SDN BASED DDoS DETECTION USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES

Report of Internship

TATA CONSULTANCY SERVICES (TCS)

15-Nov-2021 to 27-May-2022

Submitted by

Ms. SUBUHANA N

Intern Emp ID: 2170619

Final-year Postgraduate student

Centre for Artificial Intelligence

THANGAL KUNJU MUSALIAR COLLEGE OF ENGINEERING KERALA

Under the guidance of
RAJEEV AZHUVATH (TCS Mentor ID: 120914)
&
Prof. SUMOD SUNDAR (TKMCE)

MAY 2022

ACKNOWLEDGEMENT

A successful project is a fruitful culmination of efforts by many people, some directly involved and some others indirectly, by providing support and encouragement. Firstly I would like to thank the almighty for giving me the wisdom and grace to make my project a successful one. I thank him for steering me to the shore of fulfilment under his protective wings.

With a profound sense of gratitude, I would like to express my heartfelt thanks to my mentor **Rajeev Azhuvath, Tata Consultancy Services (TCS)**, for his expert guidance, constant support and cooperation.

I express my sincere gratitude to **Dr. T A Shahul Hameed**, Principal of TKMCE and **Dr. Imthias Ahamed**, Professor and Head of the Department, Centre for Artificial Intelligence, TKMCE, for their immense encouragement. I would like to thank my college guide **Prof. Sumod Sundar**, Assistant Professor, Centre for Artificial Intelligence, TKMCE, for his expert guidance and cooperation.

I also express my thanks to my loving parents, brother and friends, for their support and encouragement in the successful completion of this project work.

SUBUHANA N

Abstract

Software-Defined Networking (SDN) is an emerging paradigm, which evolved in recent years to address the weaknesses in traditional networks. The SDN's main goal is to separate the control and data planes, making network management easier and enabling for more efficient programmability. The centralized structure of SDN brings new vulnerabilities. Distributed Denial-of-Service (DDoS) is one of the most prevalent and sophisticated threat. DDoS attack tries to disrupt the available services of a victim to block the victim from providing service to the legitimate users, by sending massive malicious requests from a large number of hijacked machines. DDoS attacks are easy to initiate, hard to defend and strong destructive effect. So that accurate detection of DDoS attack is necessary. In recent years, many studies have been done to secure SDN using Machine Learning and Deep Learning techniques. In this work Supervised Machine Learning algorithms such as Logistic Regression, Gaussian Naive Bayes, SVM, Decision Tree, Random Forest and Gradient Boosting are employed for DDOS detection in SDN environments. The unsupervised Machine Learning technique such as K Means clustering with PCA and TSNE dimensionality reduction technique is used. In case of Deep Learning techniques DNN, Auto Encoder and XGBOOST, CNN, LSTM, BiLSTM, TabNet and TCN are used for DDoS detection. The dataset used is DDoS attack SDN DDoS dataset. This is a publically available DDoS dataset created in SDN environment. The experimental result shows that TCN performs very well in SDN DDoS dataset and achieve an accuracy of 99.69 percentage. The results are validated using another dataset called INSDN dataset. It is a comprehensive SDN dataset to verify the performance of intrusion detection systems. The TCN gives 99.96 percentage accuracy in INSDN dataset and implies that TCN performs very well in both the dataset.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

We are in the era of exponentially growing digitalization. The role of emerging information and communication technologies in our day-to-day life is undeniable. Moreover, it creates new security concerns. These systems are vulnerable to various cyber threats and attacks. The Software-Defined Network (SDN) is a new network paradigm promises more dynamic and efficiently manageable network architecture. The network intelligence is logically decoupled and centralized into the Control layer. The major benefit of SDN is that it separates the control and data planes, making the network more versatile and easier to manage. The entire system can be managed by a single distant computer known as the controller. Many business-related and industrialized enterprises are implementing SDN technology in their network environments.

The centralized structure of SDN brings new vulnerabilities to the system. Distributed Denial of Service (DDoS) are the most prevalent and sophisticated attack. DDoS attack tries to overwhelm the available resources of a victim from providing service for normal users, by sending massive malicious requests from a huge number of hijacked machines. DDoS attacks are easy to initiate, there have been a certain number of easily obtainable DDoS attack tools. DDoS attacks are hard to defend and it has strong destructiveness. Many companies such as Amazon, Facebook, Twitter, GitHub suffered DDoS which leads to big losses. So it is necessary to accurately detect DDoS attack and protect the system.

The first line of defense against these attacks are Network Intrusion Detection System (NIDS). Artificial Intelligence is now commonly used in defensive measure. The Machine Learning algorithms such as SVM, Naïve bayes, Decision Tree, Random Forest are used for the DDoS detection. The Deep Learning Techniques such as DNN, Auto Encoders, CNN, LSTM, TabNet and TCN can significantly improve the performance of DDoS detection systems. This work concentrates on implementation and testing of detecting DDoS attack in SDN environment by three approaches.

- Supervised with traditional ML algorithms.
- Unsupervised approach of clustering and dimensionality reduction.
- Supervised with Neural Networks.

Rest of this paper is organized as the following sections: Section 2 reviews the different machine learning and deep learning techniques for DDoS detection in SDN environments.

Section 3 defines the SDN architecture and DDoS attack. Section 4 includes dataset description and section 5 includes detailed methodology. An exhaustive discussion on the experimental results is summarized in Section 6.Section 7 includes the conclusion.

Chapter 2

Related Works

In recent years, many studies have been done to secure SDN using AI techniques. In this section, discuss several studies of DDoS attack detection systems based on machine learning and deep learning techniques.

Tonkal et al.[1] 2021 proposed a Machine Learning Approach with Neighboring Component Analysis (NCA) for DDoS Attack Detection in Software-Defined Network. The objective is to classify the SDN traffic as normal or attack traffic. The dataset used is “DDoS attack SDN Dataset”. It is a publicly available dataset including a total of 23 features. NCA algorithm is used to reveal the most relevant features by feature selection and perform an effective classification. After preprocessing and feature selection stages, the dataset was classified by k-Nearest Neighbor, Decision Tree, Artificial Neural Network, and Support Vector Machine algorithms. The obtained results are promising out that the proposed approach can achieve more efficient results compared to traditional machine learning models.

Aamir et al. [2] 2021 proposed a Clustering based semi-supervised machine learning for DDoS attack classification. The network traffic flow contains both normal and DDoS attack traffic. The clustering methods include agglomerative and K-means with feature extraction under Principal Component Analysis (PCA). A voting method is used to label the data and obtain classes to distinguish attacks from normal traffic. After labeling, supervised machine learning algorithms KNN, SVM and RF are applied to the trained models for DDoS attack classification. The dataset used is obtained with the network traffic generated in OPNET Modeler 14.5 simulator. The experimental results shows that 95 percentage, 92 percentage and 96.66 percentage accuracy scores are obtained with kNN, SVM and RF models respectively with optimized parameter tunings in given sets of values. The method is also validated for labelling accuracy using a subset of the CICIDS2017 dataset.

Cil et al. [3] 2021 proposed a Deep Learning model for Detection of DDoS attacks with feed forward based Deep Neural Network model. Deep Learning models are more effective for the detection of DDoS attacks on network traffic. Since it has feature extraction and classification algorithms in its structure, as well as layers that update themselves as it is trained, the DNN model can work quickly and accurately. The CICDDoS2019 dataset is used. The experimental results shows that the DNN model are effective for the detection and classification of DDoS attacks on network traffic.

Haider et al. [4] 2020 proposed a Deep CNN ensemble framework for efficient DDoS attack detection in software defined networks. Two similar Deep Learning models are combined to build an ensemble model and two complimentary models (RNN+LSTM) are used for creating a hybrid model. The CICIDS2017 dataset is used for the experiment. The experimental results shows that ensemble CNN model outperforms all other models.

Elsayed et al. [5] 2020 proposed a framework called DDoSNet, an intrusion detection system against DDoS attacks in SDN environments. It uses the DL approach, that combines RNN and Autoencoder. RNN can focus on the temporal dependencies. Autoencoder can significantly increase the anomaly detection accuracy. The model can reduce the data dimensionality by automatically extracting the features from input data. Each layer of the Autoencoder is a simple RNN layer. The CICDDoS2019 dataset is used. When comparing the performance of the DDOSNet to that of other classical ML algorithms, it shows that DDoSNet excels the others.

Cui et al. [6] 2021 proposed an SDN-enabled Gated Recurrent Unit Recurrent Neural Network (GRU-RNN) intrusion detection system. The proposed method was tested on the NSL-KDD and CICIDS2017 datasets, with results of 89 percentage and 99 percentage accuracy, respectively. When compared to other state-of-the-art algorithms, the GRU-RNN strategy achieves an 89 percentage detection rate in the NSL-KDD dataset with the smallest number of features. In the CICIDS2017 dataset, the GRU-DNN obtains an impressive detection rate of 99 percentage while dealing with DDoS attacks.

The advantages and limitations of recent related works are summarized in table 2.1.

Title	Technique used	Advantages	Disadvantages
Machine Learning Approach Equipped with Neighborhood Component Analysis for DDoS Attack Detection in Software-Defined Networking [1] (2021)	NCA, KNN, Decision Tree,ANN,SVM	NCA gives most relevant features by feature selection, UpToDate dataset	Does not give optimal number of features to be selected, The performance depends on the selected features
Clustering based semi-supervised machine learning for DDoS attack classification [2] (2021)	Agglomerative clustering, K means clustering, KNN, SVM, Random Forest	Optimizing and validating the model improves the performance	Clustering approach leads to high false positive values
Detection of DDoS attacks with feed forward based deep neural network model [3] (2021)	DNN	High accuracy	Failed to detect adverasarial attacks
A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks [4] (2020)	RNN, LSTM, CNN, RNN+LSTM	Improved accuracy Minimal computational complexity	Failed to detect adversarial DDoS attacks
DDoSNet: A Deep-Learning Model for Detecting Network Attacks [5] (2020)	RNN AutoEncoder	Temporal correlation of data is considered	Vanishing Gradient problem
Intrusion Detection in SDN-Based Networks: Deep Recurrent Neural Network Approach [6] (2019)	GRU-RNN GRU-DNN	The method offers a lot of potential for real-time detection	The model is not optimized for improving accuracy and reduce controller overhead

Table 2.1: Review

Chapter 3

SDN BASED DDoS DETECTION SYSTEM

3.1 SDN Architecture

Software defined networking has emerged as one of the fastest evolution in next generation networks. The software controlled activities over the network had been from 1980s. The changes in active network and introduction of network programmability created a huge impact in academia as well as industry. Decouple of control plane and data plane is the fundamental characteristic in the SDN. Centralized control over the network plane plays the major role for minimizing the time and dynamic utilization of resources, in which the control plane monitored the entire network and forward plane followed the instructions given by control plane for traffic management.The centralized network of SDN enhances the security of the entire security architecture. Fast accessing and dynamic traffic rule updation are the main factors enhancing the security of SDN network, though centralized nature of networking makes changes in policy selection and mitigates the risk of collision. OpenFlow is implemented in SDN architecture. OpenFlow application interface is interacted through control plane which is situated northbound to the data plane of southbound API.
The SDN architecture consist of three layers. Infrastructure layer This layer is integrated with various networking components which are used to form underlying network to forward network traffic. It acts as a physical layer of the network which communicates to the virtualization network laid down through the control panel. Control layer This layer is the land of control panel in SDN network. It includes the various protocol modules. This acts as intelligence of the network infrastructure. This is the area for every user to work with its own products and frameworks in SDN networking. In this layer, most of business-based logic are written. Application layer This layer is responsible for open-source platform for developer who develops innovative applications such as improving the network topology, network statics, network states etc. The SDN security vector architecture is divided into four sections, namely southbound, northbound, eastbound, and westbound. South bound Section, is popular for its interaction between controllers and switches. OpenFlow protocol is a southbound interface to the controller representing the bridge which connects the controller and forward plane such as switches. Hypervisor plays the key role in the present day SDN architecture which bridges gap the control between the controller and the protocols both southbound and northbound. Northbound Section is the most critical API in the

SDN environment. A majority of networking components exist in this section. Pyretic and frenetic is an SDN specific policy program language which communicates with controllers in the northbound section. All the security applications such as virtualized firewalls and intrusion detection system have a common API for interacting with the controller. East and West Section management of SDN architecture is done by east and west bound sections. The management plane is controlled by the distributed architecture in which instructions are conveyed through the controller for managing the data. The distributed architecture has some important functionalities including control, management, monitor and task distribution for different low-level instances.

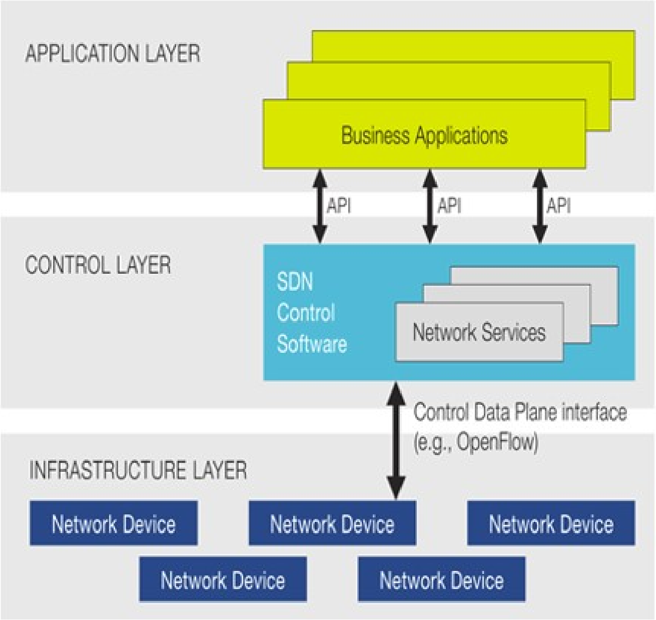

Figure 3.1: SDN architecture

3.2 DDoS Attack

SDNs has some weaknesses in their architecture. This may cause serious issues in the networks.DDoS attacks are the most prevalent and sophisticated threat. DDoS attacks in which users are denied access to network services are at the top of the attacks on the controller. rces on the target machine, and to prevent it from serving after a while by DDoS attacks. Attackers use ”botnets” created from devices called zombies hijacked by internet hackers. DDoS attacks are carried out with a large number of machines, so it is very difficult to detect and block. The frequency and severity of DDoS attacks are constantly increasing and can have fatal effects on many network services . For this reason, quick detection and prevention of DDoS attacks are some of the most important problems for network service providers and administrators. There is no built-in security mechanism on the controller that can distinguish between attack traffic and normal traffic. Therefore, it is very difficult to detect an attack.

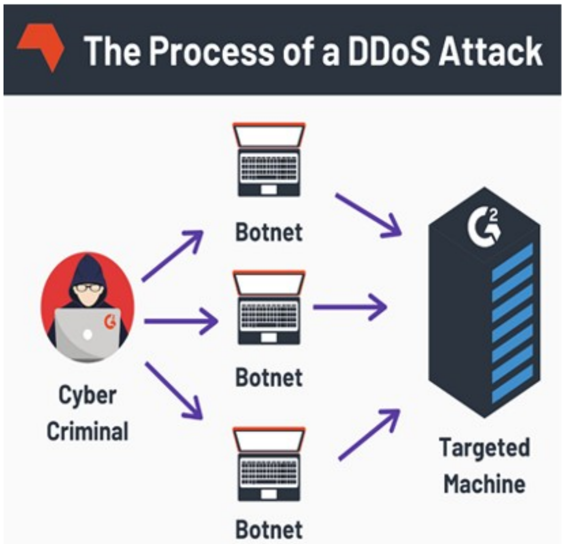

Figure 3.2: DDoS Attack

3.3 DDoS Detection System

In recent years, many studies have been done to secure SDN using machine learning and Deep learning techniques. In this work Supervised Machine learning algorithms such as Logistic Regression, Gaussian Naive Bayes, SVM, Decision Tree, Random Forest and Gradient Boosting are employed for DDOS detection in SDN environments. The unsupervised Machine Learning technique such as K Means clustering with PCA and TSNE dimensioanlity reduction technique is employed for DDoS detection. In case of Deep Learning techniques Deep Neural Network, Auto Encoder and XGBOOST classifier, CNN, LSTM, BiLSTM, TabNet and TCN are used for DDoS detection. The Deep Learning techniques can significantly improve the performance of DDoS Detection.

Chapter 4

Dataset and Data preprocessing

4.1 Dataset

4.1.1 DDoS Attack SDN Dataset

This is a SDN specific data set generated by using mininet emulator and used for traffic classification by machine learning and deep learning algorithms. The project begins with the creation of ten topologies in mininet, each with a single Ryu controller. The network simulation is run for both benign TCP, UDP, and ICMP traffic as well as malicious traffic, which includes TCP Syn attack, UDP Flood attack, and ICMP attack. The data collection contains a total of 23 features, some of which are retrieved from switches and others which are calculated. Extracted features include Switch-id, packet count, byte count, duration-sec, duration-nsec which is duration in nano-seconds, total duration is sum of duration-sec and durtaion-nsec, Source IP, Destination IP, Port number, tx bytes is the number of bytes transferred from the switch port, rx bytes is the number of bytes received on the switch port. dt field show the date and time which has been converted into number and a flow is monitored at a monitoring interval of 30 second. Calculated features include Packet per flow which is packet count during a single flow, Byte per flow is byte count during a single flow, Packet Rate is number of packets send per second and calculated by dividing the packet per flow by monitoring interval, number of Packet-ins messages, total flow entries in the switch, tx-kbps, rx-kbps are data transfer and receiving rate and Port Bandwidth is the sum of tx-kbps and rx-kbps. Last column indicates the class label which indicates whether the traffic type is benign or malicious. Benign traffic has label 0 and malicious traffic has label 1. Network simulation is run for 250 minutes, and 1,04,345 rows of data is collected. Table 4.1 summarizes the dataset's features.

4.1.2 DDoS Attack SDN Dataset Preprocessing

Before applying Machine Learning and Deep Learning techniques, we have to preprocess the dataset. The first step is to convert the Categorical variables such as source-destination IP and protocol that do not have numeric values. The rx-kbps and tot-kbps features having missing values in the dataset and that is filled with the mean value. The IP address is grouped based on 'dt' feature and count the number of request coming from the IP. Standardize the data using standard scalar.

Feature Name	Description
dt	Transmission moment of packets over the network device
switch	Switch ID
src	IP address of the sender of the packets
dst	IP address to which the packets was sent
pktcount	Number of packets
bytecount	Number of bytes
dur	Duration
dur-nsec	Duration in nano seconds
tot-dur	Total duration of network flow
flows	Number of flow packets
packetins	Total flow entries in the switch
pktperflow	Packet count during a single flow
byteperflow	Byte count during a single flow
pktrate	Number of packets per sec
Pairflow	Number of flow packets per second
Protocol	Types of communications internet protocols
port-no	Port number of the sender of the packets
tx-bytes	Number of bytes transferred from the switch port
rx-bytes	Number of bytes received on the switch port
tx-kbps	Data transfer rate
rx-kbps	Data Receiving rate
tot-kbps	Sum of tx-kbps and rx-kbps
label	Class label

Table 4.1: Features in Dataset.

4.1.3 INSDN Dataset

Co Different attacks on the files, control, and device layers are included in the InSDN dataset. The Internal and External. Comprehensive SDN dataset to verify the performance of intrusion detection systems. Total 84 features and 136743 network flows are available in the data set. The INSDN dataset includes different attacks that can strike the data, control, and application layers. The INSDN dataset consist of 5 types of attacks including DDoS, Dos, Probe, BFA and U2R. dataset's attack sources are divided into two categories:Internal and External.
Internal :- Internal users with complete entry to the SDN network are the source of these attacks. Internal attacks are uncommon in production networks, but they become more serious as time goes by, and they may trigger malicious behavior in network components. In certain cases, the attacker is unable to target network servers directly because they are protected by a high level of protection. Before initiating fresh assaults on other target servers, the attacker seeks to exploit weaknesses in the network system's individual users. The compromised hosts in the InSDN dataset are used to initiate different attacks from an internal SDN network.
External:- Outside networks are often used to initiate these attacks. The attacker is primar-

ily changing the SDN network with malevolent activities such as code vulnerabilities, refusal of service attacks, malware, and so on. We believe that the margin of the attacks in the dataset were generated from a third-party network in order to simulate real-world attack scenarios.

4.1.4 INSDN Dataset Preprocessing

: In the pre-processing stage the distribution of five category of attack is plotted. Removed the duplicate values from the dataset. The categorical variables such as Flow ID, Source and destination IP and Time stamp are converted into numerical values. The source IP is grouped based on 'Timestamp' feature and count the number of request coming from the IP. Find the correlation between features by using heat map graph. Then standardize the data using standard scalar.

Chapter 5

METHODOLOGY

5.1 Supervised Machine Learning Techniques

The supervised Machine Learning algorithms such as Logistic regression, SVM, Gaussian Naïve Bayes, SVM, Decision Tree, Random Forest and Gradient Boosting are employed for DDoS detection.

1. Logistic Regression
   Logistic regression is a linear model for binary classification problems. The dataset is loaded and splitted in to features and labels. Then the dataset is divided in to training and testing set (20percentage for testing and 80 percentage for training). The logistic regression model is created, model is trained on the dataset and test the model.

2. Gaussian Naïve Bayes
   A variant of Naïve Bayes algorithm that follows Gaussian normal distribution and supports continuous data. The Bayes Theorem is used to create Naive Bayes Classifiers. These classifiers make the assumption that the value of one feature is unrelated to the value of any other characteristic. The Gaussian Naïve bayes model is created, model is trained on the dataset and test the model.

3. Support Vector Machine (SVM)
   The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. The SVM model is created and train the model on the dataset. Evaluated the performance of the model for DDoS detection in SDN environment.

4. Decision Tree
   It is a tree structured classifier. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. This methodology is more commonly known as learning decision tree from data and the tree is called Classification tree as the target is to classify whether the traffic is Normal or DDoS. The dataset is divided into training and testing sets. The training set is applied as the input to the root of the tree. The prone procedure is carried out and repeated until all nodes becomes leaf nodes.

5. Random Forest
   Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Random forest takes the prediction from each tree and based on the majority votes of predictions it predicts the final output. The model is created with Random forest and trained on the dataset, test the model on the test set.

6. Gradient Boosting
   Powerful ensemble machine learning algorithm that uses decision trees. Decision trees are used as the weak learner in gradient boosting. A gradient descent procedure is used to minimize the loss when adding trees. The objective is to minimize the loss. The gradient boosting model is created, trained on the training set and test on the test set.

5.2 Unsupervised Approach of Clustering and Dimensionality Reduction

The goal of this unsupervised machine learning technique is to find similarities in the data point and group similar data points together. Clustering is the important aspects in case of unsupervised learning.

- Kmeans Clustering:
  K Means tries to group the data into a predetermined number of clusters. The purpose of K Means is to find data points that are similar and cluster them together while attempting to distance each cluster as far as possible. Here PCA and TSNE dimensionality reduction techniques are used and compared with K Means clustering. The dataset is loaded and apply K means on the original dataset. Feature reduction is done by PCA and apply K means on the PCA components. Then feature reduction is done by using TSNE, apply K means on the TSNE components. Finally compared the PCA and TSNE K means derived clusters.
- Principal Component Analysis (PCA):
  PCA or principal component analysis, is a well-known approach for reducing high-dimensional data to a low-dimensional space. The data or differences from our original features are ”squeezed” into what PCA refers to as main components (PC).
- T-Distributed Stochastic Neighbor Embedding (TSNE):
  T-SNE is a method for reducing high-dimensional data to a low-dimensional graph. It is also a strategy for reducing dimensionality. T-SNE can decrease dimensions with non-linear relationships, unlike PCA.
- Silhouette Method:
  This technique measures the separability between clusters. First, an average distance is found between each point and all other points in a cluster. Then it measures the distance between each point and each point in other clusters. We subtract the two average measures and divide by whichever average is larger. We ultimately want a high (ie. closest to 1) score which would indicate that there is a small intra-cluster average distance (tight clusters) and a large inter-cluster average distance (clusters well separated).

The silhouette score of PCA and TSNE K Means derived clusters are compared. For comparison, First merged the K Means clusters with the original unscaled features. We created two separate data frames. One for the PCA derives K Means clusters and one for the t-SNE K Means clusters. Obtained the univariate review of the clusters by comparing the clusters based on each individual feature.

5.3 Supervised with Neural Networks

Deep Learning is capable of automatically finding correlations in raw data, and it has the advantage of supervised and unsupervised learning. Deep Learning techniques can significantly improve the performance of DDoS detection. To classify the network traffic into normal and DDOS attack traffic, DNN, Auto Encoder and XGBOOST, CNN, LSTM. BiLSTM, TabNet and TCN are employed for DDoS detection.

(a) Deep Neural Network (DNN)
The DNN model is created with sequential model having input layer, 3 hidden layer and output layer. The input layer having 22 neurons corresponds to 22 features in the dataset and relu activation function used. The hidden layers consist of equal number of 30 neurons and relu activation function is used. There are also two dropout layers with dropout ratio 0.2. The output layer consists of 2 neurons with softmax activation function because it is a binary classification problem. Compile the model using ‘Sparse categorical cross entropy’ loss function and ‘ADAM’ optimizer. Fit the model with 20 epochs and batch size selected is 16.

(b) Auto Encoder and XGBoost classifier
Auto Encoder is a type of neural network that can be used to learn a compressed representation of raw data. Auto Encoder is used for the feature extraction. The encoder compresses the input, and the decoder attempts to recreate the input from the compressed version provided by the encoder. After training, the encoder model is saved, and the decoder is discarded. The encoder can then be used as a data preparation technique to perform feature extraction on raw data that can be used to train XGBOOST classifier. XGBoost, which stands for extreme Gradient Boosting, is a scalable, distributed gradient boosted decision tree.

(c) Convolutional Neural Network (CNN)
CNN has proven to be effective in many various studies and applications specifically in image classification field. CNN consists of multiples layers: input layer, convolutional layers, pooling layers, fully connected layer and output layer. Deepness of the CNN dependence on the number of layers used. 1D CNN is used for the DDoS detection in SDN environment. The CNN model is created with sequential model having input layer, 2 convolution layer, batch normalization layer, max pooling layer, 2 dropout layer, flatten layer and 2 dense layer. The input layer consist of 22 neurons corresponds to 22 features selected and relu activation function used. The output layer consists of 2 neurons with softmax activation function. CNN model is compiled with Sparse categorical cross entropy and Adam optimizer.

(d) Log Short Term Memory (LSTM)
Long-term and short-term memory model is a special RNN model, which is proposed to solve the problem of vanishing gradient and short term memory of RNN model. LSTM model replaces RNN cells in the hidden layer with LSTM cells to make them have long-term memory ability. The LSTM model is created with sequential model. The LSTM architecture consist of one LSTM layer with 22 neurons and tanh activation function and the kernel regularizer is l2. The Dense layer having relu activation function and l2 regularizer. The output layer having softmax activation function. The model is compiled with sparse categorical cross entropy loss function and Adam optimizer.

(e) Bi Directional Log Short Term Memory (BiLSTM)
BiLSTMs are an extension of traditional LSTMs that can improve model performance on sequence classification problems. In bi-directional, we can make the input flow in both directions to preserve the future and the past information. Bidirectional LSTMs train two instead of one LSTMs on the input sequence. The first on the input sequence as-is and the second on a reversed copy of the input sequence. The BiLSTM model is created with sequential model. The BiLSTM architecture consist of one BiLSTM layer with 22 neurons and tanh activation function and the kernel regularizer is l2. The Dense layer having relu activation function and l2 regularizer. The output layer having softmax activation function. The BiLSTM model is compiled with sparse categorical cross entropy loss function and Adam optimizer.

(f) TabNet
TabNet is a deep neural network specifically designed to learn from tabular data, developed by the research team at Google Cloud AI. TabNet inputs raw tabular data and is trained using gradient descent-based optimization, enabling flexible integration into end-to-end learning. Use sequential attention to choose which features to reason from at each decision step, enabling interpretability and better learning as the learning capacity is used for the most salient features. TabNet employs a single deep learning architecture for feature selection and reasoning. TabNet enables two kinds of interpretability: local interpretability that visualizes the importance of features and how they are combined, and global interpretability which quantifies the contribution of each feature to the trained model. TabNet trains on each row from a table, selects the relevant features in each step using a sparse learnable mask, and aggregates the predictions from each step to emulate an ensemble-like effect when making predictions. The TabNet classifier from Pytorch is used. The TabNet classifier model is defined and fit the model with 30 epochs.

(g) Temporal Convolutional Network (TCN)
TCN is a variation of Convolutional Neural Networks for sequence modelling tasks, by combining aspects of RNN and CNN architectures. TCN employs technique like multiple layers of dilated convolutions and padding of input sequences to handle different sequence lengths and detect dependencies between items that are not next to each other. The causal convolutions are used, where output at time t is convolved only with elements from time t and earlier in the previous layer, that is no information leakage from future to past. TCN uses a 1D fully-convolutional network (FCN) architecture, each hidden layer is the same length as

the input layer. TCNs possess very long effective history sizes using a combination of very deep networks and dilated convolutions. When using dilated convolutions, it is common to increase the dilated factor d exponentially with the depth of the network. This ensures the receptive field covering each input in the history, and enabled to get an extremely large receptive field as effective history by using deep networks. For the receptive field size of the TCN depends on the network depth n as well as filter size k and dilation factor d. Residual connections have proven to be very effective in training deep networks. In a residual network, skip connections are used to speed up training process and avoid vanishing gradient problems. The Keras TCN model is used. Defined the model with the compiled TCN from Keras TCN. The model is fitted with 20 epochs.

Chapter 6

Results and discussion

6.1 Hardware and experimental environment

The hardware used for the experiments includes Windows 10 Pro OS, 64-bit operating system, x64-based processor,Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz 1.50 GHz 16 GB RAM. The experimental environment was prepared by using Python 3.7 programming language. The framework used is Keras with TensorFlow as background in the Anaconda environment. Machine learning and deep learning libraries include - NumPy, Pandas, Matplotlib, and Scikit learn. Performance analysis identifies the best model having the highest accuracy. The general evaluation metrics such as Accuracy, Precision, Recall, F1 score, and confusion matrix are used.

6.2 Results of Supervised Machine Learning Techniques

The experimental result of supervised Machine Learning techniques is summarized in the Table 6.1

Evaluation Metrics	LR	GNB	SVM	DT	RF	GB
Accuracy(percentage)	77.11	67.35	78.44	99.95	100	98.52
Precision (percentage)	72.68	57.54	76.16	99.93	100	97.92
Recall (percentage)	66.29	62.41	65.17	99.95	100	99.04
F1 score (percentage)	75.54	66.17	76.67	99.95	99.95	98.46

Table 6.1: Experimental result of Supervised Machine Learning Techniques.

From Table II it is clearly visible that the Decision Tree, Random Forest and Gradient Boosting performs well and gives accuracy, precision, recall and F1 score near to 100 percentage.

6.3 Results of Unsupervised Approach of Clustering and Dimensionality Reduction

The experimental result of unsupervised Machine Learning techniques is summarized in the Table 6.2

Tecchniques	Sillhouette Score
Kmeans on original dataset	0.1961007339513506
Kmeans on PCA	0.4613677492070967
Kmeans on TSNE	0.34130406379699707

Table 6.2: Experimental result of Unsupervised Machine Learning Techniques.

We achieved a silhouette score of 0.196 in K Means on original dataset which is on the low end. We can see a definite improvement in K Means ability to cluster our data when we reduce the number of dimensions to 2 principal components. We get a K Means PCA Scaled Silhouette Score of 0.4613677492. Applying K Means to our 2 t-SNE derived components we were able to obtain a Silhouette score of 0.3413. When we compare the PCA and TSNE derived K Means cluster, PCA gives the better silhouette score.

6.4 Results of Supervise with Neural Networks in DDoS attack SDN dataset

The experimental result of Deep Learning techniques is summarized in the Table 6.3

Tecchniques	Accuracy(percentage)
DNN	98.39
AutoEncoder and XGBoost	97.66
CNN	99.50
LSTM	95.06
BiLSTM	95.09
TabNet	99.00
TCN	99.69

Table 6.3: Experimental result of Deep Learning Techniques in DDoS attack SDN dataset.

From the table 6.3, it is evident that the Deep Learning techniques can achieve very good results in case of DDoS detection SDN environments. The TCN gives the best accuracy in SDN DDoS dataset 99.67%. CNN, TabNet, DNN, Auto Encoder, LSTM, BiLSTM are also achieves good results 99.50%, 99.00%, 98.39%, 97.66%, 95.06%, 95.09% respectively.

6.5 Validation in INSDN dataset

The experimental results of Deep Learning models in INSDN dataset is summarized in Table 6.4.

Tecchniques	Accuracy(percentage)
DNN	99.95
AutoEncoder and XGBoost	99.93
CNN	99.96
LSTM	99.61
BiLSTM	99.62
TabNet	99.98
TCN	99.96

Table 6.4: Experimental result of Deep Learning Techniques in INSDN dataset.

From the Table 6.4, TabNet gives the best accuracy of 99.8% and TCN gives the second highest accuracy 99.96% in INSDN dataset. The other Deep Learning models can also achieve best accuracy for DDoS detection.

Chapter 7

Conclusion

This work focuses on propose and demonstrate the design, implementation, and testing of detecting DDoS by Machine Learning and Deep Learning solution that would allow end users to identify DDoS attack. The supervised Machine Learning technique such as Logistic regression, Gaussian Naive Bayes, SVM, Decision Tree, Random Forest and Gradient Boosting are employed for DDoS detection. The experiment results show that the decision Tree, Random Forest and Gradient Boosting achieves better results than other algorithms. In case of unsupervised Machine Learning techniques, K Means clustering with PCA and TSNE dimensionality reduction technique are employed for DDoS detection. The experimental result show that K Means with PCA gives the better silhouette score. In case of Deep learning techniques DNN, Auto Encoder and XGBoost, CNN, LSTM, BiLSTM, TabNet and TCN are used. For DDoS attack SDN dataset TCN gives the highest accuracy 99.69%. The experimental results in INSDN dataset implies that TabNet model performs well and achieves an accuracy of 99.98% and TCN gives the second highest accuracy 99.96%. The Machine Learning and Deep Learning techniques are used for the DDoS detection in SDN environment. The Deep Learning models achieves better results for DDoS detection. TabNet and TCN model performs very well and achieves highest accuracy than other Deep Learning models in both the datasets.

References

[1] Tonkal, Ö., Polat, H., Başaran, E., Cömert, Z. and Kocaoğlu, R., 2021. Machine Learning Approach Equipped with Neighbourhood Component Analysis for DDoS Attack Detection in Software-Defined Networking. Electronics, 10(11), p.1227.

[2] Aamir, M. and Zaidi, S.M.A., 2021. Clustering based semi-supervised machine learning for DDoS attack classification. Journal of King Saud University-Computer and Information Sciences, 33(4), pp.436-446.

[3] Cil, A.E., Yildiz, K. and Buldu, A., 2021. Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 169, p.114520.

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

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

[6] Cui, Y., Qian, Q., Guo, C., Shen, G., Tian, Y., Xing, H. and Yan, L., 2021. Towards DDoS detection mechanisms in software-defined networking. Journal of Network and Computer Applications, 190, p.103156.

[7] Devarajan, D. and Arora, K., 2021. Multiclass DDoS Detection using Machine Learning Ensemble. Available at SSRN 3884632.

[8] Sahoo, K.S., Panda, S.K., Sahoo, S., Sahoo, B. and Dash, R., 2019. Toward secure software-defined networks against distributed denial of service attack. The Journal of Supercomputing, 75(8), pp.4829-4874.