Mitigate Volumetric DDoS Attack using Machine Learning Algorithm in SDN based IoT Network Environment

Kumar J[1], Dr Arul Leena Rose P J[2*]
Research Scholar[1], Associate Professor[2]
Department of Computer Science-College of Science and Humanities,
SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, India[1, 2]

***Abstract*—Software-Defined Networking (SDN) is a recent trend that is combined with Internet of Things (IoT) in wireless network applications. SDN focus entirely on the upper-level network management and IoT enables monitoring the physical activity of the real-time environment via internet network connectivity. The IoT clusters with SDN often undergoes challenges like network security concerns like getting attacked by a Distributed Denial of Service (DDoS). The mitigation of network management issues is carried out by the frequent software update of SDN. On other hand, the security enhancement is needed to all alleviate the mitigation of security attacks in the network. With such motivation, the research uses machine learning based intrusion detection system to mitigate the DDoS attack in SDN-IoT network. The control layer in the SDN is responsible for the prevention of attacks in IoT network using a strong Intrusion Detection System (IDS) framework. The IDS enables a higher-level attack resistance to the DDoS attack as the framework involves feature selection-based classification model. The simulation is conducted to test the efficacy of the model against various levels of DDoS attacks. The results of simulation show that the proposed method achieves better classification of attacks in the network than other methods.**

***Keywords*—DDoS; SDN; IoT; machine learning**

I. Introduction

As the frequency of cyberattacks on governments and corporations across the world rises, academics like have been working feverishly to develop effective network intrusion detection systems (NIDS) [1].

Web-based platform attacks, DoS attacks, and malicious insiders are among the most severe forms of cybercrime [2]. Businesses risk having their intellectual property stolen, and governments risk having interruptions to their key national infrastructure if malicious software is permitted to infiltrate the system. Companies use security measures like firewalls, antivirus programs, and NIDS to keep hackers out of their networks [3].

A new architecture, the software-defined network (SDN) [4] divides the network control functions from its forwarding functions. By physically separating the control plane from the data plane, it is feasible to facilitate uncomplicated management of the network [5]. This component of SDN makes it easier to create new types of applications, which in turn promotes the need for a new sort of networking paradigm that can handle NIDS. Using SDN, developers may quickly and easily design innovative software. SDN controllers have the capacity to incorporate machine learning and deep learning (ML/DL) methodologies [6]. This facilitates better network visibility and security when IoT is interfaced with SDN.

Earlier attempts to deploy NIDS utilizing SDN controllers and deep learning techniques have met with mixed success. In [6], the authors implement a controller based on an anomalous algorithm. They constructed a deep neural network in order to cut down on the number of distinguishing criteria that may be used to discern regular traffic from abnormal traffic. In addition to this, they utilized deep learning strategies in order to evaluate their model [7].

From the problems stated above, it is found that there exists a gap of concurrent processing that effectively retain the level of accuracy while mitigating the attacks in the network.

The selection of a lightweight attack mitigation mechanism using a machine learning (ML) can pose a lighter load to the network and may not affect the network in terms of its computational burden.

In this paper, the research uses machine learning based intrusion detection system to mitigate the Distributed Denial of Service (DDoS) attack in SDN-IoT network. The Intrusion Detection System (IDS) enables a higher-level attack resistance to the DDoS attack as the framework involves feature selection-based classification model.

II. Related Works

In order to defend the control and data planes from DDoS attacks, Shoeb and Chithralekha [8] established a controller process priorities are set according to the node trust level, which is configured based on the node behaviour during regular business hours. The node worth is calculated based on its activity. In high-demand situations, the controller is set up to ignore requests from some nodes if the sum of their requests has already reached a predetermined maximum. The controller makes a rule change to one with a shorter timeout, prompting a response from the standard nodes as well.

Support vector machine (SVM) classifiers are proposed by Kokila et al. [9] for detecting DDoS attacks. The SVM must be trained with historical data before it can reliably predict the behaviour of unobserved traffic samples. When compared to

other simulated methods, the SVM has superior accuracy and fewer false positives. However, SVM is very dependent on the accuracy of the data used to train the model.

Xiao et al. [10] suggest a concept that employs a bloom filter in the SDN to detect link flooding attack. When the collector detects a deviation in link use, it checks the switch flow table to determine whether any aberrant flows can be deduced from the data. The detector can perform packet sniffing thanks to a controller that watches the network in real time. Seeing as the Bloom filter retains crucial IP features, it can be used to determine if the packet current classification is anomalous. However, neither a definition of anomalous link consumption nor the controller detection mechanisms are provided.

Lim et al. [11] suggest that changing one IP address is one way to lessen the impact of distributed denial of service attacks. If a host changes its IP address but then continues to send more than some threshold number of packets to the old address, the host will be marked as a bot and banned. It appears from the simulation results that bot-driven DDoS attacks can be countered. The question of what metric or threshold should be used to initiate defensive measures and withdrawal from a conflict remains unclear.

Similar to SVM and Self Organising Machine (SOM), the method for categorizing DDoS attacks described by Phan et al. [12] combines the two models. Both SVM and SOM models can be trained and tested using pre-existing data sets. Each protocol employs its own distinct set of SVMs to filter control-plane communications. If the SVM determines that data from a certain flow could be associated with the attack region, it will forward that data to the classifier. The SOM must decide if the current is appropriate for the current style era. Simulations show that when SVM and SOM are used together, better results are achieved than when using either approach alone.

An approach to detecting distributed denial of service (DDoS) attacks is presented by Chin et al. [13], which makes use of the interplay between a monitor, a correlater, and a controller. The Monitor component alerts the IDS whenever it finds something out of the ordinary on the network. Once the IDS has confirmed the existence of an attack, it will send the relevant data to the correlater.

Hameed and Khan [14] developed a secure protocol using a cooperative DDoS mitigation technique. The exchange of messages, certificates, and signatures are the fundamental building blocks of the C-to-C protocol. While the signature component is responsible for ensuring the accuracy and integrity of the data, the certificate component is responsible for establishing a trust connection between authenticating controllers. When a controller detects a DDoS attack, it warns its surrounding controllers as soon as possible by transmitting a list of malicious IP addresses and by modifying the policy on the data plane. As a result, access to these packets is being prohibited across the network in a number of different locations. The results of the simulation illustrate how rapidly this method may warn neighbouring controllers and thwart attacks.

Macedo et al. [15] came up with the idea of a multi-controller cluster when they were developing The DDoS Attack Mitigation in SDN Networks is comprised of these three stages: (i) identifying the overloaded controller by using control message latency or stability; (ii) selecting the controller with the highest performance for coordinated mitigation; and (iii) reducing the impact of attacks. The model identifies the overloaded controller by using control message latency or stability.

A DDoS security architecture was introduced by Sahay et al. [16]. Its primary goal is to reduce the amount of destructive Internet traffic. In order to identify network flows, and the customer end detection engine is the one that determines whether or not the traffic flow is malicious. The status of the connection is communicated from the controller belonging to the customer to the controller belonging to the service provider. A determination made by the ISP controller directs that the harmful flow be sent to the filter so that it can be examined in further detail. Nevertheless, the communication between controllers also needs to be secure.

III. Proposed Method

The research makes use of an intrusion detection system that is founded on machine learning in order to mitigate the impact that DDoS attacks have on SDN-IoT networks. The layers of SDN are provided in Fig. 1(a) and when the IoT network connected with SDN layers under attacks is illustrated in Fig. 1(b).

In Fig. 1(a) and 1(b), the control plane in SDN is responsible for mitigating the attacks in IoT network as it is embedded with a strong IDS framework that reads the network logs and classifies the traffic, and mitigates the attacks based on the anomalies present.

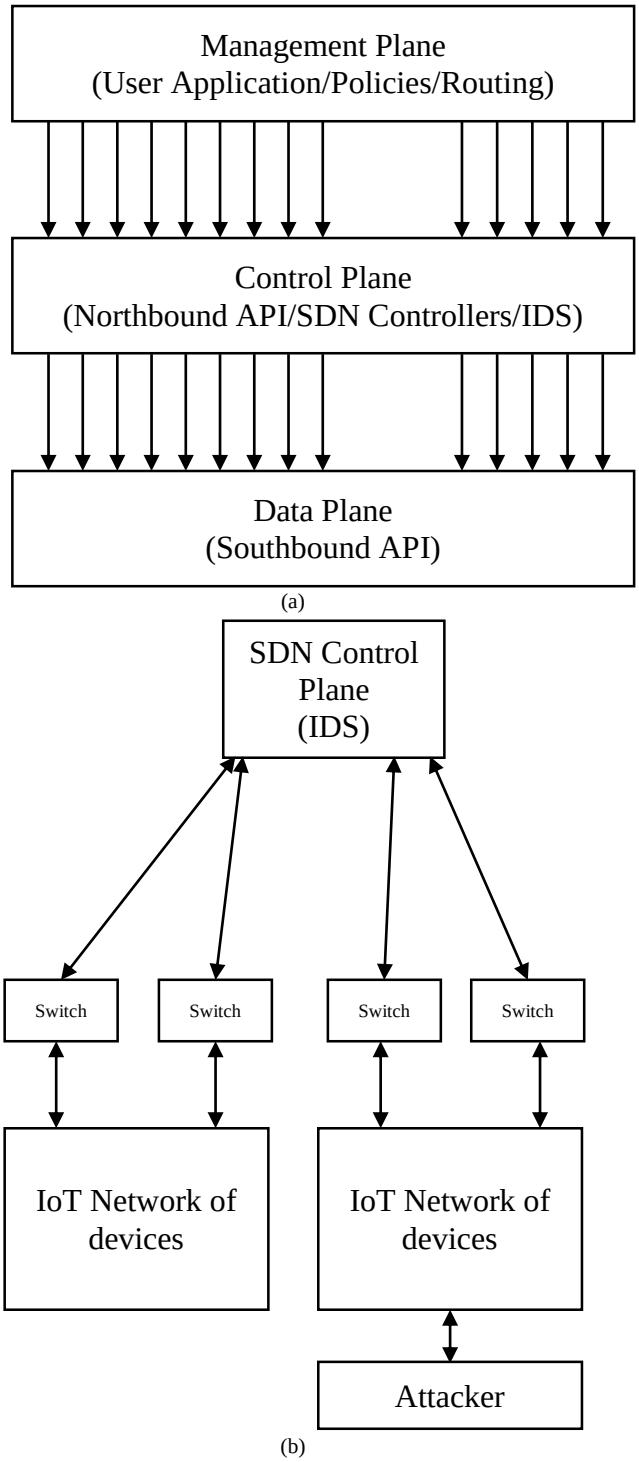

Fig. 1. SDN layered architecture b) IoT network with SDN architecture

It employs a feature selection–based classification methodology, the IDS is able to withstand more complex varieties of DDoS attack. A simulation is carried out in order to check that the model is capable of withstanding DDoS attacks of varying degrees of severity. The findings of the simulation reveal that the suggested strategy works better than competing alternatives when it comes to classifying network intrusions.

In Fig. 2, a variety of strategies and processes are illustrated for the purpose of putting an intrusion detection system into action. A number of methods have been created, and they can be roughly categorized as statistical approaches, data mining techniques, and machine learning-based methods. These methods were designed to identify anomalies in the data.

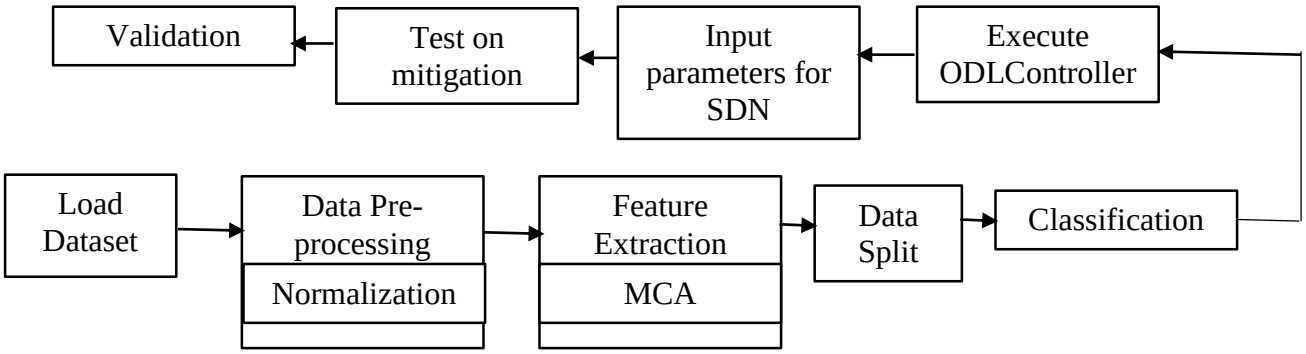

Fig. 2. Proposed IDS framework in SDN controller

A NIDS that relies on signatures can only detect previously discovered strains of malicious software. The detection system makes use of a set of rules that are derived from a combination of packet header and packet content inspection utilizing a present signature in order to identify potentially malicious network activity.

An ML/DL-based SDN-based intrusion detection system excels in a number of different areas, including security enforcement, virtual management, and QoS. SDN provides us with the opportunity to enhance the security of our networks by doing away with the requirement for specialist hardware, providing us with greater creative leeway in the manner in which we build our networks, and making it simpler for us to put into action new security measures.

A SDN can be constructed with fully adjustable features and software implementations of switches by making use of platforms that simulate and emulate real-world conditions. When it comes to putting the ideas of SDN into practice, Open Flow is one of the protocol standards that have been embraced by the largest number of organizations.

The SDN controller, also referred to as a network operating system, is an essential component of SDN networks. The SDN controller is responsible for presenting a consolidated image of the network as well as coordinating interactions with all programmable network components. In addition, there are already a variety of alternative SDN controllers that can be utilized. Fig. 4 is a representation of a network intrusion detection system that is based on SDN.

When compared to NIDS, machine learning makes it abundantly evident that researchers have started applying deep learning techniques. Deep learning is a potential strategy for the next generation of intrusion detection methods, as it can automatically uncover a correlation in the data. When applied to a wide variety of categorization difficulties in SDN networks, methodologies based on ML performed significantly better than state-of-art methods.

Classification problems appear to be where supervised machine learning systems shine the brightest. ML-based approaches performed far better than classic machine learning methods did. Due to the fact that the characteristics of attacks are unknown, unsupervised learning algorithms such as stacked autoencoders, RNNs, and hybrid-based algorithms will prove to be the most successful algorithms for implementing NIDS on an SDN platform.

A. Pre-Processing

The technique begins with the normalization of the dataset. The raw data is processed so it is standardized for the subsequent processing steps. This makes the entire thing simpler to design and more effective to put into practice. The first normalization of both the training/testing data set happens during the pre-processing step of the procedure.

The degree to which the underlying data are standardized determines a significant portion of the success of the weight coefficients selection approach. The standardization strategy does not incorporate any approaches that can be used to manipulate the detection rate in any way. Normalization of the measured values on a separate scale is conceptually shared scale before averaging the results of the measurements. There are many different kinds of normalization, and some of them require a rescaling technique in order to obtain values that are associated with an entirely new variable. The equation for the normalization of the mean and standard deviation is given as below:

$$\sigma_i^2 = -\frac{1}{(1-m-n)}\sum_{j=1}^{n}\varepsilon_j^2 \tag{1}$$

where

σ - standard deviation

m, n - parameter.

When the errors can be distinguished from one another, they can be formulated as follows:

$$g_i \sim \frac{T\sqrt{O}}{\sqrt{O+T^2-1}} \tag{2}$$

where

T, O – initialized datasets and

g - random variable.

The standard deviation is employed to analyze the change in the variable.

$$K = \frac{\mu^k}{O^k} \tag{3}$$

where

k - moment scale,

μ - normally ordered distribution

$$\mu^k = S\left(X-\mu\right)^k \tag{4}$$

where

X - random variable and

s - expected value

$$O^k = \left(\sqrt{s\left(X-\mu\right)^k}\right)^a \tag{5}$$

The scaling is used to normalize the normalizing the variable's distribution,

$$s_v = \frac{S}{X} \tag{6}$$

where

s_v - coefficient of variance.

The study determines the adjusted new normalized value. When applied in mapping fashion, the resulting value can take on values between 0 and 1. Standardization helps give a better training instance since it ensures that all of the training data shares the same field, which could range between 0 and 1. The normalizing formula is represented as below:

$$Ss = (S - S_{min})/(S_{max} - S_{min}) \tag{7}$$

where

S_s - output of the normalization procedure and S - initial value.

S_{max} and S_{min} - maximum and minimum attribute values, respectively.

The main aim of the pre-processing involves the data standardization to eliminate the restrict the duplicate information and missing statements.

During this stage of pre-processing, both the input IDS datasets and the dataset itself are standardized, and the dataset is also normalized using the missing data.

1) Feature extraction: Multilinear Component Analysis (MCA) is widely used for the extraction of most relevant feature. This is because MCA is able to extract the most nuanced features through the incorporation of these search methods. The MCA places a high value on both the redundancy that exists between features and the unique extrapolative potential that each one possesses. There are many different ways to obtain attribute information.

$$Y = V\Sigma U^{\mathrm{T}} \tag{8}$$

where

$V \in R^{S*S}$ - column orthogonal matrices of Y,

$U \in R^{M*M}$ - row orthogonal matrices of Y.

Σ - diagonal matrix

Thus an attribute obtained using the attribute function (Y) is the combination of orthogonal matrix of both rows and columns and a diagonal matrix.

The capacity of a component to supply information for the data it is believed that the image is mirrored in the variation of the projection. The following equation can be used to evaluate the performance of feature extraction:

$$C_{xc} = \frac{tc_x^j}{\sqrt{t + t\left(t-1\right)c_j^j}} \tag{9}$$

where

C_{xc} - correlation between the variables and subsets,

n - attributes.

c_x^j - correlation between the variable and attributes.

c_j^j – average inter-correlation between the attributes.

The ratio of principal components is defined as below:

$$C = \frac{\sum_{n=1}^{P}\eta_n^2}{\sum_{n=1}^{N}\eta_n^2} \tag{10}$$

A few principal components can preserve more than 90% of the overall variance of the Y data. This is the case even though the CCV ratio is calculated using only the variables.

B. Attack Detection

A model distinguishes between a huge volume of normal traffic in order to protect itself from a decentralized attack. The specified input is utilized by the attack in order to perform an estimation of the discrete scalable memory-based attack vector probability approach. Estimating the value of a random variable requires taking its distribution into account as the starting point of the process. The primary goals are to discover context and keep a close eye on relevant data as it emerges by chance. Only by persistent, day-to-day effort will it be possible to realize the goal. It is possible to have a look at the user tendencies as well as the value that is at risk. Using SVM, one may determine the likelihood of a vector value being one of several possible values. It is found by multiplying the value of the standard deviation by the constant that is used in the calculation.

$$P\left(\frac{\varnothing}{x}\right) = \frac{\left(\frac{x}{\varnothing}\right)P\left(\varnothing\right)}{P\left(x\right)} \tag{11}$$

where

P(∅) - probability distribution function,

P(x∅) - likelihood function.

P(∅/x) - evidence function.

The probability distribution provides the value of both likelihood and evidence function to find the rate at which the vector value changes.

A direct connection may be made between the likelihood and the posterior probability. The likelihood of the probability is defined as below:

$$F(x)=\left[\frac{f(x)L\left(\frac{x}{y}=y(x)\right)}{\int_{-\infty}^{\infty} f(x)(u)L\left(\frac{x}{y}=y(x)\right)(u)+u}\right] \quad (12)$$

where

$F(x)$ - prior density function.

$f(x)L\left(\frac{x}{y}=y(x)\right)$ - likelihood function.

$f(x)(u)L\left(\frac{x}{y}=y(x)\right)$ - normalizing constant.

Using the probability equation, we were able to essentially create a map of the function irregularity. Following the computation of the attack baseline probability, the results were presented. After that one is able to evaluate the significance of the attack path.

$$A=\min_{i=1:M} OS_i^* \quad (13)$$

After determining the extent of the damage caused by an attack, it is possible to classify the attack. The vector technique might make use of convolutional layers as a crucial layer in order to acquire knowledge more effectively from the input data. It does what its job title implies and reduces low-level features (kernels). However, the convolutional operator lacks rotation invariance as a property of its own. In addition, additional layers in the stack are pooling layers, which result in a reduction in the amount of data. In fact, neural networks and other fully connected structures form the core of the suggested classifier architecture. The first stage of training a classifier is called the feed phase, and the second stage is called the reverse propagation phase.

The network error is what is utilized to generate the parameter gradient, and with that, the weight matrices are updated as part of the process of context propagation. This is all done by utilizing the network error. Classifier-trained systems must be governed by large amounts of data in order to successfully accomplish classification jobs. Because the classification error is reduced in proportion to the depth of the classifier, which it is carried out.

The target is given a score based on the probabilities that are calculated for it. It is possible to calculate the difference between a single variable and a number of other variables by employing a technique known as the SVM. The recommended method can initially read and reorganize the data and then evaluate the identified technique based on the likelihood that it belongs to each class.

$$F=\Delta M-k\left(A(M)\right)^2 \quad (14)$$

where

F - feature,

M - pointed feature,

$\lambda_1\lambda_2$ - classified features.

$$\Delta M=\lambda_1\lambda_2$$

$$A(M)=\lambda_1+\lambda_2$$

The classification based on the scalable memory is defined as below:

$$F=\lambda_1\lambda_2-K\left(\lambda_1+\lambda_2\right)^2 \quad (15)$$

where

K – empirical constant.

C. Attack Mitigation

The SDN controllers are able to instruct switches regarding the destination to which packets should be sent when they make use of the Open Flow Protocol Specification. Researchers are able to conduct experiments on real-world networks thanks to a protocol standard known as Open Flow. This protocol standard details the message formats that are utilized by each controller.

This technique needs the establishment of a robust network in addition to the expensive acquisition of various assets so that it can withstand an attack. In the event of a serious DDoS, the upstream network resources need to be severed in order to preclude any reliable local response. Before subscribing to a DDoS mitigation service, there are a number of elements that need to be taken into consideration. Some of these factors include scalability, endurance, stability, and network traffic. This is only a passing phase that will resolve itself when some time has passed. The modified attack device packets will be transmitted to an analysis service at some point in time.

IV. Results and Discussions

In this section, setting up the cloud with the necessary conditions exist is part of the system model configuration. Using the deep learning model, we report the results obtained from the CTU-13 botnet and the ISCX 2012 IDS datasets. Ninety per cent of the total sample size is split evenly between the training and testing stages. To clarify, this means that 90% of the sample is used for training, while the remaining 10% is split evenly between validation and test data. Our group used a 10-cross validation technique to ensure the accuracy of the results. The current plan is to use 9 partitions from the total data samples as training samples and 1 partition as a testing sample. Each of them will be selected at random. The procedure is repeated ten times, and the final result is the mean of these ten separate estimates.

The proposed method is compared with existing machine learning methods like ANN, SVM and SOM.

TABLE I. Simulation Parameters

Parameters	Values
Packet Size	1442byte
Bandwidth	10 Mbps
Monitoring Time	20 s
Forwarding Method	Fitness route

The 99.12% accuracy achieved on the training data sample is matched by the 98.88% accuracy achieved on the test sample using Table I.

Fig. 3 shows the results of accuracy in detecting the attacks with the conventional NIDS models. The results show that the proposed NIDS has a higher rate of accuracy than the other methods. The range of accuracy using the proposed SVM classifier along with the feature extraction and pre-processing information is between the range of 95-96% and this is higher than the existing methods.

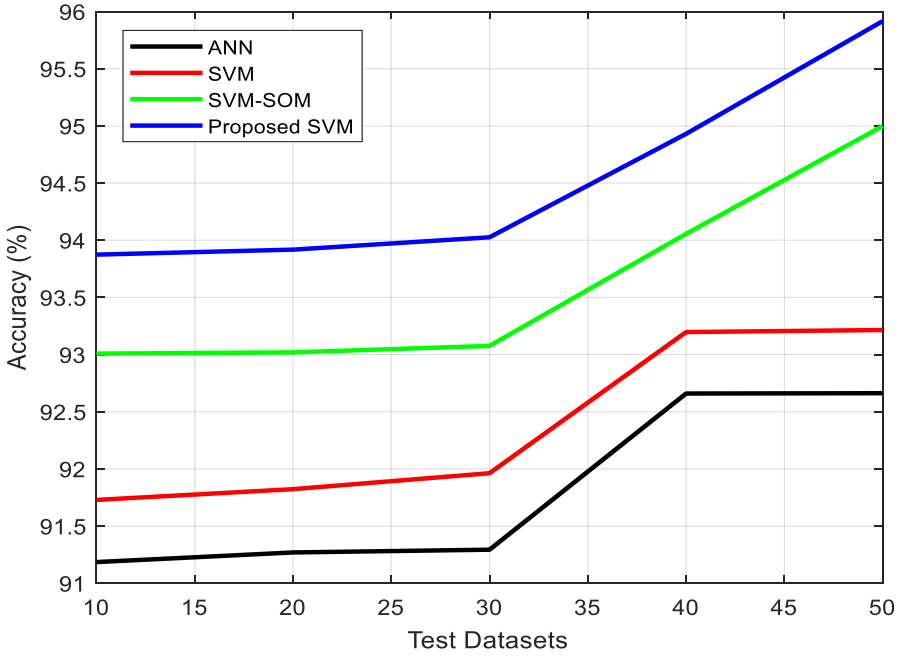

Fig. 3. Accuracy

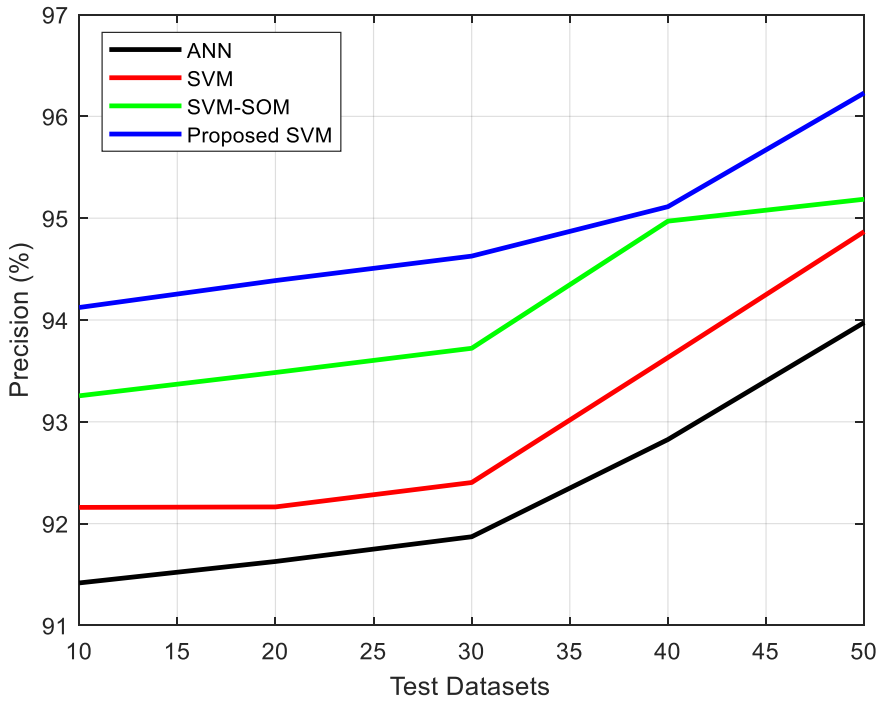

Fig. 4. Precision

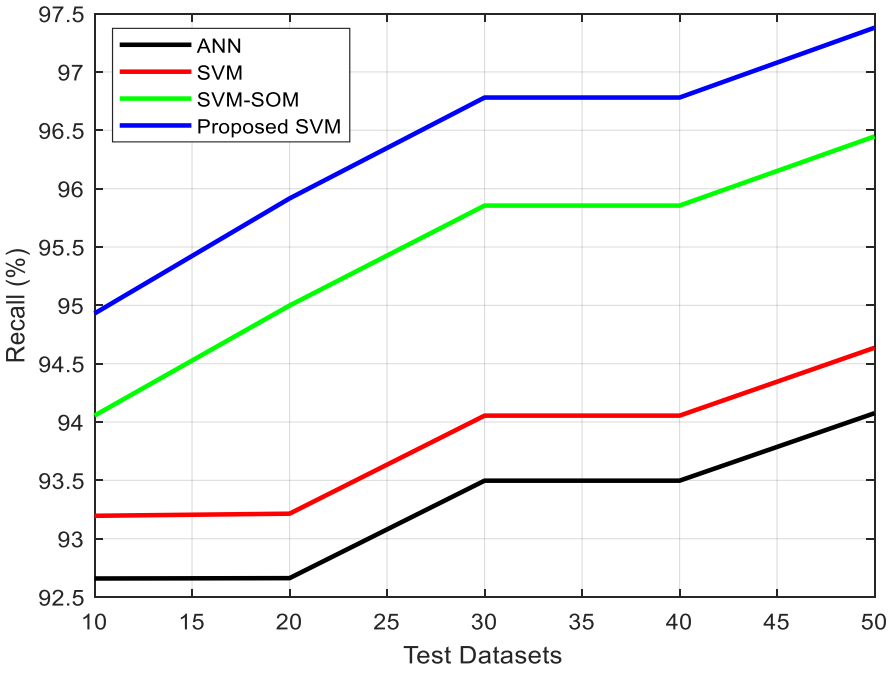

Fig. 5. Recall

Fig. 4 shows the results of precision in detecting the attacks with the conventional NIDS models. The results show that the proposed NIDS has a higher rate of precision than the other methods.

Fig. 5 shows the results of recall in detecting the attacks with the conventional NIDS models. The results show that the proposed NIDS has a higher rate of recall than the other methods.

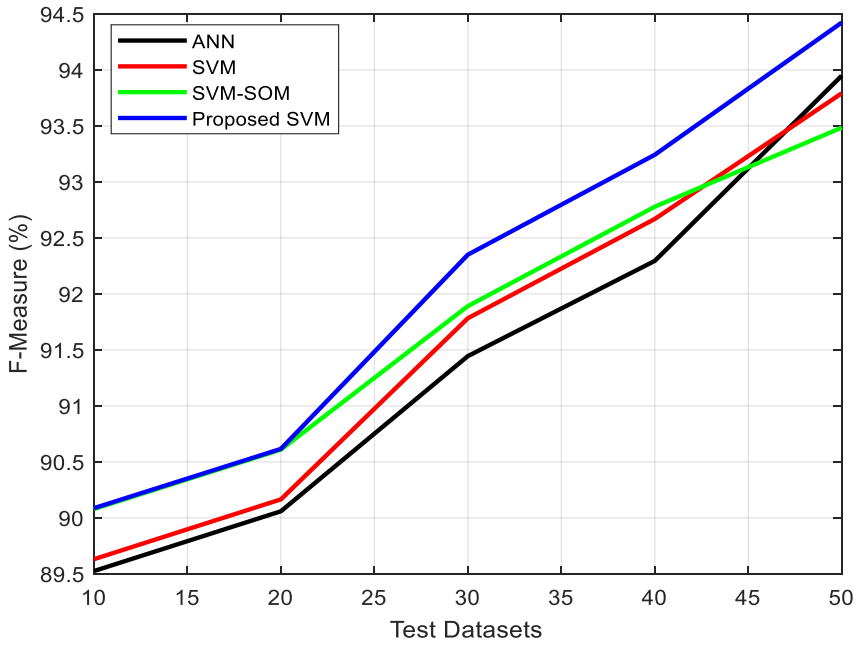

Fig. 6. F-measure

Fig. 6 shows the results of f-measure in detecting the attacks with the conventional NIDS models. The results show that the proposed NIDS has a higher rate of f-measure than the other methods.

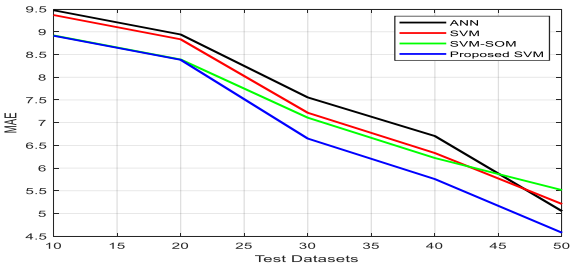

Fig. 7. MAE

Fig. 7 shows the results of mean absolute error (MAE) in detecting the attacks with the conventional NIDS models. The results show that the proposed NIDS has a reduced error than the other methods. From the results of simulation, it is seen that the proposed SVM has reduced error rate than the existing methods.

V. Conclusion

In this paper, we employ a machine learning-based IDS to protect the SDN-IoT networks from DDoS attacks. The framework involves feature selection-based classification allows the IDS to offer better defence against DDoS attacks. The resilience of the model to DDoS attacks of varied severities is tested via simulation. The simulation results show that the proposed approach is superior to other options for identifying and categorizing network intrusions. In future, the attacks on large scale network can be mitigated using multi-SDN controllers which reduce single point of failure and this cannot pose a serious computational burden on the networks.

References

[1] J. Bhayo, R. Jafaq, S. Hameed and S. A. Shah, "A time-efficient approach toward DDoS attack detection in IoT network using SDN," IEEE Internet of Things Journal, *9*(5), 3612-3630, 2021.

[2] K. M. S. Azad, N. Hossain, Md. J. Islam, A. Rahman, S. Kabir, "Preventive determination and avoidance of ddos attack with sdn over the iot networks," IEEE International Conference on Automation, Control and Mechatronics for Industry 4.0 *(ACMI)* (pp. 1-6). 2021.

[3] M. Aslam et all, "Adaptive Machine Learning Based Distributed Denial-of-Services Attacks Detection and Mitigation System for SDN-Enabled IoT," Sensors, *22*(7), 2697, 2022.

[4] H. Aldabbas, and R. amin, "A novel mechanism to handle address spoofing attacks in SDN based IoT," Cluster Computing, *24*(4), 3011-3026, 2021.

[5] A. Wani and R. Sathiya, "SDN-based intrusion detection system for IoT using deep learning classifier (IDSIoT-SDL)," CAAI Transactions on Intelligence Technology, *6*(3), 281-290, 2021.

[6] D. Javeed, T. Gao, M. T. Khan and I Ahmad, "A hybrid deep learning-driven SDN enabled mechanism for secure communication in Internet of Things (IoT)," Sensors, *21*(14), 4884, 2021.

[7] D. Javeed, T. Gao, M. T. Khan and I Ahmad, "SDN-enabled hybrid DL-driven framework for the detection of emerging cyber threats in IoT," Electronics, *10*(8), 918, 2021.

[8] A. Shoeb and T. Chithralekha, "Resource management of switches and Controller during saturation time to avoid DDoS in SDN," IEEE International Conference on Engineering and Technology (ICETECH) (pp. 152-157). 2016.

[9] R. T. Kokila, S. T. Selvi and K. Govindarajan, "DDoS detection and analysis in SDN-based environment using support vector machine classifier," IEEE sixth international conference on advanced computing (ICoAC) (pp. 205-210). 2014.

[10] P. Xiao, Z. Li, H. Qi, W. Qu, and H. Yu, "An efficient DDoS detection with bloom filter in SDN," IEEE Trustcom/BigDataSE/ISPA (pp. 1-6), 2016.

[11] S. Lim, J. Ha, H. Kim, Y. Kim, and S. Yang, "A SDN-oriented DDoS blocking scheme for botnet-based attacks," IEEE Sixth International

Conference on Ubiquitous and Future Networks (ICUFN) (pp. 63-68), 2014.

[12] T. V. Phan, N. K. Bao, and M. Park, "A novel hybrid flow-based handler with DDoS attacks in software-defined networking", Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld*)* (pp. 350-357), 2016.

[13] T. Chin, X. Mountrouidou, X. Li, and K. Xiong, "An SDN-supported collaborative approach for DDoS flooding detection and containment," In MILCOM 2015-2015 IEEE Military Communications Conference (pp. 659-664), 2015.

[14] S. Hameed, and H. A. Khan, "Leveraging SDN for collaborative DDoS mitigation," IEEE International Conference on Networked Systems (NetSys) (pp. 1-6), 2017.

[15] R. Macedo, R. D. Castro, A. Santos, Y. Ghamri-Doudane, and M. Nogueira, "Self-organized SDN controller cluster conformations against DDoS attacks effects," IEEE Global Communications Conference (GLOBECOM) (pp. 1-6),2016.

[16] R. Sahay, and G. Blanc, "ArOMA: An SDN based autonomic DDoS mitigation framework," computers & security, *70*, 482-499, 2017.