, 2021.

[22] N. Ashodia and K. Makadiya, “Detection of DDoS attacks in SDN using machine learning,” in *Proc. Int. Conf. Electron. Renew. Syst. (ICEARS)*, 2022, pp. 1322–1327.

[23] L. Yang and H. Zhao, “Ddos attack identification and defense using sdn based on machine learning method,” in *Proc. 15th Int. Symp. Pervasive Syst., Algorithms Netw. (I-SPAN)*, 2018, pp. 174–178.

[24] S. Yu, J. Zhang, J. Liu, X. Zhang, Y. Li, and T. Xu, “A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN,” *EURASIP J. Wireless Commun. Netw.*, vol. 2021, no. 1, pp. 1–21, 2021.

[25] H. Cheng, J. Liu, T. Xu, B. Ren, J. Mao, and W. Zhang, “Machine learning based low-rate DDoS attack detection for SDN enabled IoT networks,” *Int. J. Sensor Netw.*, vol. 34, no. 1, pp. 56–69, 2020.

[26] A. A. Bahashwan, M. Anbar, S. Manickam, T. A. Al-Amiedy, M. A. Aladaileh, and I. H. Hasbullah, “A systematic literature review on machine learning and deep learning approaches for detecting DDoS attacks in software-defined networking,” *Sensors*, vol. 23, no. 9, p. 4441, 2023.

[27] F. Alanazi, K. Jambi, F. Eassa, M. Khemakhem, A. Basuhail, and K. Alsubhi, “Ensemble deep learning models for mitigating DDoS attack in software-defined network,” *Intell. Autom. Soft Comput.*, vol. 33, no. 2, pp. 923–938, 2022.

[28] X. Jiang, H.-R. Zhang, and Y. Zhou, “Multi-granularity abnormal traffic detection based on multi-instance learning,” *IEEE Trans. Netw. Service Manag.*, vol. 21, no. 2, pp. 1467–1477, Apr. 2024.

[29] M. Waqas, M. A. Tahir, and R. Qureshi, “Ensemble-based instance relevance estimation in multiple-instance learning,” in *Proc. 9th Eur. Workshop Visual Inf. Process. (EUVIP)*, 2021, pp. 1–6.

[30] M. Waqas, M. A. Tahir, and R. Qureshi, “Deep Gaussian mixture model based instance relevance estimation for multiple instance learning applications,” *Appl. Intell.*, vol. 53, no. 9, pp. 10310–10325, 2023.

[31] M. Waqas, M. A. Tahir, and S. A. Khan, “Robust bag classification approach for multi-instance learning via subspace fuzzy clustering,” *Expert Syst. Appl.*, vol. 214, Mar. 2023, Art. no. 119113.

[32] A. Zainudin, L. A. C. Ahakonye, R. Akter, D.-S. Kim, and J.-M. Lee, “An efficient hybrid-DNN for DDoS detection and classification in software-defined IIoT networks,” *IEEE Internet Things J.*, vol. 10, no. 10, pp. 8491–8504, May 2023.

[33] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,” in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, 2019, pp. 1–8.

[34] H. Aydın, Z. Orman, and M. A. Aydın, “A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defense system design in public cloud network environment,” *Comput. Secur.*, vol. 118, Jul. 2022, Art. no. 102725.

[35] Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu, and S. Camtepe, “AE-MLP: A hybrid deep learning approach for DDoS detection and classification,” *IEEE Access*, vol. 9, pp. 146810–146821, 2021.

[36] S. J. Rani et al., “Detection of DDoS attacks in D2D communications using machine learning approach,” *Comput. Commun.*, vol. 198, pp. 32–51, Jan. 2023.

[37] Dec. 2019, B. L. Dalmazo, V. M. Deolindo, and J. C. Nobre, “Public dataset for evaluating port scan and slowloris attacks,” Dataset, Harvard Dataverse. [Online]. Available: https://doi.org/10.7910/DVN/ZJOT5G

[38] M. S. El Sayed, N.-A. Le-Khac, M. A. Azer, and A. D. Jurcut, “A flow-based anomaly detection approach with feature selection method against DDoS attacks in SDNs,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 4, pp. 1862–1880, Dec. 2022.

[39] M. S. Elsayed, N.-A. Le-Khac, and A. D. Jurcut, “InSDN: A novel SDN intrusion dataset,” *IEEE Access*, vol. 8, pp. 165263–165284, 2020.

[40] A. Makuvaza, D. S. Jat, and A. M. Gamundani, “Deep neural network (DNN) solution for real-time detection of distributed denial of service (DDoS) attacks in software defined networks (SDNs),” *SN Comput. Sci.*, vol. 2, pp. 1–10, Feb. 2021.

[41] I. Sharafaldin, A. H. 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*, vol. 1, 2018, pp. 108–116.

[42] A. Agarwal, M. Khari, and R. Singh, “Detection of DDoS attack using deep learning model in cloud storage application,” *Wireless Pers., Commun.*, vol. 127, pp. 1–21, Mar. 2021.

[43] Y. Liu, T. Zhi, M. Shen, L. Wang, Y. Li, and M. Wan, “Software-defined DDoS detection with information entropy analysis and optimized deep learning,” *Future Gener. Comput. Syst.*, vol. 129, pp. 99–114, Apr. 2022.

[44] S. Singh and S. Jayakumar, “DDoS attack detection in SDN: optimized deep convolutional neural network with optimal feature set,” *Wireless Pers. Commun.*, vol. 125, no. 3, pp. 2781–2797, 2022.

[45] U. KDD, “The third international knowledge discovery and data mining tools competition dataset KDD cup 1999 data,” Dataset, UCI KDD Archive. [Online]. Available: http://kdd. ics. uci. edu/databases/kddcup99/kddcup 99. html

[46] L. Zhou, Y. Zhu, T. Zong, and Y. Xiang, “A feature selection-based method for DDoS attack flow classification,” *Future Gener. Comput. Syst.*, vol. 132, pp. 67–79, Jul. 2022.

[47] T. G. Palla and S. Tayeb, “Intelligent Mirai malware detection for IoT nodes,” *Electronics*, vol. 10, no. 11, p. 1241, 2021.

[48] 2007, P. Hick, E. Aben, K. Claffy, and J. Polterock, “the caida ddos attack 2007 dataset,” dataset, Caida. [Online]. Available: http://www.caida.org/data/passive/ddos-20070804_ dataset. xml

[49] M. S. Elsayed, N.-A. Le-Khac, S. Dev, and A. D. Jurcut, “DDoSNet: A deep-learning model for detecting network attacks,” in *Proc. IEEE 21st Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, 2020, pp. 391–396.

[50] R. Priyadarshini and R. K. Barik, “A deep learning based intelligent framework to mitigate DDoS attack in fog environment,” *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 3, pp. 825–831, 2022.

[51] P. A. Alves Resende and A. Costa Drummond, “Http and contact-based features for botnet detection,” *Secur. Privacy*, vol. 1, no. 5, p. e41, 2018.

[52] M. Waqas, Z. Khan, S. Anjum, and M. A. Tahir, “Lung-wise tuberculosis analysis and automatic CT report generation with hybrid feature and ensemble learning,” presented at the CLEF (Working Notes), 2020, pp. 1–10.

[53] M. Hanif, M. Waqas, A. Muneer, A. Alwadain, M. A. Tahir, and M. Rafi, “DeepSDC: Deep ensemble learner for the classification of social-media flooding events,” *Sustainability*, vol. 15, no. 7, p. 6049, 2023.

[54] M. Waqas, Z. Khan, S. U. Ahmed, and A. Raza, “MIL-mixer: A robust bag encoding strategy for multiple instance learning (MIL) using MLP-mixer,” in *Proc. 18th Int. Conf. Emerg. Technol. (ICET)*, 2023, pp. 22–26.

[55] J. Min, S. Yuejie, G. Qing, G. Zihe, and X. Suofei, “DDoS attack detection method for space-based network based on SDN architecture,” *ZTE Commun.*, vol. 18, no. 4, pp. 18–25, 2021.

[56] N. Ahuja, G. Singal, D. Mukhopadhyay, and N. Kumar, “Automated DDoS attack detection in software defined networking,” *J. Netw. Comput. Appl.*, vol. 187, Aug. 2021, Art. no. 103108.

[57] T. V. Phan, N. K. Bao, and M. Park, “A novel hybrid flow-based handler with DDoS attacks in software-defined networking,” in *Proc. Int. IEEE Conf. Ubiquitous Intell. Comput., Adv. Trust. Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People, Smart World Congr. (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, 2016, pp. 350–357.

[58] K. S. Sahoo et al., “An evolutionary SVM model for DDoS attack detection in software defined networks,” *IEEE Access*, vol. 8, pp. 132502–132513, 2020.