Journal of Computer Science

Original Research Paper

A Deep Ensemble Framework for DDoS Attack Recognition and Mitigation in Cloud SDN Environment

[1,2]S. Annie Christila and [1]R. Sivakumar

[1]*Department of Computer Science, CHRIST (Deemed to be University), Bengaluru, India*
[2]*Department of Computer Science, St. Francis de Sales College, Bengaluru, India*

Article history
Received: 26-04-2024
Revised: 08-07-2024
Accepted: 09-07-2024

Corresponding Author:
S. Annie Christila
Department of Computer Science, CHRIST (Deemed to be University), Bengaluru, India
Email: annie.s@res.christuniversity.in

Abstract: Much research has been done in the recent past on the absolute shift of Internet infrastructure in order to make it more significantly programmable, configurable and make it more conveniently feasible. Software Defined Networking (SDN) forms the basis for this absolute shift in Internet infrastructure. When you look at the benefits of an SDN-based cloud environment they are monumental. Namely, network traffic control and elastic resource management. The SDN-based cloud environment becomes susceptible to cyber threats, especially like that of Distributed Denial of Service (DDoS) attacks and other cyber-attacks that perturb the SDN-based cloud environment. Hence, automated Machine Learning (ML) models are an efficient way to protect against these cyber-attacks. This research will develop a deep learning-based ensemble model for DDoS attack detection and classification (DLEM-DDoS) in a cloud environment. Long Short-Term Memory (LSTM), 1-D Convolutional Neural Networks (1D-CNN) and Gated Recurrent Unit (GRU) are the three DL models integrated into an ensemble model that classifies the incoming packet by majority voting classifiers. Network traffic data including source and destination IP addresses, packet and byte counts, packet and byte rates, flow duration, protocol types and port numbers are fed into the DLEM-DDoS model. This model preprocesses this data by converting categorical values (like protocol types) into numerical values and removing any missing values. Once collected and preprocessed, the data is fed into deep learning models (LSTM, 1D-CNN, GRU) within the framework for analysis. Finally, in this research using the DLEM-DDoS technique an efficient DDoS attack mitigation scheme in an SDN-based cloud environment is demonstrated. The report shows comprehensive stimulations as well as a superiority into the current approaches in terms of several measures.

Keywords: Distributed Denial of Service (DDoS), Software Defined Network (SDN), Deep Learning Based Ensemble Model (DLEM), Machine Learning (ML), Virtual Machine (VM)

Introduction

When it comes to Software Defined Networks (SDN) Cloud Computing (CC) and Software Defined Networks (SDN) have made a significant impact at a pivotal moment of SDN both in the industry and academia wider range of acceptance in the network community (Sahay *et al.*, 2019). In the current scenario, it is essential to come up with secured cost-effective computing resources for the healthcare/banking and other critical service industries. Cloud is one such solution to bring cost-effective secured computing resources and SDN is the technology that provides networking connectivity and also, provides a method to move VM from one server to the other in the cloud environment to achieve cost-effectiveness. Servers can be co-located or placed in different data centers. The flexibility SDN brings to the cloud environment helps to provide cost-effective VMs to the customers.

However, this SDN-based cloud environment is prone to security attacks, especially through Distributed Denial of Service attacks. Hence there is a need to identify and mitigate the DDoS attack security problem that occurs in SDN-based cloud networking environments. In this study method is proposed to identify and mitigate DDoS attack problems in SDN SDN-based cloud environment.

Science Publications

SDN structure separates the control plane from the data plane. The control plane was consistently synchronized. SDN's plan is to empower the development and simplifying the network via network programming (Jing, 2017). Centralization assists in disentangling the use of network strategy through programming, which is different from customary systems that make use of lower-level device configurations. The programmability of the network could effectively control the underlying data plane from SDN (Mayoral *et al*., 2017). SDN has greatly helped towards the achievement of the CC network model. The CC has a considerable point of interest compared to traditional computing methods (Pillutla and Arjunan, 2019) and also provides extensible and dynamic virtualization resources. Both CC and SDN have fundamentally developed in the scholarly industry and community due to their key characteristics.

Keeping aside, the advantages of SDN that decouple the control plane in the data plane have increased the attack surface of SDN in comparison with traditional network systems (Agrawal and Tapaswi, 2021). Besides the conventional attacks (repudiation, spoofing, elevation of privilege tampering, information disclosure and Denial of Service (DoS), also SDN becomes susceptible to several new attacks such as data leakage (through packet processing timing analysis and flow rule discovery), man-in-the-middle attack, unauthorized access (of applications, controller switches and so on) malicious application (for installing fake rules) and other configuration problems (Mousavi and St-Hilaire, 2018). SDN is proven more vulnerable to DoS attack that has obviously considerable repercussions. When DoS attacks or their distributed form that is DDoS attacks, are successfully deployed, they crippled the controller of the network system by disabling a switch or the entire network system by disabling a component. The DoS attacks on SDN include the networking resources or overwhelming computing thus a switch becomes incapable of forwarding the packet. Unwanted enormous requests for installing new rules in the OFF switch are sent to the controller as an effective DoS attack (Banikazemi *et al*., 2013).

The attacker uses a DDoS attack to make this cloud service inaccessible to legitimate users. In such attacks, the attacker puts greater pressure on the service given by the server which is more vulnerable to security risks on the public networks (Jeong *et al*., 2012). This leads to the usage of each bandwidth of the victim and it becomes inaccessible. Anomaly and Signature-based mitigation methods have been developed to prevent DDoS attacks however they require exclusive devices for investigating and gathering data traffic via the applications of stochastic analysis and Machine Learning (ML) (Yan *et al*., 2015). They need separate memory space to compare the inward data traffic and the attack data traffic and almost all of them are worked in offline mode to identify trustworthy sources that have created the attacks. Various ML-based solutions for identifying DDoS attacks in CC have been introduced (Lin *et al*., 2014). The major problem in ML-based solutions is the recognition of this attack having higher accuracy.

This study designs a deep learning-based ensemble model for DDoS attack detection and classification (DLEM-DDoS) in SDN oriented cloud environment. The DLEM-DDoS technique primarily carries out the data pre-processing in two levels namely data transformation and null value removal. Besides, an ensemble of majority voting classifiers is designed by the incorporation of three DL models namely Long Short-Term Memory (LSTM), 1-D Convolutional Neural Networks (1D-CNN) and Gated Recurrent Unit (GRU). At last, the DLEM-DDoS technique designs an effective DDoS attack mitigation scheme in the SDN-enabled cloud environment. To evaluate the enhanced performance that the DLEM-DDoS approach yields, a set of simulations are carried out on the standard dataset and the results are studied under different measures.

Problem Statement

To design and implement an improved Distributed Denial of Service (DDoS) attack detection and prevention mechanisms and also to prove minimal False Positive (FP) rates in a Software Defined Network (SDN) based cloud environment.

Objectives

- Primary objectives: To develop a DDoS prevention mechanism in the SDN and to arrive at an effective DDoS attack detection and mitigation mechanism in the SDN
- Secondary objectives: To minimize the False Positive rates of DDoS attack detection and to minimize the execution time which conserves more energy and improves the QoS

Literature Review

Tan *et al*. (2020) projected an architecture for defining and detecting DDoS attacks in SDN environments. At first, it deploys a trigger model for detecting DDoS attacks on data planes for screening abnormal flows in the system. Next, it uses an integrated ML approach based on KNN and K-means to detect the suspectable flows defined by the detection trigger method and to exploit the asymmetry and rate characteristics of the flows. Pillutla and Arjunan (2019) presented a Fuzzy Self-Organizing Map-based DDOS Mitigation (FSOMDM) method i.e., desirably and optimally developed to improve the SDN capability of CC. FSOMDM is the improved NN system that efficiently replaces the neurons of the conventional Kohonen NN system by upgrading the fuzzy rules. The properties of software-related traffic study are employed and the fuzzy rules are utilized for examining the dimensions of input space where a single-valued outcome was acquired to enable the mitigations of DDoS.

Phan and Park (2019) developed effective solutions to address DDoS attacks from SDN-based cloud platforms. Firstly, introduced a hybrid ML method that depends on the Self-Organizing Map (SOM) and SVM method for improving traffic classification. Next, presented an enhanced History-based IP Filtering system (eHIPF) for improving the attack detection speed and rate. At last, proposed a novel method that integrates hybrid ML and eHIPF systems for making a DDoS attack protector to SDN SDN-based cloud platform. Virupakshar *et al*. (2020) proposed an approach using raw socket programming and an OpenStack-integrated firewall to monitor the network traffic. The data sets created in controlled DDoS attack environments, namely DT, KNN, NB and DNN methods are compared against the training model.

Bhushan and Gupta (2019) introduced a flow-table sharing method for protecting the SDN-based cloud systems in flow table over-loading DDoS attacks. The presented method uses an idle flow-table of OpenFlow switch for protecting the switch flow-table from overloading. The presented method increased the tolerance of cloud platforms towards DDoS attacks along with the minimum contribution of SDN controllers. Chen *et al*. (2018) made use of XGBoost, in the form of a detection system in SDN SDN-based cloud network. The XGBoost classifiers use the flow packet dataset gathered by TcpDump for detecting DDoS attacks and compare it with other classifications.

Agrawal and Tapaswi (2021) discussed a new method that traces back the location of the attack source and mitigates and detects the shrew attack. The attack can be identified by the data entropy variation and the attack source is traced back with the packet marking system. Kushwah and Ranga (2021) introduced a system for detecting DDoS attacks based on an enhanced Self adoptive Evolutionary-Extreme Learning Machine (SaE-ELM). The presented method was enhanced by integrating more than two features. At first, it adopts a better-suited crossover operator. Next, it automatically determines the suitable number of hidden layers. This feature improves the classification and learning abilities of the system.

The Proposed Model

The research shows a new model in which it presents the Deep Learning Ensemble Model for DDoS (DLEM-DDoS) that technique efficiently detects and mitigates DDoS attacks in SDN-based cloud environments. Operating at the SDN switches' data plane, it follows a three-stage process: Data preprocessing, classification and attack mitigation. This involves converting categorical network traffic features (like IP addresses and protocols) into numerical values and removing or handling any missing values to ensure a complete and consistent dataset. An ensemble of three deep learning models-LSTM, 1D-CNN and GRU-is used. Each model is trained on the pre-processed data to capture important temporal patterns. The final classification is determined through a majority voting scheme, which combines the predictions of all three models for improved accuracy. When an attack is detected, the system executes a real-time mitigation strategy, which includes rerouting or blocking malicious traffic to protect network services. By integrating these stages, the DLEM-DDoS technique provides a robust and accurate solution for combating DDoS attacks, leveraging the strengths of multiple deep learning models to ensure effective detection and mitigation. The DLEM-DDoS technique involves an effective DDoS attack mitigation scheme from the SDN-enabled cloud environment. Figure 1 shows the comprehensive modus operandi of the proposed method.

Data Pre-Processing

In the initial stage, pre-processing of data takes place in two major levels namely data transformation and null value removal. During the data transformation process, the categorical values are changed into numerical values. In this study, the categorical values exist under three attributes namely SRC, DST and protocol, which are then transformed into numerical ones. Besides, the null values exist in the actual dataset and the resultant pre-processed data are passed into the classification stage.

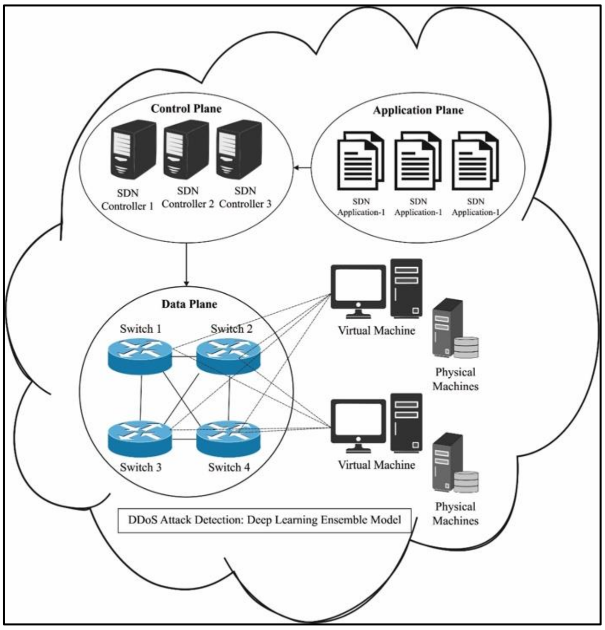


Fig. 1: Overall system architecture

Ensemble of DL Models

During the DDoS attack detection and classification process, the ensemble of DL models using the majority voting scheme gets executed. It is an ensemble ML model which integrates the predictive results of many models and it helps to boost the model results. It can be employed for improving high-quality performance compared to a single model. During the classification process, the estimations for every label are summed and the label having the majority vote is considered.

LSTM Model

LSTM is also a modified version of the RNN model that has the ability to solve the gradient vanishing problem by the use of input gate *i*, output gate 0, forget gate *f* and memory cell state. It helps to enhance the storage process of the NN in receiving the input and training data. It is helpful to model the time series data like text owing to its unique features (Hochreiter and Schmidhuber, 1997). The LSTM model comprises 3 control gates namely input gate i_t, forget gate f_t, outcome gate 0_t and memory cell state c_t, which affects the capability of storing and updating data. The input gate output provides a value in the range of 0-1 depending upon the input $h_{t-1}$and w_t. If the outcome becomes 1, it is indicated that the cell state data is totally recollected and if the outcome becomes 0, it is totally uncontrolled. Figure 2 depicts the LSTM framework.

Then, the input gate layer chooses which value requires upgrading and the tanh layer generates a new candidate value vector $\widetilde{c_t}$, that is appended to the cell state.

Consequently, they can be integrated with updating the cell state c_t (Eqs. 2-4); at last, the output gate defines the resultant dependent upon the cell state. Amongst them $W_f, U_f, b_f, W_i, U_i, b_j, W_c, U_c, b_c$ and W_0, U_0, b_0 are the internal trained parameters from LSTM, $\sigma(\cdot)$ is the sigmoid activation function and $\odot$ implies the dot multiplication (Pogiatzis and Samakovitis, 2020).

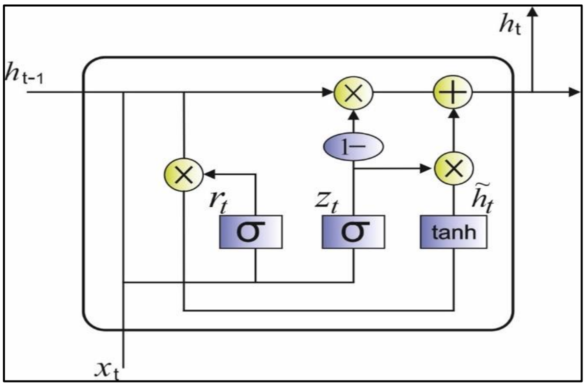


Fig. 2: LSTM structure

$$f_t = \sigma\left(W_f w_t + U_f h_{t-1} + b_f\right) \tag{1}$$

$$i_t = \sigma(W_i w_t + U_i h_{t-1} + b_i) \tag{2}$$

$$\widetilde{c_t} = tanh(W_c w_c + U_c h_{t-1} + b_c) \tag{3}$$

$$c_t = i_t \odot \overline{c_t} + f_t \ \odot c_{t-1} \tag{4}$$

$$0_t = \sigma(W_0 w_t + U_0 h_{t-1} + b_0) \tag{5}$$

$$h_t = o_t \tanh \ \odot (c_t) \tag{6}$$

This model uses network traffic features like source and destination IP addresses, packet and byte counts, packet and byte rates, flow duration, protocol types and port numbers as input data. It processes this information through its unique architecture, utilizing input, forget and output gates to handle memory and learn from temporal patterns. The model's performance is assessed with various metrics and the outcomes are visualized to demonstrate its effectiveness in identifying DDoS attacks within an SDN-based cloud environment. The LSTM model outputs a set of probabilities for each category (normal and attack), which are employed to classify each input sequence.

1D-CNN Model

The 1D-CNN is utilized for supervised learning on time-series data. The presented framework is in Fig. 1. We employ 1-time step, thirty-two convolutional filters, five kernel sizes and forty-two features in a convolutional layer at the top rate. It applied a 'sigmoid' activation function in the convolutional layer. Determining a single filter will permit the 1D-CNN to learn a single feature in the initial convolutional layer (Chen *et al*., 2022). This is inadequate; thus, we create thirty-two filters that allow us to extract thirty-two distinct features on the initial convolutional layer. The output of the initial convolutional layer gives a 37×32 neuron matrix. All the filters contain their own weight along with the determined kernel size, which is considered as the input matrix's length. Other convolutional layers follow the initial convolutional layer. Max pooling, a type of convolutional layer is adapted for a down-sampling output. During the max pooling process, max filters select the maximal values where the filter is used. It minimizes the spatial size of the output and decreases the computation difficulty and the feature count. Like the dropout layer, a max pooling layer offers generalization. It used 1D max pooling having a pooling stride size 1 and size 2. Eventually, we add two FC layers to implement the classification. In general, the FC layer includes learnable parameters. To avoid overfitting, we apply the softmax function that calculates the probability for all the classes. We also applied dropout methods beforehand FC layer. A dropout offers generalization (a regularization of outcomes for hidden data) by dropping out (set to 0) a percentage of output from the preceding layer. Apparently, this non-sensical action forces the network to depend on certain inputs, for improving generalization and preventing overfitting. We applied the BN method to improve performance results and make fast convergence. At the training time, it can be performed by normalizing, all the features at batch level (scaling input to zero mean and unit variance) and rescaling again considered the entire training dataset. This model generates a set of probabilities for each category (normal and attack). These probabilities help in determining whether each input sequence represents regular network traffic or a DDoS attack.

GRU Model

The RNN model is presented for capturing the temporal correlation. The GRU is an extended version of the RNN depending upon the LSTM model. If error signals are propagated in the backward direction over time in the traditional RNN, they seem to have vanished and resulted in network failure in learning the data. The GRU model sustains the capability of preventing the above-mentioned problem and also minimizes the complexity of the structure with no loss of effectual learning capability. The earlier hidden state helps to save the earlier memory and the reset gate helps to manage the way of combining the input to the earlier memory and the upgrade gate manages the way of adding the candidate's hidden state into the hidden state (Chung *et al*., 2014). The GRU cell was defined using the following equations:

$$\begin{aligned} z_t &= \sigma(W^z x_t + U^z h_{t-1} + b^z) \\ r_t &= \sigma(W^r x_t + U^r h_{t-1} + b^r) \end{aligned} \tag{7}$$

$$\begin{aligned} \widetilde{h_t} &= \tanh\left(W x_t + U(r_t \odot h_{t-1}) + b\right), \\ h_t &= (1 - z_t) \odot \widetilde{h_t} + z_t \odot h_{t-1} \end{aligned} \tag{8}$$

where, h_{t-1} indicates the hidden state at $t-1$ and $x_t, z_t,$ ⬚ rt, h_t and $\widetilde{h_t}$ imply the input of GRU cell, outcome of upgrade gate, output of reset gate, candidate hidden state and hidden state at t, correspondingly. W and U indicates the weight matrix of the FC layer (Xie *et al*., 2021) and b refers to the bias vectors. σ and $\tanh$ indicate sigmoid and $\tanh$ activation functions correspondingly. $\odot$ denotes the element-wise multiplication of 2 matrices of equivalent sizes. This model processes network traffic data through its specialized architecture, employing update and reset gates to manage memory and learn temporal patterns. It outputs a set of probabilities for each category (normal and attack), which are used to classify each input sequence.

DDoS Mitigation Scheme

DDoS mitigation describes the procedure of effectively defending a target network or server from DDoS attacks. By using the cloud-based protection services o network equipment, a targeted victim was capable of mitigating the inward threats. There exist four phases of DDoS attack mitigation as illustrated in Fig. 3.

Detection: To stop distribution attacks, a website must be capable of distinguishing attacks from a higher amount of normal traffic. When the product is released or other announcements have a website flooded with legit novel visitors, the final step the site wants is to throttle them or else stop them from seeing the website contents. Common attack patterns, preceding data and IP reputation assist in detecting appropriate attacks.

Response: During this phase, the DDoS defense system is responsive to inward threats through the intelligent removal of intrusive bot traffic, also pulling in the rest of the traffic. By utilizing the WAF page rule to application Layer (L7) attack, or other filtration processes for managing low-level (L3/L4) attacks including Memcached or NTP amplification, then the network was capable of mitigating the attempts at distraction.

Routing: With smartly routing traffic, an efficient DDoS mitigate solution breaks the residual traffic into controllable chunks preventing DoS.

Adaptation: Evaluates traffic to pattern namely repeat offending IP blocks and certain protocols that have been improperly utilized. Adopting, protection services could reinforce against upcoming attacks.

Algorithm 1: Pipeline of Mitigation Process

```
defdeal_With_Attackers(self, victims):
    Push_backs = set()
    attackers = set()
    for victim in victims:
    Victim_Host = self.get_Victim_Host(victim)
    victim_Switch self.get_Victim_Switch(victim)
    victim_Attackers = self.get_Attackers(victim)
    victim_Attackers = self.get_Attackers(victim)
print("Attackers for victim %s: %s" %
              (victim_Attackers, victim_Host))
if not victim_Attackers:
        push_backs.add(victim)
else:
 attackers= attackers.union(victim_Attackers)
```

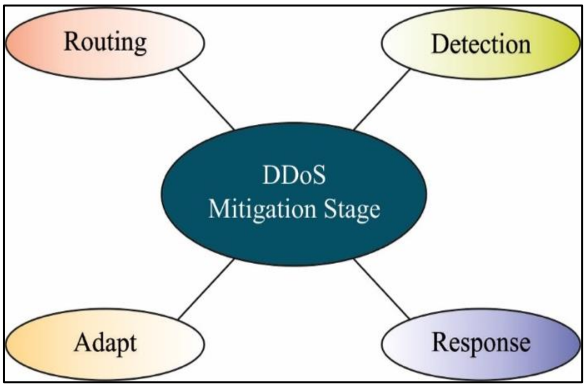


Fig. 3: Stages of DDoS mitigation

Materials and Methods

Proposed an improved Distributed Denial of Service (DDoS) attack detection and prevention mechanisms in a Software Defined Network (SDN) based cloud environment.

This method operating at the SDN switches' data plane, it follows a three-stage process: Data preprocessing, classification and attack mitigation. This involves converting categorical network traffic features (like IP addresses and protocols) into numerical values and removing or handling any missing values to ensure a complete and consistent dataset. An ensemble of three deep learning models-LSTM, 1D-CNN and GRU-is used. The final classification is determined through a majority voting scheme for improved accuracy. When an attack is detected, the system executes a real-time mitigation strategy. Figures 1-3 shows the comprehensive modus operandi of the proposed method.

Dataset

In this study, SDN specific data set generated by using mininet emulator is used for traffic classification. (Mukhopadhyay and Singal, 2020). Multiple topologies were created in mininet in which switches are connected to single Ryu controller. Data set is created by network simulation runs for benign TCP, UDP and ICMP traffic and malicious traffic which is the collection of TCP Syn attack, UDP Flood attack, ICMP attack. The dataset includes a set of 23 attributes. Information about the attributes are provided in results and discussion section below.

Results and Discussion

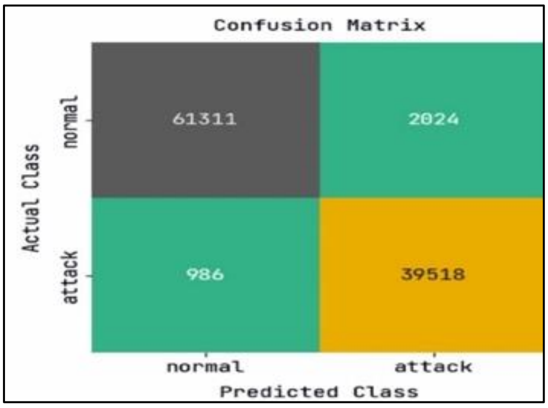
The performance evaluation of the DLEM-DDoS technique is carried out with the benchmark DDOS attack SDN dataset (Mukhopadhyay and Singal, 2020) which is produced by the use of a mininet emulator. The dataset includes a set of 23 attributes namely dt, switch, SRC, dst, pktcount, bytecount, dur, dur_nsec, tot_dur, flows, packetins, pktperflow, byteperflow, pktrate, Pairflow, Protocol, port_no, tx_bytes, rx_bytes, tx_kbps, rx_kbps, tot_kbps, label. The dataset includes instances under benign (class 0) and malicious (class 1). After the data pre-processing, the number of instances becomes 103839 with the inclusion of 63335 instances under benign class and 40504 instances under malicious class.

Figure 4 demonstrates the confusion matrices generated by the three DL models and the DLEM-DDoS technique. Figure 4a shows that the LSTM approach has detected and classified 61311 instances into the normal class and 39518 instances into the attack class. Similarly, Fig. 4b-c displays that the 1D-CNN technique has detected and categorized 61801 instances into normal class and 39277 instances into attack class. Likewise, Fig. 4d illustrates that the DLEM-DDoS approach has recognized 61894 instances in the normal class and 39594 instances in the attack class.

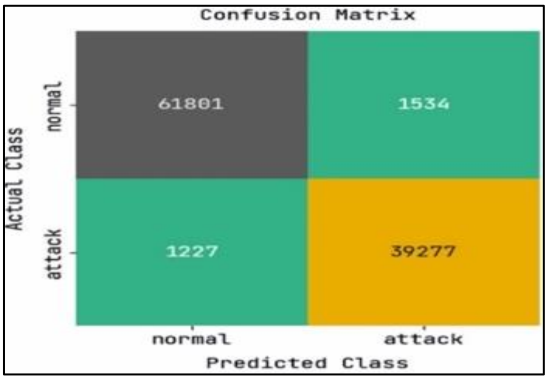
Figure 5 offers the training and validation (accuracy/loss) graph analysis of the three DL models. The training and validation accuracy analysis reported that the validation accuracy seems the maximum related to the training accuracy. Also, the validation loss is considered to be lower compared to the training loss.

Table 1 and Fig. 6 show the overall precision-recall analysis of LSTM, 1D-CNN, GRU and DLEM-DDoS models. From the figures, it is clear that the DLEM-DDoS approach has offered better classification efficiency than the other methods.

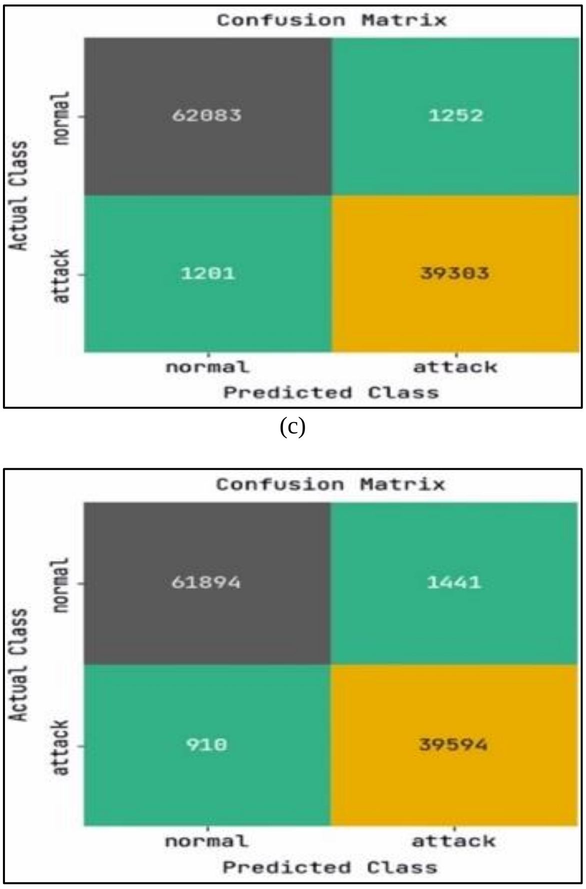
The ROC analysis of the DLEM-DDoS system with other DL models is provided in Fig. 7. It is proven from the results that the LSTM model has obtained ROC of 0.9718 and 0.9718 under normal and attack categories respectively. Also, the outcomes demonstrated that the 1D-CNN model has gained ROC of 0.9727 and 0.9727 under normal and attack classes correspondingly. Furthermore, the results show that the GRU approach has obtained ROC of 0.9753 and 0.9753 under normal and attack classes correspondingly. Lastly, the outcomes outperformed that the DLEM-DDoS system has reached ROC of 0.9774 and 0.9774 under normal and attack classes correspondingly.



(a)



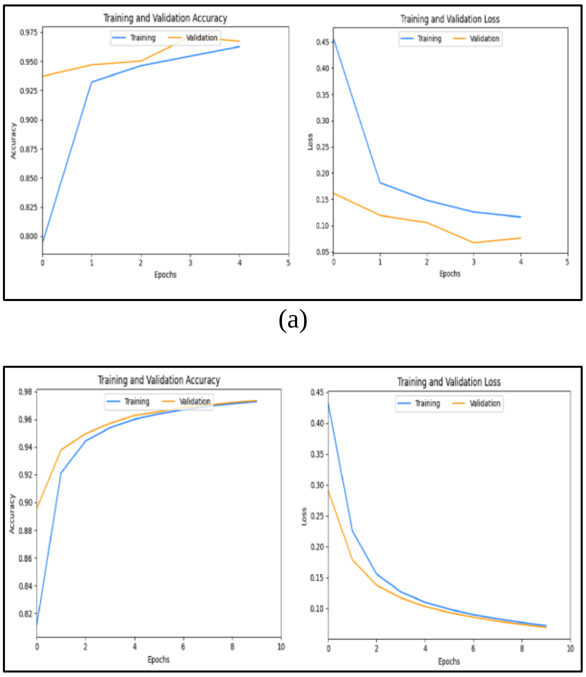
(b)



(c)

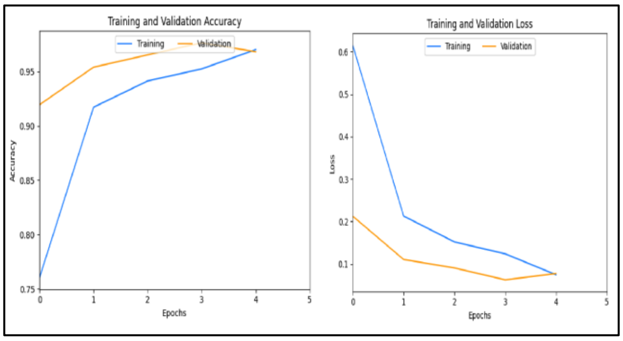
(d)

Fig. 4: Confusion matrix; (a) LSTM (b); 1D-CNN (c); GRU (d); DLEM-DDoS



(a)

(b)



(c)

Fig. 5: Training and validation (accuracy/loss); (a) LSTM (b); 1D-CNN (c); GRU

Table 1: Precision-recall curve analysis of LSTM, 1D-CNN, GRU and DLEM-DDoS models

Model	Normal class	Attack class
LSTM	0.9722	0.9376
1D-CNN	0.9716	0.9451
GRU	0.9737	0.0952
DLEM-DDoS	0.9777	0.0952

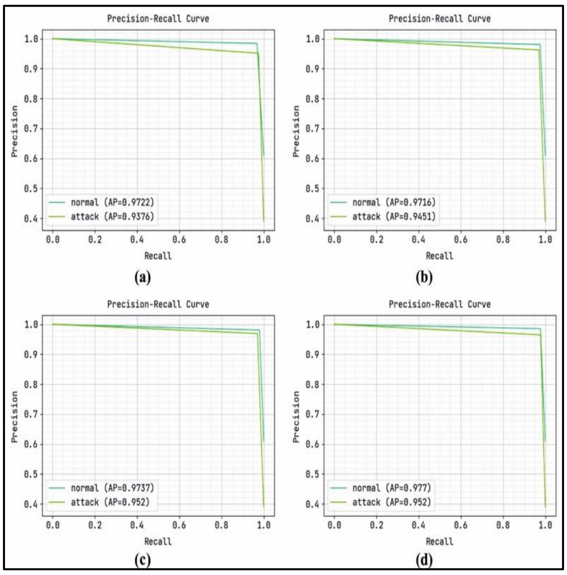


Fig. 6: Precision recall curve a) LSTM b) 1D-CNN c) GRU d) DLEM-DDoS

Table 2: ROC analysis of LSTM, 1D-CNN, GRU and DLEM-DDoS models

Model	Normal class	Attack class
LSTM	0.9718	0.9718
1D-CNN	0.9727	0.9727
GRU	0.9753	0.9753
DLEM-DDoS	0.9774	0.9774

Table 2 and Fig. 7 show the overall ROC analysis of LSTM, 1D-CNN, GRU and DLEM-DDoS models.

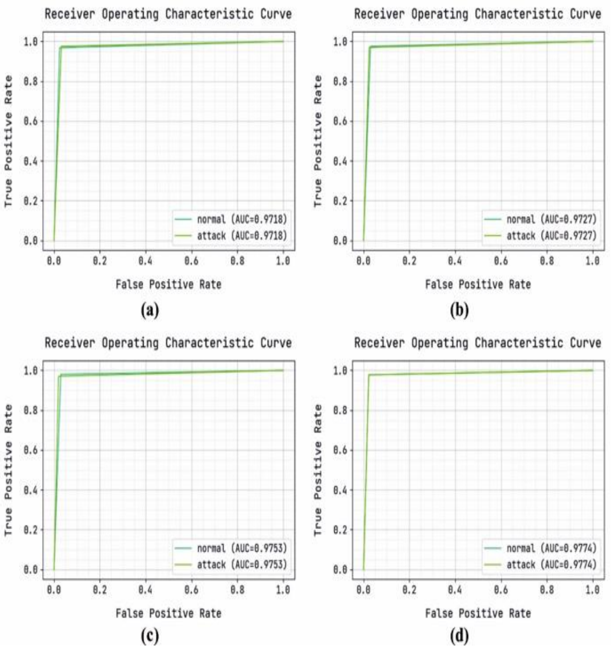


Fig. 7: ROC analysis; (a) LSTM (b); 1D-CNN (c); GRU (d); DLEM-DDoS

The results demonstrated that the LSTM model has obtained $an\ accu_y$ of 97.10, $prec_n$ of 95.13, $reca_l$ of 97.57%, $F1_{score}$ of 96.33 and ROC of 97.18%. In addition, it is found from the results that the 1D-CNN approach has reached $accu_y$ of 97.34, $prec_n$ of 96.24, $reca_l$ of 96.97%, $F1_{score}$ of 96.60 and ROC of 97.53%. Also, the results portrayed that the GRU approach has gained $accu_y$ of 97.64, $prec_n$ of 96.91, $reca_l$ of 97.03%, $F1_{score}$ of 96.97% and ROC of 97.53. However, the outcomes exhibited that the DLEM-DDoS methodology has achieved $accu_y$ of 97.74, $prec_n$ of 96.49, $reca_l$ of 97.75, $F1_{score}$ of 97.12 and ROC of 97.74%.

Table 3 and Figs. 8-9 shows the overall classification result evaluation of the DLEM-DDoS approach with other techniques in terms of different measures.

To demonstrate the improved results of the DLEM-DDoS approach, a comparison analysis is done with recent methods in Table 2.

Table 4 and Fig. 10 illustrates the $accu_y$ analysis of the DLEM-DDoS technique with existing approaches. The figure reported that the ELM-C-Means method has obtained a lesser $accu_y$ value. In line with this, the E-ELM, SaE-ELM and SaE-ELM-Ca techniques have attained slightly increased $accu_y$ values. Besides that, the V-ELM approach has offered moderately increased $accu_y$ value. Though the BRS technique has achieved close to optimal outcomes with $accu_y$ of 97.55%, the DLEM-DDoS approach has attained enhanced results with $accu_y$ of 97.74%.

Table 4 and Fig. 11 showcases the $prec_n$, $reca_l$ and F_{score} evaluation of the DLEM-DDoS technique with current techniques. The figure revealed that the ELM-C-Means approach has gained lesser $prec_n$, $reca_l$ and F_{score} values. Similarly, the E-ELM, SaE-ELM and SaE-ELM-Ca methodologies have reached slightly increased $prec_n$, $reca_l$ and F_{score} values. Besides, the V-ELM system has offered moderately higher $prec_n$, $reca_l$ and F_{score} values. Then, the BRS scheme has provided near-optimal results yielding the $prec_n$, $reca_l$ and F_{score} of 96.37, 96.37 and 96.37%, the DLEM-DDoS methodology has accomplished enhanced outcome with the $prec_n$, $reca_l$ and F_{score} of 97.75, 96.49 and 97.12%.

After examining the above-mentioned tables and figures, it can be confirmed that the DLEM-DDoS technique is capable of maximum classification performance compared to the other existing approaches in terms of various measures.

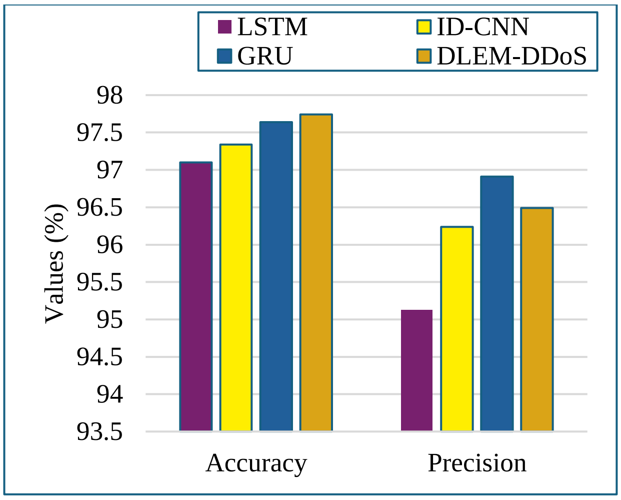


Fig. 8: Accuracy and Precision analysis of DLEM-DDoS technique

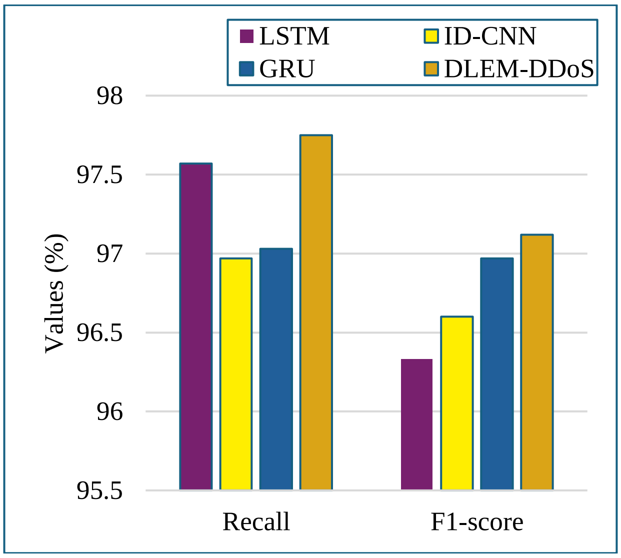


Fig. 9: Recall and F1_score analysis of DLEM- DDoS technique

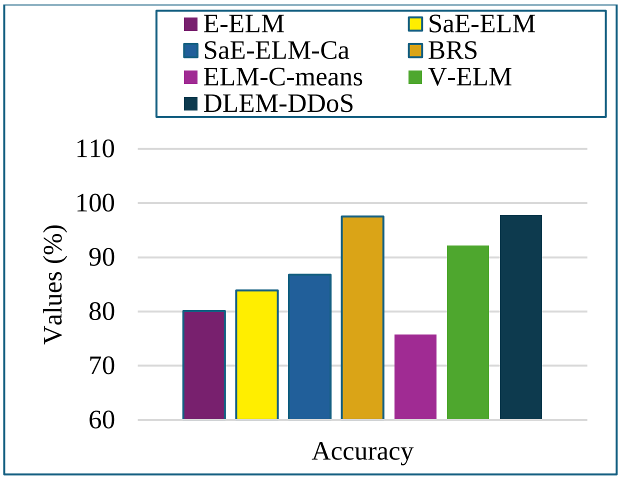


Fig. 10: Accuracy analysis of DLEM-DDoS approach with contemporary techniques

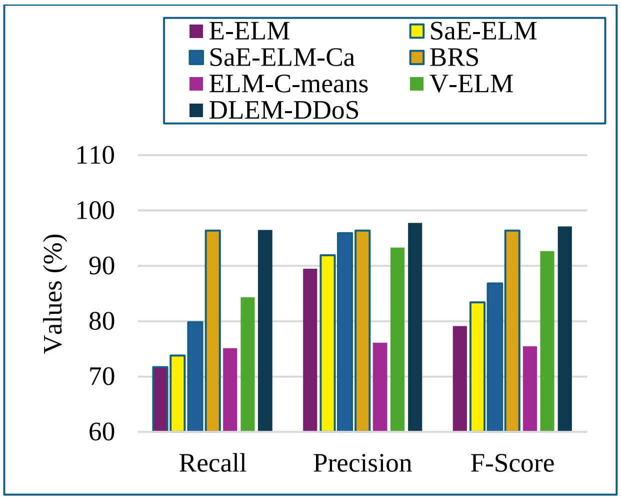


Fig. 11: Comparative analysis between DLEM-DDoS approach and common techniques

Table 3: Result analysis of DLEM-DDoS approach with different metrics

Methods	LSTM	ID-CNN	GRU	DLEM-DDoS
Accuracy	97.10	97.34	97.64	97.74
Precision	95.13	96.24	96.91	96.49
Recall	97.57	96.97	97.03	97.75
F1-score	96.33	96.60	96.97	97.12
ROC score	97.18	97.27	97.53	97.74

Table 4: Comparative analysis of the DLEM-DDoS approach with recent algorithms

Methods	Precision	Recall	Accuracy	F-Score
E-ELM	89.50	71.70	80.10	79.10
SaE-ELM	91.90	73.80	83.90	83.40
SaE-ELM-Ca	95.90	79.80	86.80	86.80
BRS	96.37	96.37	97.55	96.37
ELM-C-means	76.12	75.14	75.77	75.48
V-ELM	93.29	84.34	92.11	92.68
DLEM-DDoS	97.75	96.49	97.74	97.12

Conclusion

In this study, a unique and efficient DLEM-DDoS method has been devised for the detection and classification of DDoS attacks from the SDN-based cloud environment. The DLEM-DDoS technique gets executed at the switches existing in the data plane of the SDN framework for detecting and mitigating DDoS attacks. The proposed DLEM-DDoS technique follows a three-stage process namely pre-processing, ensemble DL-based classification and attack mitigation. To examine the heightened performance of the DLEM-DDoS approach, an extensive array of simulations were performed on the standard dataset and the results are examined under varying aspects. An extensive set of comparative result analyses reported its supremacy compared to the recent benchmark approaches in terms of several metrics. hence, the presented DLEM-DDoS mechanism can be employed as an efficient tool to achieve DDoS detection in the SDN-oriented cloud environment. As a futuristic approach, the detection efficacy of the DLEM-DDoS technique can be further increased using the design of feature selection approaches. It is essential to come up with secured cost-effective computing resources for the health care/banking and other critical service industries. Cloud is one such solution to bring cost-effective computing resources and the method proposed in this study helps to identify and mitigate the DDoS attack security problem that occurs in a cloud networking environment and thereby increases the availability of back-end servers used for health care/banking sectors without downtime.

Funding Information

The authors have not received any financial support or funding to report.

Author’s Contributions

S. Annie Christila: Study, Conceptualization, conducted all the experiments, data analysis, validation and contributed to writing the manuscript.

R. Sivakumar: Checked the experiments done, Reviewed and revised the data analysis and manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all the other authors have read and approved the manuscript and that no ethical issues are involved.

Conflicts of Interest

The authors declare that they do not have any conflicts of interest.

References

Agrawal, N., & Tapaswi, S. (2021). An SDN-assisted defense mechanism for the shrew DDoS attack in a cloud computing environment. *Journal of Network and Systems Management*, *29*(2), 12. https://doi.org/10.1007/s10922-020-09580-7

Banikazemi, M., Olshefski, D., Shaikh, A., Tracey, J., & Wang, G. (2013). Meridian: an SDN platform for cloud network services. *IEEE Communications Magazine*, *51*(2), 120-127. https://doi.org/10.1109/MCOM.2013.6461196

Bhushan, K., & Gupta, B. B. (2019). Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *Journal of Ambient Intelligence and Humanized Computing*, *10*, 1985-1997. https://doi.org/10.1007/s12652-018-0800-9

Chen, T., Guo, L., Duan, A., Gao, H., Feng, T., & He, Y. (2022). A feature learning-based method for impact load reconstruction and localization of the plate-rib assembled structure. *Structural Health Monitoring*, *21*(4), 1590-1607. https://doi.org/10.1177/14759217211038

Chen, Z., Jiang, F., Cheng, Y., Gu, X., Liu, W., & Peng, J. (2018, January). XGBoost classifier for DDoS attack detection and analysis in SDN-based cloud. In 2018 IEEE international conference on big data and smart computing (bigcomp) (pp. 251-256). IEEE. https://doi.org/10.1007/s12652-018-0800-9

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*. https://doi.org/10.48550/arXiv.1412.3555

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Jeong, K., Kim, J., & Kim, Y. T. (2012, April). QoS-aware network operating system for software defined networking with generalized OpenFlows. In *2012 IEEE Network Operations and Management Symposium* (pp. 1167-1174). IEEE. https://doi.org/10.1109/NOMS.2012.6212044

Jing, G. (2017). Research on application of DDos attack detection technology based on software defined network. *Acta Tech CSAV*, *62*(1B), 489-498.

Kushwah, G. S., & Ranga, V. (2021). Optimized extreme learning machine for detecting DDoS attacks in cloud computing. *Computers & Security*, *105*, 102260. https://doi.org/https://doi.org/10.1016/j.cose.2021.102260

Lin, Y. D., Pitt, D., Hausheer, D., Johnson, E., & Lin, Y. B. (2014). Software-defined networking: Standardization for cloud computing's second wave. *Computer*, *47*(11), 19-21. https://doi.org/10.1109/MC.2014.329

Mayoral, A., Vilalta, R., Muñoz, R., Casellas, R., & Martínez, R. (2017). SDN orchestration architectures and their integration with cloud computing applications. *Optical Switching and Networking*, *26*, 2-13. https://doi.org/10.1016/j.osn.2015.09.007

Mousavi, S. M., & St-Hilaire, M. (2018). Early detection of DDoS attacks against software defined network controllers. *Journal of Network and Systems Management*, *26*, 573-591. https://doi.org/10.1007/s10922-017-9432-1

Mukhopadhyay, N. D., & Singal, G. (2020). Ddos attack sdn dataset. *Mendeley Data*, *1*. https://data.mendeley.com/datasets/jxpfjc64kr/1

Phan, T. V., & Park, M. (2019). Efficient distributed denial-of-service attack defense in SDN-based cloud. *IEEE Access*, *7*, 18701-18714. https://doi.org/10.1109/ACCESS.2019.2896783

Pillutla, H., & Arjunan, A. (2019). Fuzzy self organizing maps-based DDoS mitigation mechanism for software defined networking in cloud computing. *Journal of Ambient Intelligence and Humanized Computing*, *10*, 1547-1559. https://doi.org/10.1007/s12652-018-0754-y

Pogiatzis, A., & Samakovitis, G. (2020). Using bilstm networks for context-aware deep sensitivity labelling on conversational data. *Applied Sciences*, *10*(24), 8924. https://doi.org/10.3390/app10248924

Sahay, R., Meng, W., & Jensen, C. D. (2019). The application of software defined networking on securing computer networks: A survey. *Journal of Network and Computer Applications*, *131*, 89-108. https://doi.org/10.1016/j.jnca.2019.01.019

Tan, L., Pan, Y., Wu, J., Zhou, J., Jiang, H., & Deng, Y. (2020). A new framework for DDoS attack detection and defense in SDN environment. *IEEE Access*, *8*, 161908-161919. 10.1109/ACCESS.2020.3021435

Virupakshar, K. B., Asundi, M., Channal, K., Shettar, P., Patil, S., & Narayan, D. G. (2020). Distributed Denial of Service (DDoS) attacks detection system for OpenStack-based private cloud. *Procedia Computer Science*, *167*, 2297-2307. https://doi.org/10.1016/j.procs.2020.03.282

Xie, J., Zhang, H., Liu, L., Li, M., & Su, Y. (2021). Decomposition-Based Multistep Sea Wind Speed Forecasting Using Stacked Gated Recurrent Unit Improved by Residual Connections. *Complexity*, *2021*(1), 2727218. https://doi.org/10.1155/2021/2727218

Yan, Q., Yu, F. R., Gong, Q., & Li, J. (2015). Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues and challenges. *IEEE Communications Surveys & Tutorials*, *18*(1), 602-622. https://doi.org/10.1109/COMST.2015.2487361