# Detecting DDoS Attacks in Software Defined Networks Using Deep Learning Techniques: A Survey

Ntumpha P. Mwanza and Jugal Kalita
*(Corresponding author: Ntumpha P. Mwanza)*

Department of Computer Science, University of Colorado Colorado Springs
1420 Austin Bluffs Pkwy, Colorado Springs, CO 80918, USA
Email: pmwanza@uccs.edu

*(Received June 17, 2022; Revised and Accepted Feb. 3, 2023; First Online Feb. 28, 2023)*

## Abstract

Deep Learning (DL) is increasingly being used in Software Defined Networks (SDNs) to detect Distributed Denial of Service (DDoS) attacks because of high attack detection accuracy. This paper presents a survey on the types of deep learning techniques used to detect DDoS attacks in SDNs. Attack statistics show that DDoS attacks are on an increase. Some of the factors that have contributed to the increase in DDoS attacks is the inability of current techniques to detect unknown DDoS attacks, which can be referred to as zero-day attacks. In this work, we look at deep learning techniques and how they are used to detect DDoS attacks. The current techniques' weaknesses are discussed and recommendations are made.

*keywords: Deep Learning; Machine Learning; Software Defined Network; Traffic Classification*

## 1 Introduction

Recent years have seen a sharp increase in internet traffic and at the same time a significant decrease in the use of network physical infrastructure. Network physical infrastructure is rapidly being replaced by the increased use of smart technology and the cloud as infrastructure. Smart technology has led to an increase in the use of smart IoT devices, connected to the internet. Virtualization has also contributed to an increase in internet traffic [49, 81]. Traditional networks are hardware-based, and for them to be used with smart technologies, there needs to be more hardware infrastructure in place to function effectively. For this reason, Software Defined Networks were introduced.

Software Defined Networks (SDN) were introduced because of the enormous increase in network connectivity and exposure to vulnerabilities. Because SDN is software-based, network management as well as its performance usually improves [68]. The SDN is made up of three planes, the Application Plane, the Control Plane, and the Data Plane [34]. The control plane is responsible for all the activities of the SDN. It has a centralized operational architecture, which makes managing network resources easy. The centralized architecture has made the SDN vulnerable to security risks and threats of attacks [67]. One common attack it is exposed to is the Distributed Denial of Service (DDoS) attack [41]. DDoS attacks attack the controller of the SDN. The controller controls and manages how the SDN operates and functions. A DDoS attack sends continuous requests to the SDN controller overwhelms the controller and denies legitimate traffic requests, which do not get any response from the network resources [15].

SDNs are widely used today because of their ability to separate the control plane from the data plane. The control plane and the data plane were separated because of increasing network traffic and the need to have a reliable network with high performance. The control plane is responsible for routing and network management. It operates by making a single application program able to control multiple programs. The Data Plane is responsible for forwarding programmable packets such as OpenFlow [94]. OpenFlow protocols let a server tell network switches where to route packets. The SDN controller can collect network data because it has an overall view of the network, making it easy to facilitate applications such as machine learning algorithms to be implemented in the controller [75]. Machine learning algorithms can be used to perform traffic analysis and improve traffic classification [43, 55].

This paper summarizes recent developments in detecting DDoS attacks in SDN using deep learning techniques. The SDN architecture is presented in Section 2, related surveys are presented in Section 3. Section 4 presents DD0S attacks. Deep and other machine learning techniques are presented in Section 5. Section 6 presents detecting DDoS attacks in SDN using deep learning tech-

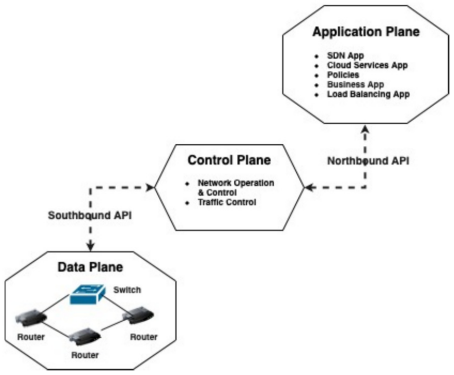


Figure 1: SDN Architecture

niques, Section 7 presents research issues and challenges and Section 8 is the conclusion.

# 2 SDN Architecture

Software Defined Networks break down the network into smaller disjoint parts. An SDN divides the functions of traditional networks into different parts and configures the parts according to functionality. Figure 1 shows the SDN architecture. Changes can be made instantly to the network functions. The SDN performs load balancing [11]. For example, if one area of the network is overwhelmed with data packets, the SDN routes the packets to where there is enough capacity. SDN uses one central point of operation, which is called the Control Plane to manage the entire network [1].

Each function of the SDN is programmed to operate automatically [8]. The SDN can allocate resources where they are needed the most on the network. The SDN has real-time centralized control of the network, which improves network performance and enhances the optimization of network function [70]. The other benefit of the SDN is virtualization [14]. Virtualization makes it possible to access both virtual and physical elements from one location. Virtualization allows multiple virtual networks to share resources from the same infrastructure and a virtual network has a simpler topology then the physical network.

## 2.1 Data Plane

Virtual switches and physical switches are found on the data plane [87]. Virtual switches communicate with other virtual machines by using software programs. Virtual switches are also referred to as software switches. Switches forward, drop, and modify packets received from the control plane [91]. The data and control planes communicate using network interfaces.

## 2.2 Control Plane

The control plane controls the entire operation of the network in a systematic and coordinated way [95]. It makes decisions and controls all operations through the SDN controller [7]. It provides control functionality using Application Programming Interfaces (APIs) to monitor the networks using an open interface. APIs are software interfaces that allow two or more applications to communicate with each other. The controller interacts with the data plane using the southbound API; the southbound APIs are used to communicate between the SDN controller and the switches and routers of the network [44]. The control plane updates forwarding rules that help with network management [93]. The southbound API facilitates the communication between the data plane and the control plane through the switches.

## 2.3 Application Plane

The responsibility of the application plane is to host applications that instruct the controller to perform changes depending on the requirements of its northbound APIs [47]. Applications such as end-user business applications, network virtualization, mobility management, and security applications are found on the application plane [24]. The application plane performs optimization and network management [76]. Depending on the network information and business requirements, the application plane can implement control logic, which is responsible for modifying network behavior.

# 3 Related Surveys

In this section, we present a review of published survey papers. SDN and Machine Learning (ML) techniques have improved the way DDoS attacks are detected and classified in SDN. The SDN controller provides centralized control and management, allocates resources, and directs network traffic. In a DDoS attack, the attacker attacks the controller so that network resources become inaccessible. Table 1 shows a summary of related survey papers on DDoS attacks in SDN. Table 1 is divided into two subsections; Subsection A is Traditional ML algorithms, and subsection B is SDN Techniques. Subsection A discusses published papers on how Traditional ML algorithms are applied in SDN to detect DDoS attacks. Subsection B discusses published papers on how SDN techniques are applied in SDN to detect DDoS attacks. SDN techniques protect the SDN by continuously monitoring network traffic. If malicious traffic is detected, the SDN controller takes action by applying the techniques to firewalls by updating the firewall rules to allow or block the traffic.

Table 1: Summary of related surveys on DDoS attacks in SDN

| Publication | Year | Technique | Focused Area | Classification of DDoS | Research Recommendations |
|---|---|---|---|---|---|
| Zhao *et al.* [96] | 2019 | ML Algorithms | Network Security | No | ✓ |
| Yan *et al.* [92] | 2018 | ML Algorithms | Network Traffic Classification | No | ✓ |
| Nguyen *et al.* [60] | 2018 | ML Algorithms | Network Security Applications | No | × |
| Ahmad *et al.* [3] | 2020 | ML Algorithms | Network Security | No | ✓ |
| Sultana *et al.* [80] | 2019 | ML Algorithms | NIDS Security | No | ✓ |
| Gebremariam *et al.* [31] | 2019 | ML Algorithms | Network Security | No | × |
| Alamri *et al.* [5] | 2021 | ML Algorithms | Network Traffic Classification | No | ✓ |
| Sahoo *et al.* [72] | 2018 | ML Algorithms | Network Traffic Classification | No | × |
| Singh *et al.* [77] | 2020 | ML Algorithms | SDN Security | No | ✓ |
| Da Costa *et al.* [17] | 2019 | SDN Techniques | IoT Network Security | No | × |
| Dharmadhikari *et al.* [20] | 2019 | SDN Techniques | Network Security | No | × |
| Dantas *et al.* [18] | 2020 | SDN Techniques | IoT Security | No | × |
| Pajila *et al.* [12] | 2019 | SDN Techniques | IoT Security | No | × |
| Eliyan *et al.* [26] | 2021 | SDN Techniques | Network Security | No | × |
| Fajar *et al.* [28] | 2018 | SDN Techniques | Network Security | No | ✓ |
| Aladaileh *et al.* [4] | 2020 | SDN Techniques | Control plane Security | No | ✓ |
| Dong *et al.* [23] | 2019 | SDN Techniques | Cloud Security | No | ✓ |
| Ubale *et al.* [84] | 2020 | SDN Techniques | Network Security | No | × |
| Herrera *et al.* [9] | 2019 | SDN Techniques | Network Security | No | ✓ |
| Sahay *et al.* [71] | 2019 | SDN Techniques | Network Security | No | × |
| Our Survey | 2022 | Deep Learning | SDN Security | Yes | ✓ |

### 3.1 Traditional Machine Learning Algorithms

Zhao *et al.* [96] discussed using ML algorithms in the context of SDNs in two ways. In the first approach, they used ML algorithms in SDN to classify network traffic. In the second approach, they used ML algorithms with network applications in SDN to classify network traffic. They compared the two approaches for performance and accuracy. They concluded that more has to be done to improve ML algorithms' ability to classify the network traffic in SDNs. Yan *et al.* [92] discussed new research on traffic classification technologies in SDNs. They analyzed challenges in traffic classification in SDNs and made recommendations.

Nguyen *et al.* [60] discussed the landscape of ML-enabled security intrusion detection in SDNs. They analyzed the vulnerabilities and attack methods and concluded by developing a new ML-based SDN security mechanism. Ahmad *et al.* [3] discussed evaluating traditional ML algorithms such as SVM and Logistic Regression to counter DoS and DDoS attacks in SDNs. Results showed that SVM produced the best results against all the other traditional machine learning algorithms used.

Sultana *et al.* [80] discussed traditional machine learning algorithms that leverage SDNs to implement Network Intrusion Detection Systems (NIDSs). They outlined various intrusion detection mechanisms using Deep Learning approaches. They used the SDN as the platform to carry out the analysis. They concluded that more needs to be done to be able to monitor real-time intrusion detection systems in high-speed networks. Gebremariam *et al.* [31] discussed traditional ML algorithms used for different applications such as network planning, management, and security in SDN and Network functions virtualization (NFV) environments. NFV is the replacement of network appliance hardware with virtual machines [13]. They concluded by laying out the challenges of detecting DDoS attacks in SDN and NFV using traditional ML algorithms.

Alamri *et al.* [5] reviewed and compared traditional ML algorithms to detect DDoS attacks in SDN. They evaluated NB, K-NN, SVM, DT, RF, and XGBoost algorithms based on accuracy, precision, recall, and f1-score. Results showed that XGBoost had the best overall performance. Sahoo *et al.* [72] discussed using traditional ML algorithms to detect DDoS attacks in SDN. They compared KNN, NB, SVM, RF, and LR algorithms for performance based on prediction and classification accuracy. Results showed that LR had the best prediction and classification accuracy compared to the other algorithms.

Singh *et al.* [77] discussed the SDN architecture and its ability to protect itself against DDoS attacks. The authors reviewed over 70 published publications in this area. Results from the review showed that 47% of the approaches are theory-based, 42% used traditional machine learning-based and 20% used artificial neural network-based.

### 3.2 Software Defined Network Techniques

Da Costa *et al.* [17] discussed providing security to network infrastructure using ML techniques. The ML techniques were used to enhance the Internet of Things (IoT) and an Intrusion Detection System (IDS). Results showed challenges in fully securing IoT and IDS systems. Dharmadhikari *et al.* [20] discussed and summarized DDoS attacks on SDNs and how they are detected and mitigated. The authors made a comparison using past and present studies and recommended the need to have strong mitigation techniques in place. They also acknowledged that DDoS attacks cannot be fully prevented.

Dantas *et al.* [18] discussed the need to come up with new virtual techniques to prevent DDoS attacks in SDNs in an IoT environment. The authors explained that when a technique is applied based on a scenario in the IoT environment, justification has to be given regarding its suit-

ability. A summary was provided based on the strengths and weaknesses of the techniques to detect DDoS attacks in an IoT environment.

Pajila *et al.* [12] discussed the growth and security of IoT systems. It is heterogeneous in design, and the way it operates through the internet exposes it to DDoS attacks. The authors concluded that there is a gap in modeling platforms, such as not having context-based security, and clients controlling access. Eliyan *et al.* [26] discussed two countermeasures that can be used for detecting, mitigating, and preventing DoS and DDoS attacks in SDNs. The two approaches are Intrinsic and Extrinsic approaches. The intrinsic approach is applied to SDN components and their functionalities. The extrinsic approach is applied to the network traffic flow and feature characteristics in SDNs. They concluded that more research had to be performed to improve the detection accuracy of DoS and DDoS attacks in SDNs.

Fajar *et al.* [28] discussed security vulnerabilities in the SDN controller. They concluded that the current defense mechanisms are not effective against DDoS attacks. Aladaileh *et al.* [4] discussed techniques used to detect DDoS attacks in SDNs. They explained the important role the SDN controller plays in protecting the SDNs. They further gave a detailed summary of the state-of-the-art and made future research recommendations such as combining different DDoS attack patterns, to create a more complex and effective defense technique.

Dong *et al.* [23] discussed DDoS attacks in both SDNs and the Cloud, along with a summary of DDoS attacks in SDNs and how they are detected and prevented. The authors recommended using traffic classification models to improve DDoS attack detection. Ubale *et al.* [84] discussed an SDN's ability to prevent DDoS attacks because of architectural design. The authors gave specific types of DDoS attacks that the SDN is unable to prevent and recommended future research to help thwart vulnerabilities against DDoS attacks.

Herrera *et al.* [9] discussed security concerns of SDNs by comparing different studies already concluded in this area by universities and the cybersecurity industry. The authors focused on security concerns such as the effectiveness of the current countermeasures used to detect DDoS attacks in SDN. They recommended more research to address the security concerns of the SDN.

Sahay *et al.* [71] discussed the benefits of SDNs in providing network security compared to traditional networks. The centralized architecture of the SDN controller makes it easy to dynamically configure the network, and also makes it easy to identify and mitigate DDoS attacks. The controller can analyze the entire network traffic in real time because of its global view of the network. The authors recommended more research on SDNs and applications situated in SDNs. They recommended that new SDN applications pay attention to network security.

# 4 DDoS Attacks

DDoS attacks attack the server or network with the intention of disrupting its normal function [38]. They continuously send malicious traffic requests to the system. The system is eventually unable to respond to requests and stops working and functioning normally. The system becomes unable to respond to requests coming from legitimate sources or users [73]. Legitimate users are trying to order books from Amazon.com website; as they browse through the website looking for specific books, the requested website is sending requests to the server database [74]. At the same time, the attackers are also sending multiple fake malicious requests to the server as well. The attackers' continuous requests coming from different sources overwhelm the system and use up its network bandwidth [10]. This causes the legitimate users' computers to be denied service because the server has been overwhelmed with fake malicious requests. Figure 2 presents a taxonomy of the types of DDoS attacks, and Figure 3 illustrates a DDoS attack.

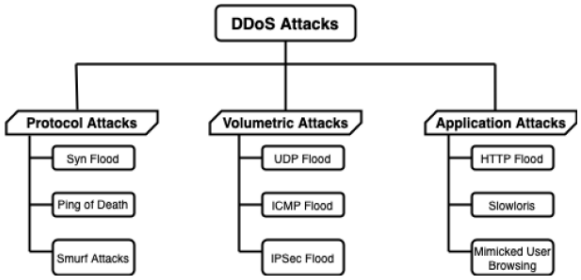


Figure 2: Types of DDoS attacks

## 4.1 Types of DDoS Attacks

### 4.1.1 Volume-Based Attacks

Volume-based attacks are DDoS attacks that aim to deplete the bandwidth of a network system [42]. The attackers use bots to amplify the attack by spreading malware which is transmitted through the bots and is spread through network traffic. Examples of volumetric-based attacks are UDP flood, ICMP flood, and IPSec flood attacks.

### 4.1.2 Protocol Attacks

Protocol attacks are DDoS attacks that prevent a legitimate user's computer from establishing a connection with the host computer [53]. This attack uses the 3-way handshake, SYN, SYN-ACK, and ACK, to carry out an attack. The legitimate user client computer sends a request (SYN), to the host computer. The host computer responds to the client computer accepting to establish a connection (SYN-ACK). Once the client computer receives the acknowledgment to establish a connection, it

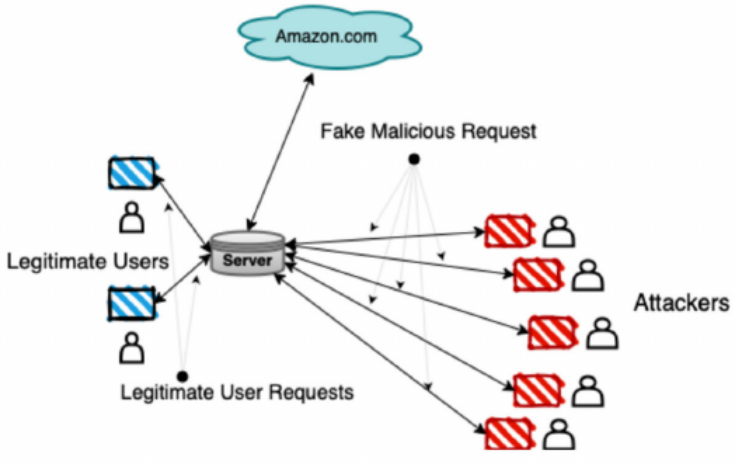


Figure 3: Distributed denial of service attack

sends back an acknowledgment (ACK), and then the connection is established. What the protocol attack does is when the client computer sends a request to the host computer to establish a connection, the attackers send continuous SYN requests, to the host computer at the same time the client computer sends the request to establish a connection. The aim is to exhaust the system so that it is unable to respond and establish a connection with the client's computer. Syn flood, Ping of Death, and Smurf DDoS attacks are examples of protocol attacks.

#### 4.1.3 Application Plane Attacks

Application plane attacks attack the applications that make networks function properly [51]. They attack the applications by taking advantage of security flaws in the applications to carry out an attack. The application becomes unable to communicate with other applications and users. HTTP Flood, Slowloris, and Mimicked User Browsing attacks are an example of application plane attacks [32].

## 5 Deep and other Machine Learning Techniques

Attackers are taking advantage of the growing number of IoT devices connected to the internet, and the increasing amount of network traffic to the internet to launch DDoS attacks. Attackers are using more complex and sophisticated attack methods to carry out these DDoS attacks, which are difficult to detect. Because of the large amount of labeled data used to carry out these attacks, deep learning-based detection techniques may be the best techniques to use to detect DDoS attacks. DL techniques produce the best detection rate and classification accuracy when a large amount of labeled data is available. Compared with DL techniques, traditional ML techniques may not produce as high accuracy in detecting DDoS attacks.

DL techniques require a large amount of labeled data input to produce the desired accuracy. The data used for training need to be correctly labeled for the trained model to correctly classify unseen examples. If the input data are not correctly labeled, the model is unlikely to produce the correct classification. Most DDoS attacks are unknown attacks (zero-day attacks); this means the training dataset does not contain any similar labeled examples. DL techniques are the best techniques to detect DDoS attacks because:

- DL techniques are excellent, during training at discovering useful hidden features in the data. A trained DL network can extract features from previously unseen examples, and classify such examples well.
- Some DL techniques can learn long-term dependencies of temporal patterns.

Below we present the taxonomy of commonly used DL techniques for detecting DDoS attacks in SDN in Figure 4 and we discuss the techniques. We end the section with a short discussion of relevant traditional machine learning approaches also since many of the papers presented in this survey refer to such approaches in addition to DL methods.

### 5.1 Discriminative Learning Techniques

Discriminative Learning Techniques are techniques that learn the boundaries between classes in a dataset. The techniques use probability estimates and maximum likelihood to create new instances, in order to find the boundary separating one class from the other.

#### 5.1.1 Multilayer Perceptron

Multilayer Perceptron (MLP) are neural networks that are made up of one or more densely connected hidden layers between the input and output layers. Figure 5 shows the architecture of MLP. MLPs are trained by adjusting

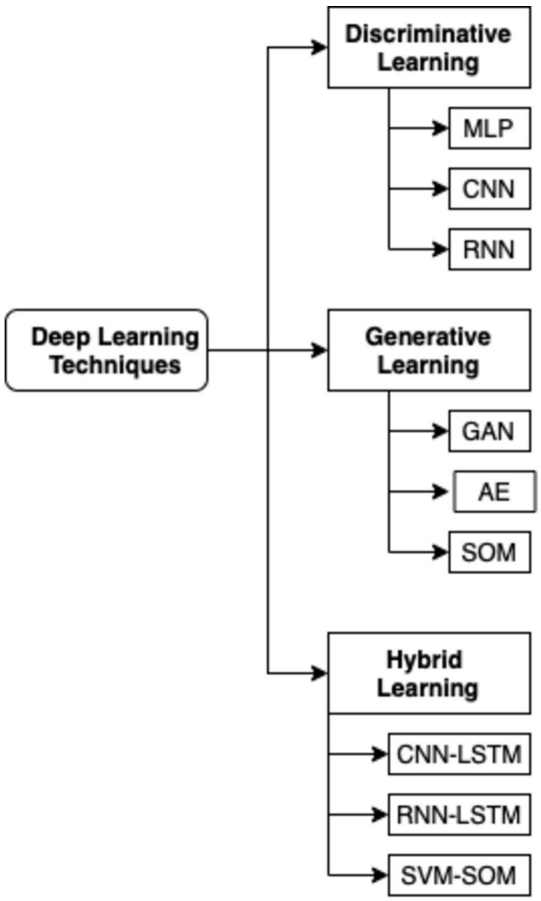


Figure 4: Taxonomy of detection methods for DDoS attacks in SDN using deep learning techniques

the weights of each connection after it is shown a dataset of labeled examples one by one [82]. The error or loss between the expected result and the output of the MLP determines the adjustments to be made to the weights. This process continues until the loss is reduced to a level that does not change the outcome.

#### 5.1.2 Convolutional Neural Network

Convolutional Neural Networks (CNNs) are primarily used for image classification and object detection [37]. CNNs have strong extraction capabilities that are used to automatically extract useful features from input data. The input data is passed through different layers of the CNN for feature extraction. As data move from one layer to the other, features are extracted at various levels of abstraction [30]. Figure 6 shows an example of CNN architecture.

#### 5.1.3 Recurrent Neural Network

Recurrent Neural Networks (RNNs) are used in Natural Language Processing (NLP) [57]. The input at a time step is processed and produces an output. Then the output of this step is processed together with the input to the next step. This allows the RNN to remember the inputs in the previous steps in a sequence [36]. The output of RNN at a certain step is dependent on previous input elements in a sequence as illustrated in Figure 7. In this figure, A is the input layer, B is the hidden layer with a recurrent loop, and C is the output layer. X, Y, and Z are network parameters used to improve the output of the model.

### 5.2 Generative Learning Techniques

Generative Learning Techniques are techniques that focus on the distribution of individual classes in a dataset. The technique uses likelihood and probability estimates to model data points and differentiates between class labels in a dataset. The technique uses joint probability, by creating instances where a given feature input (a) and the desired output (b) exist at the same time.

#### 5.2.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) consist of two models, a Generator and a Discriminator [2]. The generator creates fake samples, which are then used to fool the discriminator [52]. The discriminator is trained on real as well as generated fake samples, and learns to distinguish between real and fake samples well. After the discriminator has determined whether a sample is real or fake, the result is sent back to the generator which is trained to generate better fake images. The generator is always trained with real samples. Figure 8 shows the GAN architecture.

#### 5.2.2 Autoencoders

Autoencoders (AEs) are unsupervised neural networks [86]. They consist of two parts, the Encoder, and the Decoder. The encoder takes the input and learns how to compress and encode the data into a code. Then the decoder learns how to reconstruct the encoded data representation to create output [54]. The output is similar to the original input. The difference between the input and output is the input contains signals that have noise while the output has no noise, the signal has been denoised, as shown in Figure 9.

#### 5.2.3 Self-Organizing Maps

Self-Organizing Maps (SOMs) are shallow neural networks that are trained on unlabelled data [35]. They are used for clustering high-dimensional inputs to easily visualize the two-dimensional output. A SOM has two layers, the input layer, and the output layer connected by edges with weights [90]. The weights determine the specific location of each neuron in the two-dimensional space. Weights are trained and updated to change the position of neurons into clusters that we can easily see. They can also be used to combine diverse datasets to find patterns. Figure 10 shows the SOM architecture.

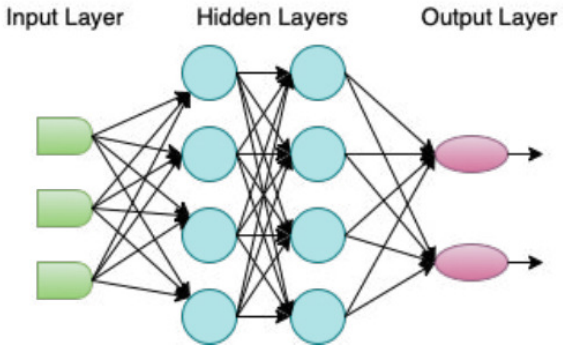


Figure 5: Multilayer perceptron

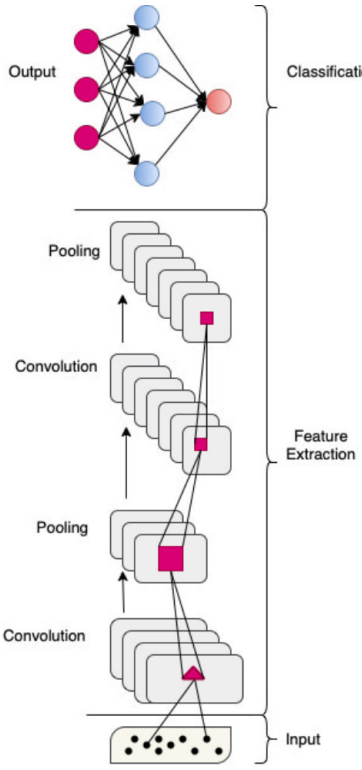


Figure 6: Convolutional neural network

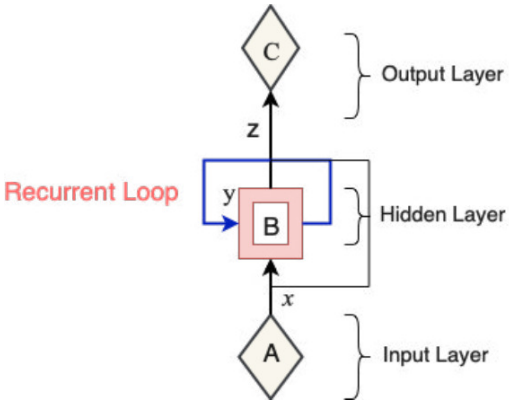


Figure 7: Recurrent neural networks

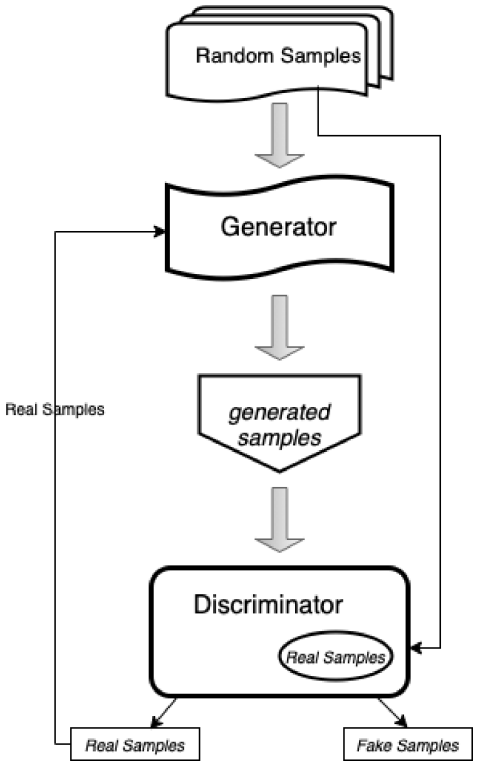


Figure 8: Generative adversarial networks

## 5.3 Hybrid Learning Techniques

Hybrid learning techniques are techniques that are comprised of a combination of two or more deep learning models, such as discriminative or generative deep learning models. The combination of these models can be used to extract more meaningful and robust features, depending on the target use.

### 5.3.1 Convolutional Neural Network-Long Short Term Memory

The Convolutional Neural Network-Long Short Term Memory (CNN-LSTM) model has been used for sequence prediction with spatial inputs. Figure 11 shows the CNN-

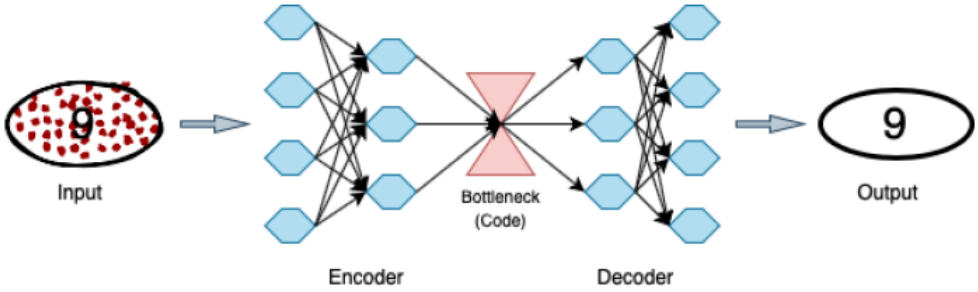


Figure 9: Autoencoders

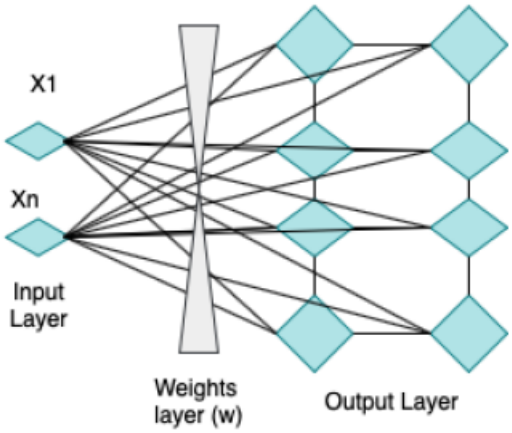


Figure 10: Self-organizing maps

LSTM architecture [46, 50]. The input are images that are fed into the CNN. The CNN extracts features from the image and feeds them to the [40]. Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning long-term dependencies. Traditional Recurrent Neural Networks are incapable of learning to detect long-term dependencies, and LSTMs, were introduced to fix this problem.

## 5.4 Reinforcement Learning Technique

Reinforcement Learning is a machine learning technique that involves an agent learning how to interact with its environment by performing actions and seeing the results of actions [22]. The agent learns the best action to maximize its long-term reward. The agent gets positive feedback for each good action and negative feedback or a penalty for each bad action. The agent learns from the feedback and experience from its environment without depending on any labeled data [89].

## 5.5 Traditional Machine Learning Technique

Traditional machine learning methods use computational, statistical, and mathematical methods to deploy algorithms that extract patterns out of raw data on input. They automatically learn from the data and past experiences and be able to make a prediction. Traditional machine learning classification algorithms usually can perform well when trained with a small amount of data compared to DL. However, DL approaches usually produce better accuracy, assuming a large amount of labeled data is available.

### 5.5.1 Support Vector Machine

Support Vector Machines (SVMs) are supervised learning algorithms that have been widely used for solving complex classification and regression problems [65]. SVMs can perform data transformations that can be leveraged to determine boundaries between data classes when trained on examples from predefined classes. Given a set of high-dimensional data points or vectors in a vector space, it looks for the separating hyperplane that separates the vector space into subspaces containing sub-sets of vectors. Each sub-set corresponds to one class. Assuming binary classification, the separating hyperplane maximizes the margin between the two subspaces. Classification can be performed by finding the hyperplane that differentiates the two classes very well.

### 5.5.2 Decision Tree

A Decision Tree (DT) is a supervised machine learning technique that produces a tree-like structured classifier in which data is repeatedly divided at each row based on certain rules until the outcome is generated [64]. They are used for solving both classification and regression problems. A DT has two types of nodes, Decision Nodes, and Leaf Nodes. The decision nodes are used to make decisions and the leaf nodes are the output of those decisions and do not add any more branches.

### 5.5.3 Random Forest

Random Forest (RF) is a supervised machine learning technique that uses an ensemble of decision trees for both classification and regression. The forest consists of a number of decision trees created by sampling training instances and sampling attributes of training instances. Each tree individually classifies an unseen instance, and the classification with the most votes is selected [79].

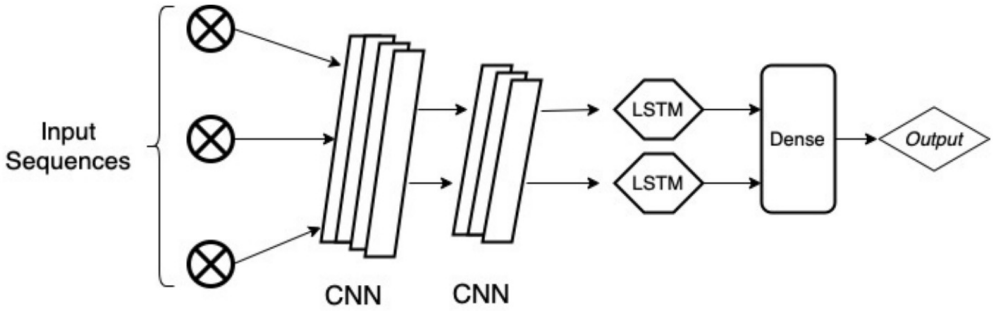


Figure 11: Convolutional neural network-long short-term memory

#### 5.5.4 Naive Bayes

Naïve Bayes (NB) is a supervised learning technique used for classification and applies the Bayes theorem, with the assumption that all features of the data instances are independent of each other. The features independently contribute to making a probability of a label given the observed features [16].

#### 5.5.5 Logistic Regression

Logistic Regression (LR) is a supervised learning technique that is used to predict the probability of the occurrence of a class with the help of independent variables or features. There are only two outcomes or classes, 1 or 0, true or false [98].

## 6 Detecting DDoS Attacks in SDN Using Deep Learning Techniques

An SDN uses the controller to control the entire network's functions by intelligently allocating and prioritizing network resources. The advantages of SDNs are that they can work without human intervention, can be programmed to make decisions, can allocate network resources, and can route traffic to the right destination within the SDN. The SDN controller is responsible for the security of the SDN. Unfortunately, the SDN controller itself is vulnerable to DDoS attacks.

The centralized architecture of the SDN exposes it to DDoS attacks and is regarded as a single source of failure. An SDN itself is unable to determine whether the network traffic is normal or anomalous, which makes it difficult for the SDN to detect and prevent DDoS attacks. This vulnerability has led to the introduction of Deep Learning for the detection of DDoS attacks. Deep learning intelligently learns data flow features in the network traffic and classifies them as either normal or anomalous.

Below we discuss published papers on deep learning approaches used to detect DDoS attacks in SDN. The section is divided into Discriminative, Generative, and Hybrid Learning approaches. Table 2 is a summary of published papers on approaches to Detect DDoS Attacks in SDNs. The tables is divided into approaches, year the papers were published, the techniques used and the datasets used.

### 6.1 Discriminative Learning Techniques

Lee *et al.* [48] proposed DL Intrusion Detection and Prevention System (DL-IDPS) to detect and prevent DDoS and brute force attacks in SDN. They evaluated the performance of the proposed system with MLP, CNN, and LSTM models. Results showed that the system produced the best performance with an accuracy of 99% detecting brute force attacks and 100% accuracy detecting DDoS attacks.

Wang *et al.* [88] proposed an SDN-Home Gateway (HGW) framework that improves the SDN controller management of smart devices connected to the network. The SDN-HGW can control end-to-end network management. But SDN-HGW cannot carry out real-time encrypted packet inspection, which puts the network at risk of DDoS attacks. To overcome this risk, the authors proposed a classifier called DataNet, developed based on MLP and CNN models. DataNet can detect and classify encrypted network packets in real-time. Results from an evaluation showed that DataNet had a detection and classification accuracy of 98%.

Narayanadoss *et al.* [59] proposed a DL model that relies on SDN traffic to get information about the flow size and timestamp measurements. They compared 3 techniques, MLP, CNN, and LSTM, to determine how many correlations are present in the traffic flow from compromised nodes. Results showed that all the models achieved above 80% detection rate of compromised nodes and LSTM had the best detection rate of 87%.

Janabi *et al.* [39] proposed a DL Early Warning Proactive System (DL-EWPS) predict network attacks in SDNs

Table 2: Summary of approaches to detect DDoS attacks in SDNs

| Approach | Publication | Year | Technique | Dataset Used |
|---|---|---|---|---|
| Discriminative | Lee *et al.* [48] | 2020 | MPL,CNN,LSTM | - |
| | Wang *et al.* [88] | 2018 | CNN | ISCX |
| | Narayanadoss *et al.* [59] | 2019 | CNN,RNN,LSTM | Mininet-WiFi |
| | Janabi *et al.* [39] | 2022 | CNN | InSDN |
| | Haider *et al.* [33] | 2020 | CNN | CICIDS2017 |
| | Polat *et al.* [66] | 2022 | RNN | - |
| Generative | AlEroud *et al.* [6] | 2019 | GAN | DARPA |
| | Novaes *et al.* [62] | 2021 | GAN | CICDDoS 2019 |
| | Ujjan *et al.* [85] | 2020 | SAE | SM1, SM2 |
| | Meng *et al.* [56] | 2020 | SOM | - |
| Hybrid | Khan *et al.* [45] | 2021 | CNN-LSTM | IOT-23 |
| | Nugraha *et al.* [63] | 2020 | CNN-LSTM | - |
| | Ding *et al.* [21] | 2020 | Hybrid CNN | UNSW-NB15, KDDCup 99 |
| | Qin *et al.* [69] | 2019 | CNN+RNN | SIM-DATA, CTU-13 |
| | Gadze *et al.* [29] | 2021 | RNN-LSTM | - |
| | Elsayed *et al.* [27] | 2020 | RNN | CICDDoS2019 |
| | Deepa *et al.* [19] | 2019 | SVM-SOM | CAIDA 2016 |
| | Nam *et al.* [58] | 2018 | SOM | DDoS Attack 2007 |
| | Novaes *et al.* [61] | 2020 | LSTM-Fuzzy | CICDDoS 2019 |

using CNN for classification. The system converted numerical data to RGB images to improve CNN classification and added extra features from flow tables statistics. The system achieved 100% DDoS attack classification accuracy.

Haider *et al.* [33] proposed an ensemble CNN framework to detect DDoS attacks in SDN. The authors compared the ensemble CNN with ensemble RNN, ensemble LSTM, and hybrid reinforcement learning. They used the CICIDS2017 dataset which was fully labeled with 80 features of network traffic with benign and attack traffic. They used random forest regression for the feature selection of the 80 features of the network traffic. They evaluated the proposed model with other models. The proposed ensemble CNN framework achieved the best accuracy in detecting DDoS attacks with 99.45%.

Polat *et al.* [66] proposed an Recurrent Neural Network (RNN) classifier to detect DDoS attacks in SDN-based Supervising Control and Data Acquisition (SCADA) system. The RNN classifier was evaluated in with Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) models. Results showed the proposed RNN classifier had the best accuracy of 96.67% detecting attacks in SDN-based SCADA system.

## 6.2 Generative Learning Techniques

AlEroud *et al.* [6] proposed an approach that generates attacks on SDN to train the SDN to learn to detect attacks. Generative Adversarial Networks were used for adversarial training. They evaluated the approach using two scenarios; in the first scenario, GAN was not used and in the second scenario, GAN was used. Results showed the second scenario had the best performance by accurately identifying attacks in SDN when GAN was used, and the first scenario had low attack detection accuracy in identifying attacks in SDN when GAN was not used.

Novaes *et al.* [62] proposed an adversarial training system that uses Generative Adversarial Network (GAN) to detect and defend the SDN against DDoS attacks. The system was evaluated with other methods, CNN, LSTM, and MLP. The methods were tested in two separate scenarios. In the first scenario, the GAN had the best accuracy of 99.78% detecting DDoS attacks in SDN. In the second scenario, the GAN had the best accuracy of 94.38% detecting DDoS attacks in SDNs. Thus, the GAN had the best performance in both scenarios.

Ujjan *et al.* [85] proposed a sflow and Adaptive Polling band sampling with Snort IDS and Stacked Autoencoders (SAE) for detecting DDoS attacks in IoT networks. The model uses snort IDS to identify network traffic and uses Stacked Autoencoder to classify traffic as either benign or DDoS attack traffic. Snort IDS and SAE were used in both sFlow and Adaptive polling sampling. Adaptive polling sampling is an algorithm is used to refine polling intervals based on the rate of change of network traffic flow. Results showed good sflow achieved 95% DDoS attack accuracy with less than 4% false positive rate and Adaptive polling-based sampling achieved 95% DDoS attack accuracy with less than 8% false positive rate.

Meng *et al.* [56] proposed a SOM-based DDoS attack defense mechanism to detect DDoS attacks on Internet of Things devices. The mechanism uses the connection between the SDN and the Internet of Things (IoT) devices to detect DDoS attacks. When a DDoS attack is detected, the proposed mechanism blocks traffic to and from the IoT devices. Results showed the proposed mechanism accurately detected DDoS attacks in IoT devices.

### 6.3 Hybrid Learning Techniques

Khan *et al.* [45] The proposed Hybrid Deep Learning architecture comprises CNN (Convolution Neural Network) and LSTM (Long Short Term Memory) models to detect sophisticated malware attacks in the Internet of Medical Things (IoMT). The model was evaluated with Convolution Neural Network (CNN) and Gated Recurrent Units (GRU) model, and Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) model. Results showed that the CNN-LSTM model outperformed CNN-GRU and GRU-LSTM models with an accuracy of 99.83% detecting sophisticated IoMT malware.

Nugraha *et al.* [63] proposed a Hybrid CNN-LSTM model used to detect Slow DDoS attacks in SDN-based networks. Slow DDoS attacks are a type of DDoS attack that aim to disrupt services provided by an application to a network server by sending small amounts of attack data with legitimate traffic over a long period. They used this model because of its high accuracy and recall when detecting different types of DDoS attacks. The model uses the output from the feature extractor to classify. Results showed the model achieved 99% accuracy in detecting slow DDoS attacks. The model was evaluated against MLP and 1-class SVM models, and it outperformed both methods.

Ding *et al.* [21] Proposed a Hybrid Convolutional Neural Network (HYBRID-CNN) to extract deep features from the smart grid network flow that traditional ML methods connect extract. The HYBRID-CNN is a method that utilizes CNNs to effectively memorize global features by one-dimensional (1D) data and to generalize local features by two-dimensional (2D) data. Two datasets were used for the evaluation. The method was evaluated and compared with other models, LSTM and CNN-LSTM. The Hybrid-CNN had the highest performance with an accuracy of 95.64% and had the highest detection rate of 98.56%.

Qin *et al.* [69] proposed a CNN-RNN model to detect and classify anomalies in network traffic. They compared the CNN-RNN model with a Tree-Shaped deep Neural Network (TSDNN) model using two datasets, CTU-13 [78] and a self-generated dataset Sim-data dataset. CNN-RNN model had the best accuracy of 99.8% compared to TSDNN with 99.7% accuracy in detecting network attacks.

Gadze *et al.* [29] proposed using CNN and RNN-LSTM to detect DDoS attacks such as TCP, UDP, and ICMP flood attacks on the SDN Controller. They also compared the performance of the DL models with traditional ML techniques. Performance was based on accuracy, recall, true-negative rate, and time taken in detecting and mitigating DDoS attacks in the SDN controller. RNN-LSTM had the best results overall in detecting DDoS attacks in the SDN controller.

Elsayed *et al.* [27] proposed DDoSNet, an intrusion detection method to detect DDoS attacks in SDN. This method uses an RNN-AE to detect and classify benign or DDoS attack traffic on input. The model is in two stages, the unsupervised pre-training stage and the time-tuning stage. The first stage extracts useful feature representation, and the second stage trains the last layer of the network using labeled samples. The proposed method had an accuracy of 99% correctly detecting and classifying DDoS in SDN.

Deepa *et al.* [19] proposed an ensemble technique using k-Nearest Neighbors (kNN), Naïve Bayes (NB), SVM, and SOM techniques to detect DDoS attacks in SDN controller. The hybrid SVM-SOM outperformed the other models with a detecting accuracy of 98%.

Nam *et al.* [58] proposed a DDoS attack detection algorithm that uses Self Organizing Map (SOM) with other techniques such as k-Nearest Neighbors (kNN), SOM-kNN, SOM distributed neurons, and SOM distributed center to classify network traffic as normal or anomalous. The techniques' performance was evaluated based on detection rate, false-positive rate, and processing time. kNN had the best detection rate and largest processing time. SOM-kNN had the second-best detection rate and the lowest false positive rate. SOM-kNN had the best performance because it had the best DDoS attack classification rate and the lowest false positive rate.

Novaes *et al.* [61] proposed an LSTM-FUZZY model that characterizes, detects, and mitigates DDoS and Portscan attacks in SDN environments. Two scenarios were used to evaluate the model. In the first scenario, the LSTM-FUZZY model was evaluated against k-Nearest Neighbor (KNN), LSTM-2, MLP, Particle Swarm Optimization Digital Signature (PSO-DS) [25], and SVM models to detect DDoS attacks in SDN environment by applying mitigation policies based on the attack type identified by the detection module. LSTM-FUZZY had the best performance with 96.22% DDoS attack accuracy. In the second scenario, the CICDDoS 2019 dataset was used. LSTM-FUZZY achieved 99.20% DDoS attack accuracy.

## 7 Research Issues and Challenges

Attackers have improved their DDoS attack approaches and techniques. These improvements have made DDoS attacks more sophisticated and hard for most defense systems to detect. The improvements have also led to an increase in the number of DDoS attack types. Each attack type has a different attack pattern, which makes it difficult to put effective defensive systems in place. Most of these attacks are zero-day attacks, which have no defense system developed. Defense approaches are usually developed only after an attack has already materialized. Deep learning techniques have shown to be effective in the accurate detection and classification of attacks, including making zero-day defense possible. Based on the published papers we have read, we have observed the following.

- Experiments conducted show that the source and destination IP addresses were not used when extracting features and classification of the network traf-

fic. But the source and destination IP addresses are very important for analyzing and classifying traffic as either normal or anomaly. After the features are extracted, they are used to classify and categorize as either normal or anomaly traffic. Removing the source and destination IP addresses will not classify the traffic accurately.

- DL techniques are used in both virtual and physical networks to detect DDoS attacks. An experiment was performed [29] to compare the performance between RNN-LSTM, SVM, and Naive Bayes to detect DDoS attacks such as TCP, UDP, and ICMP in the SDN controller. The experiment was conducted only on virtual networks and concluded that the defense system was effective in detecting and classifying DDoS attacks accurately. The problem with this approach is that the authors did not experiment with their approach on physical networks. The types of DDoS attacks on virtual and physical networks are different. For example, DDoS attacks like the Reflection attacks can only attack physical networks, but not virtual networks. The results may have been accurate on virtual networks, but physical networks were not tested.

- Many defense systems leverage DL techniques to improve the detection and classification accuracy of DDoS attacks in SDNs. Most of these defense systems focus only on attacks that are in the network traffic, from input to output. They do not focus on attacks that attack physical devices such as switches and routers. For example, attacks that are not detected on input might attack physical devices on the network that are responsible to store or route data packets. These DDoS attacks can overwhelm the system by rerouting traffic and sending fake requests to the SDN controller. This problem can be overcome using an Intrusion Prevention System (IPS). The IPS is a piece of network security software or hardware that continuously monitors the network for threats and can automatically apply countermeasures to stop the attack or attacks by dropping packets and blocking traffic to affected hardware.

- Published literature tells us how good the defense systems that use DL techniques are in detecting DDoS attacks in SDN. Results show the proposed systems have excellent accuracy in classifying traffic as normal or as DDoS. But most of these results do not tell us what the false positive and negative rates are. With an increase of new DDoS at- tacks, it is important to be able to know how accurately the defense systems can still detect and classify traffic correctly without having high false positives and negatives. It is important to develop defense systems that will have low or no false positive and negative rates on new attacks, as the attackers are becoming good at fooling the defense system in place.

- DL techniques use datasets that contain different historical DDoS attack patterns to help train the DL models so that they can accurately detect and classify DDoS attacks that match the patterns in the dataset. This has proved to be an effective approach for detecting known attacks. But this approach is not effective for detecting zero-day DDoS attacks, because the attack pattern(s) are unknown. The problem is that historical attack patterns cannot be used to predict or discover new attacks and generate new patterns. The attack has to materialize for the attack patterns to be added to a dataset. Datasets are effective in detecting and classifying known DDoS attacks. The DL technique's accuracy is dependent on how accurate the data is in the dataset.

- Even if there is a sophisticated machine learning approach, datasets used for training may not provide all the feature patterns needed, to carry out a comprehensive analysis of network traffic so that a system can identify DDoS attacks accurately. Datasets may have limitations. One of the most important limitations is the size of datasets; a smaller dataset may not store all the necessary features that come with an attack. Datasets should be large enough to be able to contain a large amount of DDoS attack patterns and features that can be used by DL techniques training models and help the models accurately detect and classify DDoS attacks.

# 8 Deep Learning Techniques Limitations

- DL techniques need large amounts of data to learn patterns in the data. With the rising number of new DDoS attacks, the model needs to be continuously updated with the new attack data. But the new attack data may not be available to train the model on the new DDoS attack patterns.

- Using a small amount of training data in DL models such as CNNs is likely to produce low accuracy and is unable to generalize well to unseen examples. The situation may be ameliorated with transfer learning [97] and/or multi-task learning [83] but this needs further investigation.

- Good hardware support is another DL limitation. DL techniques require a good amount of hardware support because of their high computational power, which is expensive to acquire and maintain. This problem can be addressed by using a Graphical Processing Unit (GPU). GPUs are needed for training, but trained models usually run fast.

- DL techniques are unable to reassign, re-categorize and relabel previously stored data without retraining the model.

- Using poor-quality of data with errors and noise for training will prevent the DL model from detecting required patterns and the model will not perform well.

## 9 Conclusion

In this paper, we examined different published papers that use DL techniques to detect DDoS attacks in SDN. We compared three categories of DL, discriminative, generative, and hybrid learning. Results show that DL techniques have good performance in accurately detecting and classifying DDoS attacks. With the increase in internet connection traffic through smart devices, IoT plays a major role in the increase of DDoS attacks. DDoS attacks will spread by being part of the internet or network traffic. DL techniques are the best methods to detect DDoS attacks because an increase in internet traffic (training data), will make the techniques learn more robust features and increase the classifier's performance. If performance falls, the number of hidden layers can be increased to improve performance and feature classification accuracy. But DL techniques still depend on datasets for training on known attack patterns, which are stored as historical data in the dataset. Combining different DL techniques has also shown improvement in DDoS attack detection accuracy on SDN networks.

## Acknowledgments

All authors approved the version of the manuscript to be published.

## References

[1] A. Abuarqoub, "A review of the control plane scalability approaches in software defined networking," *Future Internet*, vol. 12, no. 3, pp. 49, 2020.

[2] A. Aggarwal, M. Mittal, and G. Battineni, "Generative adversarial network: An overview of theory and applications," *International Journal of Information Management Data Insights*, vol. 1, no. 1, pp. 100004, 2021.

[3] A. Ahmad, E. Harjula, M. Ylianttila, and I. Ahmad, "Evaluation of machine learning techniques for security in SDN." in *IEEE Globecom Workshops*, pp. 1–6, 2020.

[4] M. A. Aladaileh, M. Anbar, I. H. Hasbullah, Y. W. Chong, and Y. K. Sanjalawe, "Detection techniques of distributed denial of service attacks on software defined networking controller–a review," *IEEE Access*, vol. 8, pp. 143985–143995, 2020.

[5] H. A. Alamri and V. Thayananthan, "Analysis of machine learning for securing software-defined networking," *Procedia Computer Science*, vol. 194, pp. 229–236, 2021.

[6] A. AlEroud and G. Karabatis, "Sdn-gan: generative adversarial deep nns for synthesizing cyber attacks on software defined networks," in *OTM Confederated International Conferences*, pp. 211–220, 2020.

[7] J. Ali and B. Roh, "Quality of service improvement with optimal software-defined networking controller and control plane clustering," *Comput. Mater. Contin*, vol. 67, pp. 849–875, 2021.

[8] N. Anerousis, P. Chemouil, A. A. Lazar, N. Mihai, and S. B. Weinstein, "The origin and evolution of open programmable networks and SDN," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1956–1971, 2021.

[9] J. Arevalo Herrera and J. E. Camargo, "A survey on machine learning applications for software defined network security," in *International Conference on Applied Cryptography and Network Security*, pp. 70–93, 2019.

[10] M. Asad, M. Asim, T. Javed, M. O. Beg, H. Mujtaba, and S. Abbas, "Deepdetect: Detection of distributed denial of service attacks using deep learning," *The Computer Journal*, vol. 63, no. 7, pp. 983–994, 2020.

[11] M. R. Belgaum, S. Musa, M. M. Alam, and M. M. Suúd, "A systematic review of load balancing techniques in software-defined networking," *IEEE Access*, vol. 8, pp. 98612–98636, 2020.

[12] P. Beslin Pajila and E. Golden Julie, "Detection of DDoS attack using SDN in IoT: A survey," in *Intelligent Communication Technologies and Virtual Mobile Networks*, pp. 438–452, 2019.

[13] M. S. Bonfim, K. L. Dias, and S. F. Fernandes, "Integrated NFV/SDN architectures: A systematic literature review," *ACM Computing Surveys*, vol. 51, no. 6, pp. 1–39, 2019.

[14] P. Borylo, G. Davoli, M. Rzepka, A. Lason, and W. Cerroni, "Unified and standalone monitoring module for NFV/SDN infrastructures," *Journal of Network and Computer Applications*, vol. 175, pp. 102934, 2021.

[15] A. Cetinkaya, H. Ishii, and T. Hayakawa, "An overview on denial-of-service attacks in control systems: Attack models and security analyses," *Entropy*, vol. 21, no. 2, pp. 210, 2019.

[16] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective näive bayes algorithm," *Knowledge-Based Systems*, vol. 192, pp. 105361, 2020.

[17] K. A. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019.

[18] F. S. Dantas Silva, E. Silva, E. P. Neto, M. Lemos, A. J. Venancio Neto, and F. Esposito, "A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios," *Sensors*, vol. 20, no. 11, pp. 3078, 2020.

[19] V. Deepa, K. M. Sudar, and P. Deepalakshmi, "Design of ensemble learning methods for DDoS detec-

tion in SDN environment," in *International Conference on Vision Towards Emerging Trends in Communication and Networking*, pp. 1–6, 2019.

[20] C. Dharmadhikari, S. Kulkarni, S. Temkar, S. Bendale, and B. Student, "A study of DDoS attacks in software defined networks," *IRJET*, vol. 6, no. 12), 2019.

[21] P. Ding, J. Li, L. Wang, M. Wen, and Y. Guan, "Hybrid-cnn: An efficient scheme for abnormal flow detection in the SDN-based smart grid," *Security and Communication Networks*, vol. 2020, 2020.

[22] Z. Ding, Y. Huang, H. Yuan, and H. Dong, "Introduction to reinforcement learning," in *Deep Reinforcement Learning*, pp. 47–123, 2020.

[23] S. Dong, K. Abbas, and R. Jain, "A survey on distributed denial of service , no. ddos) attacks in SDN and cloud computing environments," *IEEE Access*, vol. 7, pp. 808130-828, 2019.

[24] Z. Eghbali and M. Z. Lighvan, "A hierarchical approach for accelerating IoT data management process based on SDN principles," *Journal of Network and Computer Applications*, vol. 181, pp. 103027, 2021.

[25] M. Elbes, S. Alzubi, T. Kanan, A. Al-Fuqaha, and B. Hawashin, "A survey on particle swarm optimization with emphasis on engineering and network applications," *Evolutionary Intelligence*, vol. 12, no. 2, pp. 113–129, 2019.

[26] L. F. Eliyan and R. Di Pietro, "Dos and DDoS attacks in software defined networks: A survey of existing solutions and research challenges," *Future Generation Computer Systems*, vol. 122, pp. 149–171, 2021.

[27] M. S. Elsayed, N. A. Le-Khac, S. Dev, and A. D. Jurcut, "Ddosnet: A deep-learning model for detecting network attacks," in *IEEE 21st International Symposium on" A World of Wireless, Mobile and Multimedia Networks*, pp. 391–396, 2020.

[28] A. P. Fajar and T. W. Purboyo, "A survey paper of distributed denial-of-service attack in software defined networking," *International Journal of Applied Engineering Research*, vol. 13, no. 1, pp. 476–482, 2018.

[29] J. D. Gadze, A. A. Bamfo-Asante, J. O. Agyemang, H. Nunoo-Mensah, and K. A. B. Opare, "An investigation into the application of deep learning in the detection and mitigation of DDoS attack on SDN controllers," *Technologies*, vol. 9, no. 1, pp. 14, 2021.

[30] T. T. Gao, H. Li, and S. L. Yin, "Adaptive convolutional neural network-based information fusion for facial expression recognition," *International Journal of Electronics and Information Engineering*, vol. 13, no. 1, pp. 17–23, 2021.

[31] A. A. Gebremariam, M. Usman, and M. Qaraqe, "Applications of artificial intelligence and machine learning in the area of SDN and NFV: A survey," in *16th International Multi-Conference on Systems, Signals & Devices*, pp. 545–549, 2019.

[32] B. B. Gupta and A. Dahiya, "Distributed Denial of Service , no. DDoS) Attacks: Classification, Attacks, Challenges and Countermeasures," *CRC press*, 2021.

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

[34] S. H. Haji, S. Zeebaree, R. H. Saeed, S. Y. Ameen, H. M. Shukur, N. Omar, M. A. Sadeeq, Z. S. Ageed, I. M. Ibrahim, and H. M. Yasin, "Comparison of software defined networking with traditional networking," *Asian Journal of Research in Computer Science*, vol. 9, no. 2, pp. 1–18, 2021.

[35] A. A. Hameed, B. Karlik, M. S. Salman, and G. Eleyan, "Robust adaptive learning approach to self-organizing maps," *Knowledge-Based Systems*, vol. 171, pp. 25–36, 2019.

[36] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, "Recurrent neural network wave functions," *Physical Review Research*, vol. 2, no. 2, pp. 023358, 2020.

[37] M. Hussain, J. J. Bird, and D. R. Faria, "A study on cnn transfer learning for image classification," in *UK Workshop on computational Intelligence*, pp. 191–202, 2018.

[38] G. A. Jaafar, S. M. Abdullah, and S. Ismail, "Review of recent detection methods for http DDoS attack," *Journal of Computer Networks and Communications*, vol. 2019, 2019.

[39] A. H. Janabi, T. Kanakis, and M. Johnson, "Convolutional neural network-based algorithm for early warning proactive system security in software defined networks," *IEEE Access*, vol. 10, pp. 14301–14310, 2022.

[40] D. Jiang, H. Li, and S. Yin, "Speech emotion recognition method based on improved long short-term memory networks," *International Journal of Electronics and Information Engineering*, vol. 12, no. 4, pp. 147–154, 2020.

[41] B. Karan, D. Narayan, and P. Hiremath, "Detection of DDoS attacks in software defined networks," in *3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions*, pp. 265–270, 2018.

[42] S. Karapoola, P. K. Vairam, S. Raman, and V. Kamakoti, "Net-police: A network patrolling service for effective mitigation of volumetric DDoS attacks," *Computer Communications*, vol. 150, pp. 438–454, 2020.

[43] R. Karthika and M. Maheswari, "Detection analysis of malicious cyber attacks using machine learning algorithms," *Materials Today: Proceedings*, vol. 68, pp. 26-34, 2022.

[44] S. Kaur, K. Kumar, N. Aggarwal, and G. Singh, "A comprehensive survey of DDoS defense solutions in SDN: Taxonomy, research challenges, and future directions," *Computers & Security*, vol. 110, pp. 102423, 2021.

[45] S. Khan and A. Akhunzada, "A hybrid dl-driven intelligent SDN-enabled malware detection framework for internet of medical things," *Computer Communications*, vol. 170, pp. 209–216, 2021.

[46] T. Y. Kim and S. B. Cho, "Predicting residential energy consumption using cnn-lstm neural networks," *Energy*, vol. 182, pp. 72–81, 2019.

[47] Z. Latif, K. Sharif, F. Li, M. M. Karim, S. Biswas, and Y. Wang, "A comprehensive survey of interface protocols for software defined networks," *Journal of Network and Computer Applications*, vol. 156, pp. 102563, 2020.

[48] T. H. Lee, L. H. Chang, and C. W. Syu, "Deep learning enabled intrusion detection and prevention system over SDN networks," in *IEEE International Conference on Communications Workshops*, pp. 1–6, 2020.

[49] J. Li, D. Li, Y. Yu, Y. Huang, J. Zhu, J. Geng, "Towards full virtualization of SDN infrastructure," *Computer Networks*, vol. 143, pp. 1–14, 2018.

[50] P. Li, M. Abdel-Aty, and J. Yuan, "Real-time crash risk prediction on arterials based on LSTM-CNN," *Accident Analysis & Prevention*, vol. 135, pp. 105371, 2020.

[51] H. Lin, S. Cao, J. Wu, Z. Cao, and F. Wang, "Identifying application-layer DDoS attacks based on request rhythm matrices," *IEEE Access*, vol. 7, pp. 164480–164491, 2019.

[52] X. Liu and C. J. Hsieh, "Rob-gan: Generator, discriminator, and adversarial attacker," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11234–11243, 2019.

[53] A. D. Lopez, A. P. Mohan, and S. Nair, "Network traffic behavioral analytics for detection of DDoS attacks," *SMU data science review*, vol. 2, no. 1, pp. 14, 2019.

[54] T. Luo and S. G. Nagarajan, "Distributed anomaly detection using autoencoder neural networks in wsn for IoT," in *IEEE International Conference on Communications*, pp. 1–6, 2018.

[55] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research*, vol. 9, pp. 381–386, 2020.

[56] Y. Meng, Z. Huang, S. Wang, G. Shen, and C. Ke, "Som-based DDoS defense mechanism using SDN for the internet of things," *arXiv preprint arXiv*, 2003.06834, 2020.

[57] M. Morchid, "Parsimonious memory unit for recurrent neural networks with application to natural language processing," *Neurocomputing*, vol. 314, pp. 48–64, 2018.

[58] T. M. Nam, P. H. Phong, T. D. Khoa, T. T. Huong, P. N. Nam, N. H. Thanh, L. X. Thang, P. A. Tuan, V. D. Loi, *et al.*, "Self-organizing map-based approaches in DDoS flooding detection using SDN," in *International Conference on Information Networking*, pp. 249–254, 2018.

[59] A. R. Narayanadoss, T. Truong-Huu, P. M. Mohan, and M. Gurusamy, "Crossfire attack detection using deep learning in software defined its networks," in *IEEE 89th Vehicular Technology Conference*, pp. 1–6, 2019.

[60] T. N. Nguyen, "The challenges in SDN/ml based network security: A survey," *arXiv preprint arXiv*, 1804.03539, 2018.

[61] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proenca, "Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment," *IEEE Access*, vol. 8, pp. 83765–83781, 2020.

[62] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proenca Jr, "Adversarial deep learning approach detection and defense against DDoS attacks in SDN environments," *Future Generation Computer Systems*, vol. 125, pp. 156–167, 2021.

[63] B. Nugraha and R. N. Murthy, "Deep learning-based slow DDoS attack detection in SDN-based networks," in *IEEE Conference on Network Function Virtualization and Software Defined Networks*, pp. 51–56, 2020.

[64] H. H. Patel and P. Prajapati, "Study and analysis of decision tree-based classification algorithms," *International Journal of Computer Sciences and Engineering*, vol. 6, no. 10, pp. 74–78, 2018.

[65] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine Learning*, pp. 101–121, 2020.

[66] H. Polat, M. Türkoğlu, O. Polat, and A. Sengür, "A novel approach for accurate detection of the DDoS attacks in SDN-based scada systems based on deep recurrent neural networks," *Expert Systems with Applications*, vol. 197, pp. 116748, 2022.

[67] A. Pradhan and R. Mathew, "Solutions to vulnerabilities and threats in software defined networking , no. sdn)," *Procedia Computer Science*, vol. 171, pp. 2581–2589, 2020.

[68] M. Priyadarsini and P. Bera, "Software defined networking architecture, traffic management, security, and placement: A survey," *Computer Networks*, vol. 192, pp. 108047, 2021.

[69] Y. Qin, J. Wei, and W. Yang, "Deep learning based anomaly detection scheme in software-defined networking," in *20th Asia-Pacific Network Operations and Management Symposium*, pp. 1–4, 2019.

[70] D. S. Rana, S. A. Dhondiyal, and S. K. Chamoli, "Software defined networking (SDN) challenges, issues and solution," *International Journal of Computer Science and Engineering*, vol. 7, no. 1, pp. 884–889, 2019.

[71] R. Sahay, W. Meng, and C. D. Jensen, "The application of software defined networking on securing computer networks: A survey," *Journal of Network and Computer Applications*, vol. 131, pp. 89–108, 2019.

[72] K. S. Sahoo, A. Iqbal, P. Maiti, and B. Sahoo, "A machine learning approach for predicting DDoS traffic in software defined networks," in *International*

*Conference on Information Technology*, pp. 199–203, 2018.

[73] M. M. Salim, S. Rathore, and J. H. Park, "Distributed denial of service attacks and its defenses in IoT: A survey," *The Journal of Supercomputing*, vol. 76, no. 7, pp. 5320–5363, 2020.

[74] A. Sangodoyin, B. Modu, I. Awan, and J. P. Disso, "An approach to detecting distributed denial of service attacks in software defined networks," in *IEEE 6th International Conference on Future Internet of Things and Cloud*, pp. 436–443, 2018.

[75] I. H. Sarker, "Machine learning: Algorithms, real world applications and research directions," *SN Computer Science*, vol. 2, no. 3, pp. 1–21, 2021.

[76] A. Shirmarz and A. Ghaffari, "Performance issues and solutions in SDN-based data center: A survey," *The Journal of Supercomputing*, vol. 76, no. 10, pp. 7545–7593, 2020.

[77] J. Singh and S. Behal, "Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions," *Computer Science Review*, vol. 37, pp. 100279, 2020.

[78] K. Sinha, A. Viswanathan, and J. Bunn, "Tracking temporal evolution of network activity for botnet detection," *arXiv preprint arXiv*, 1908.03443, 2019.

[79] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," *Expert Systems with Applications*, vol. 134, pp. 93–101, 2019.

[80] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," *Peer-to-Peer Networking and Applications*, vol. 12, no. 2, pp. 493–501, 2019.

[81] J. Sun, Y. Zhang, F. Liu, H. Wang, X. Xu, Y. Li, "A survey on the placement of virtual network functions," *Journal of Network and Computer Applications*, vol. page 103361, 2022.

[82] H. Taud and J. Mas, "Multilayer perceptron , no. mlp)," in *Geomatic Approaches for Modeling Land Change Scenarios*, pp. 451–455, 2018.

[83] K. H. Thung and C. Y. Wee, "A brief review on multitask learning," *Multimedia Tools and Applications*, vol. 77, no. 22, pp. 29705–29725, 2018.

[84] T. Ubale and A. K. Jain, "Survey on DDoS attack techniques and solutions in software-defined network," in *Handbook of Computer Networks and Cyber Security*, pp. 389–419, 2020.

[85] R. M. A. Ujjan, Z. Pervez, K. Dahal, A. K. Bashir, R. Mumtaz, and J. González, "Towards sflow and adaptive polling sampling for deep learning based DDoS detection in SDN," *Future Generation Computer Systems*, vol. 111, pp. 763–779, 2020.

[86] C. Wang, B. Wang, H. Liu, and H. Qu, "Anomaly detection for industrial control system based on autoencoder neural network," *Wireless Communications and Mobile Computing*, vol. 2020, 2020.

[87] H. Wang, H. Xu, C. Qian, J. Ge, J. Liu, and H. Huang, "Prepass: Load balancing with data plane resource constraints using commodity SDN switches," *Computer Networks*, vol. 178, pp. 107339, 2020.

[88] P. Wang, F. Ye, X. Chen, and Y. Qian, "Datanet: Deep learning based encrypted network traffic classification in SDN home gateway," *IEEE Access*, vol. 6, pp. 55380–55391, 2018.

[89] Z. Wang and T. Hong, "Reinforcement learning for building controls: The opportunities and challenges," *Applied Energy*, vol. 269, pp. 115036, 2020.

[90] C. S. Wickramasinghe, K. Amarasinghe, and M. Manic, "Deep self-organizing maps for unsupervised image classification," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 11, pp. 5837–5845, 2019.

[91] R. Xia, H. Dai, J. Zheng, H. Xu, M. Li, and G. Chen, "Packet-in request redirection: A load balancing mechanism for minimizing control plane response time in SDNs," *Journal of Systems Architecture*, vol. 129, p. 102590, 2022.

[92] J. Yan and J. Yuan, "A survey of traffic classification in software defined networks," in *1st IEEE International Conference on Hot Information-Centric Networking*, pp. 200–206, 2018.

[93] L. Yang, B. Ng, W. K. Seah, L. Groves, and D. Singh, "A survey on network forwarding in software-defined networking," *Journal of Network and Computer Applications*, vol. 176, pp. 102947, 2021.

[94] L. Yao, J. Liu, D. Wang, J. Li, and B. Meng, "Formal analysis of SDN authentication protocol with mechanized protocol verifier in the symbolic model," *International Journal of Network Security*, vol. 20, no. 6, pp. 1125–1136, 2018.

[95] A. Yazdinejadna, R. M. Parizi, A. Dehghantanha, and M. S. Khan, "A kangaroo-based intrusion detection system on software-defined networks," *Computer Networks*, vol. 184, pp. 107688, 2021.

[96] Y. Zhao, Y. Li, X. Zhang, G. Geng, W. Zhang, and Y. Sun, "A survey of networking applications applying the software defined networking concept based on machine learning," *IEEE Access*, vol. 7, pp. 95397–95417, 2019.

[97] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[98] X. Zou, Y. Hu, Z. Tian, and K. Shen, "Logistic regression model optimization and case analysis," in *IEEE 7th International Conference on Computer Science and Network Technology*, pp. 135–139, 2019.

# Biography

**Ntumpha Patrick Mwanza** received his Bsc. in Computer Science degree from Cavendish University, Zambia, and M.S in Cybersecurity and Leadership from the University of Washington Tacoma, USA. Currently, he is a

Ph.D. student at the University of Colorado, Colorado Springs. His research interests include Distributed Denial of Service (DDoS) attack detection, Malware Detection and Classification using deep learning.

**Jugal Kalita** received his Ph.D. from the University of Pennsylvania, Philadelphia. He is a professor and department chair of computer science at the University of Colorado, Colorado Springs. His research interests are in machine learning and natural language processing. He has published over 250 papers in international journals and referred conference proceedings and has written four books.