International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Distributed Dos Attacks Detection Based on Machine Learning Techniques in Software Defined Networks

M. Kandan[1], P. Shobha Rani[2], T. Sathiya[3], K. Bhanu Rajesh Naidu[4], M. Maheswari[5]

Submitted: 07/01/2024 **Revised**: 13/02/2024 **Accepted**: 21/02/2024

Abstract: Manageability, scaling, and enhanced efficiency are all benefits of Software Defined Networking (SDN). However, if the controller is prone to DDoS attacks, SDN presents a unique set of security challenges. DDoS attacks have resulted in massive economic losses for civilization. They have evolved into one of the most significant challenges to Internet security. In a cloud and large data world, most existing detection approaches based on a single function and defined parameter values are unable to detect early DDoS attacks. The network connectivity and integration capacity of the SDN controller are overloaded while it is vulnerable to DDoS attacks. The high amount of flow that the controller is producing for the attack packets causes the switch flow database ability to fill up, which lowers network output to a critical threshold. Artificial Neural Network (ANN) techniques were used in this paper to detect DDoS attacks in SDN. The test findings showed that the highest accuracy rate in DDoS threat detection was achieved when an ANN classification method was combined with wrapper function selection applied. The proposed system is tested against existing benchmarks on a current state-of-the-art Flow-based dataset. The results demonstrate how Feature Selection (FS) strategies and the ANN approach may both reduce processing times and reduce processing difficulties in SDN DDoS attack detection.

Keywords: *Distributed Denial of Attacks (DDoS); Software Defined Networks (SDN); Artificial Neural Network (ANN); Machine Learning; Feature Selection*

1. Introduction

Computer networks, chips, virtual networks, and handheld devices have all been subject to increased scrutiny in recent years [1]. Data network security has garnered a lot of interest as an effective medium for knowledge sharing. Distributed Denial of Service (DDoS) attacks have been a long-standing problem in computer network security. DDoS attacks are a long-used type of network attack. It's in charge of a swarm of robot systems that send a variety of irrelevant network design challenges to a single host [2]. It consumes and drains the server's energy, preventing regular people from accessing the target host's services [3]. Although the DDoS attack mode is easier, it has much greater network disruption capacity than other network attacks. Furthermore, in recent years, this conventional attack strategy has continued to do significant harm to the Internet, with the rate of attack, loss incurred, DDoS sophistication, variation, and challenge of protection both increasing more than before.

Currently, the most popular strategies for defending against DDoS attacks in a conventional network environment are attack detection and attack response [4]. Attack signatures, congestion patterns, protocols, and source addresses serve as critical foundations for identifying “DDoS attacks, resulting in a successful detection method. Misuse-based and anomaly-based detection” are the two types of detection models that can be used. Misuse-based identification is a feature-matching algorithm-based method of detecting misuse. To determine if a DDoS attack takes place, it compares the collected and extracted user experience features to a catalogue of established DDoS attack features. Monitoring devices make use of anomaly-based identification. Tracking systems may evaluate if target system's behaviors and a user's activities differ from an expected characteristics, and if an attack has occurred, by determining the specific environment and the individual's normal decision process. After a DDoS attack has been initiated, network traffic can be appropriately filtered or blocked. Attack focus on accessibility to devices is restricted to the highest degree

[1]*Assistant Professor, Department of Computing Technologies, School of Computing, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur 603203, Tamilnadu, India.*
Email: kandanm@srmist.edu.in
ORCID: 0000-0001-8311-4182
[2]*Professor, Department of Computer Science &Engineering, R.M.D Engineering College, Kavaraipettai 601206*
Email: psr.cse@rmd.ac.in
ORCID: 0000-0002-5465-8444
[3]*Assistant Professor (Sr.G), Department of CSE, Sona College of Technology, Salem.*
Email: sathiya.t@sonatech.ac.in
ORCID: 0000-0003-0033-1515
[4]*Assistant Professor, Department of Computer Science & Technology, Madanapalle Institute of Technology & Science, Kadiri Road, Angallu, Madanapalle, Andhra Pradesh, India.*
Email: bhanurajesh9493@gmail.com
ORCID: 0009-0009-5157-3910
[5]*Associate Professor, Department of Computer Science and Engineering, Panimalar Engineering College, Chennai*
Email: m.mahe05@gmail.com
ORCID: 0000-0001-8328-3808

allowed in order to reduce the effect of a DoS network attack.

DDoS attack identification involving cloud computing networks and software-oriented networks have become more in demand as a result of the development of cloud infrastructure components and software-based networking (SDN) systems [5]. Cloud computing, as a modern computing paradigm, offers efficient distributed computing, huge storage, and a wide range of service capabilities [6]. It has been an essential tool for addressing big data issues [7]. As a result, building a cloud infrastructure is a critical step in ensuring the efficiency, availability, and security of cloud computing [8].

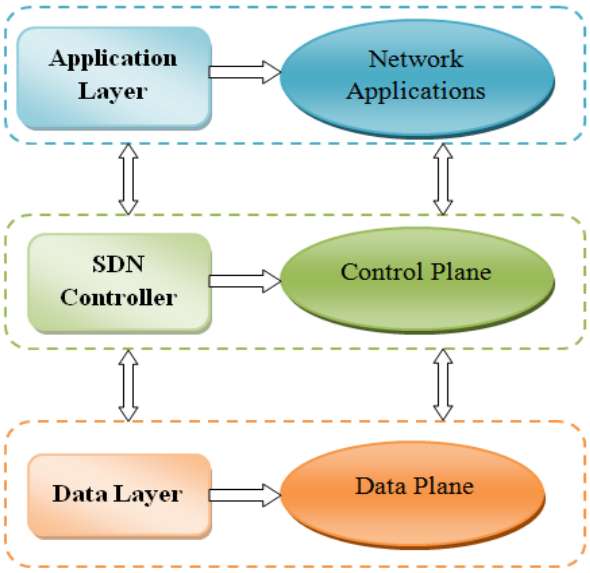

Fig. 1.1. Software Defined Networking (SDN) Architecture

Common networks usually consist of a large range of computers with complicated protocols, such as host machines, switches, routers, and packet forwarders, firewalls, and so on. Manually switching the control depending on conditions is the responsibility of network devices. SDN [9] is a representation of programmable networks based on recent innovations. An SDN network's intelligence is logically controlled, and the network is divided into two sections: “control plane and data plane”. Figure 1.1 shows framework for SDN architecture. A control plane manages the whole network using the Open Flow protocol, which provides network flow information. Packets are able to transmit by the data plane in accordance with controller standards. Security is the ways to reach addressed in SDN [10], as SDN outperformed the network environment sector. There are several different kinds of threats, each categorized according to the device's vulnerability. DDoS attacks and intrusion detection systems (IDS) are examples of attacks. When a group of attackers performs a DDoS assault on network hardware, important effects will occur. SDN has a large number of controllers, each with its own set of applications that require additional protection.

Machine learning has recently been evaluated and the results of defence [11]. Machine learning has been commonly used to establish attack detection models [12]. In a conventional network setting, cloud environment, and software-defined network design, artificial neural network approach plays an essential part. An explanation for this is that the machine-learning approach could extract valuable information from data and integrate it with previous knowledge to classify and forecast new data. As a result, machine-learning models could be more accurate than standard detection systems [13]. The standard network environment, cloud environment, and SDN infrastructure are all considered to provide attack detection for DDoS security mechanisms, according to the above review of protection steps. As a result, research into the use of artificial neural network algorithms to detect DDoS attacks is crucial. However, DDoS attack data is often dynamic and uncertain and diverse, and context traffic scale has a larger effect on the detection method, lowering a model's detection accuracy.

SDN management, security, and optimization will benefit from machine learning-based methods that are more complex, effective, and intelligent. For the purpose of ensuring network stability, it is essential to monitor DDoS attacks and execute necessary measures. With the goal to construct an autonomous structure that may comprehend and function, SDN information flow might be integrated with artificial neural network based detection of DDoS attacks technologies integrated into SDN topologies. Furthermore, SDNs with embedded machine learning applications could be used as a proposed methodology for establishing a stable framework in 5G network integration tests.

The remaining part of the paper has been structured as: Related works are described in the section 2. The proposed system is discussed in Section 3. The section 4 summarizes the performance analysis of our proposed system, including the testing configuration, research scenario, and analysis results. The paper's conclusion and a description of its prospective future applications are provided in section 5.

2. Related Works

DDoS attacks will seriously damage a network and result in significant financial damages for the group that has been targeted. One of the most popular methods used by hackers to launch cyber-attacks is through social engineering. In recent years, studies have suggested a wide variety of attack identification methods aimed at reducing the damage caused by DDoS attacks. The term "traditional network framework" refers to an Internet context based on an open system connectivity proposed methodology. In this context, Saied et al. suggested an artificial neural network-based technique for identifying known and unknown DDoS attacks [14]. Bhuyan et al. suggested an observational test method for detecting low- and high-rate DDoS attacks [15]. Tan et al. [16] suggested a multivariate correlation analysis-based DDoS attack detection tool. Yu et al. suggested a traffic correlation coefficient-based DDoS attack detection system [17]. Wang et al. investigated the features of DDoS botnets in detail [18]. The Jpcap API was used by Kumar and others to track and evaluate DDoS attacks [19]. Khundrakpam et al. [20] suggested an application-layer DDoS attack detection model based on entropy and ANN.

The network infrastructure platform of cloud computing as the main infrastructure is referred to as the cloud ecosystem. Karnwal et al. provided the systematic approach for XML and HTTP DDoS attacks on cloud data storage [21], while Sahi et al. introduced a test and security system for TCP-flood DDoS attacks in the cloud setting [22]. Rukavitsyn et al. suggested cloud-based self-learning DDoS attack management system. The word "software-defined network" defines a modern network framework, which includes the OpenFlow networking protocol which uses a controller to specify data sharing requirements for routers and switches. Ashraf used ML identification tools to identify DDoS attacks on the network in this respect [23]. In the SDN network, Mihai–Gabriel suggested an intellectual elastic risk evaluation approach focused on neural networks and risk theorem [24]. In software-defined networks, Yan et al. suggested efficient controller management approach to minimize DDoS attacks [25]. Chin et al. [26] suggested a DDoS flood attack approach that uses SDN to selectively detect packets. Dayal et al. looked at how DDoS attacks behaved under SDN [27]. Under the SDN network, Ye et al. suggested a technique for detecting DDoS attacks using SVM [28]. It apart from the above detection approaches, certain effective cryptographic algorithms could be used to protect the system's privacy.

Ye et al. [16] integrated traffic information about networks from providing systems on the data layer using the controller. To detect DDoS threats, SVM was implemented to extract six-tuple key parameters from the switch traffic database that are related to DDoS attacks. There was a high rate of detection accuracy identified. However, it has been stated that the test accuracy rating for the attack flow using the “Internet Control Message Protocol (ICMP)” was quite low. Many security requirements are required, according to Xue et al. [17], since the SDN controller controls several switches in the data plane. They claimed that existing equipment and applications that had not yet been modified to the SDN network could not provide protection. DDoS attacks on data plane switches, in general, cause severe damage to the network's SDN platform's integrity. Using SVM-optimized C and G requirements, an integrated validation-genetic method was employed to identify DDoS attacks.

To reduce security risks, Nanda et al. [18] suggested using machine learning techniques that were educated on historic network attack data to detect possible threats links and targets. Various machine learning techniques were used in this study. These were claimed that by using machine learning techniques to estimate malicious users in the data plane, it would be helpful to define them. The SDN controller can fast and efficiently wrote new terms to detect an attack after establishing user identities, which is crucial for the network's reliability and continuity. A two-step analysis was performed by Banerjee and Chakraborty [20]. In the first step, Neural Network and artificial intelligence techniques are learned to differentiate between the attack and regular traffic. A three-way handshake system was then used to identify the intruder. An Access Control List was created to block the detected intruder.

In SDN-driven Cloud Infrastructures, Mowla et al. [21] proposed Cognitive Switch-based DDoS Signalling and Reduction. To classify traffic, Classifier and Multilayer Perceptron approaches were used, and this classification caused the introduction of cost - effective on OpenFlow switches, preventing new types of flooding attacks and detecting and defending against any potential DDoS attacks. DDoS attacks in SDN networks are explored in this review, and ML techniques with feature selection methods are proposed for detecting attacks. As a result, the aim is to develop high-fertility ML-based DDoS attack detection systems for SDN networks. The detection of a DDoS attack on the SDN controller and data plane switches is critical for network stability and detection of

legitimate traffic during the attack. If the controller detects attack traffic, it is easier for the controller to avoid the attack by adding new terms to the flow tables of data plane switches. This gives a huge advantage in terms of aiding an assault. To detect DDoS attacks, this paper suggests using feature selection approaches in conjunction with machine learning techniques. It assumes that our strategy will help SDN track DDoS attacks more effectively.

3. Proposed Work

Artificial neural networks are a great deal of often used machine learning approaches in IDS. The foundation of ANN techniques is optimal feature selection and network parameters. The matrices could be used to track patterns and developments, and the functions are the matrices. The characteristic of neural networks arises from their connected, widely synchronized processing units that are organized in layers. Studying a biological neuron, which is an essential part of biological neural systems like the cortex, brain stem, and peripheral ganglia, provided the inspiration for both its development and operation. Weighted input signals are produced by neurons or nodes and are combined and processed by an amplification method to reduce the output of the neurons prior having sent to nodes in the subsequent layer [29]. The findings are then used in a classification or regression analysis. ANN methodology has been applied to a wide range of tasks, yielding significant improvement in detection, logic, and prediction.

Figure 3.1 shows the techniques involved with using feature selection techniques and ML algorithms. In addition, the characteristics and classes of the dataset, as well as how the data is collected from the dataset, are discussed in this section. The features in the dataset, which were generated as part of the analysis, were derived from the most important metrics for the SDN architecture's continuity. Artificial intelligence is used to classify attack data following the collection of DDoS attacks. After attributes in the sample are chosen to use techniques, the accuracy of the classifiers on the obtained feature space was checked.

Fig. 3.1. Proposed Framework for DDoS Attacks Detection in SDN

In this analysis, artificial neural network with FS approaches are include to identify DDoS attacks. Machine learning (ML) is a method of assuming the unknown by applying the effects which can be determined from the data that is currently available by using statistical and computational techniques. In the research, several ML techniques are being established. The classification of systems might be described by the kernel, distance, neural network, and probability-based parameters. Current study indicates that no single paradigm works optimally for all categorization functions. Specific ML methods, including ANN systems, have demonstrated good results for solving classification problems, according to the literature. Numerous properties could be included in the datasets used to train algorithms for machine learning. Given that a few of these features may significantly affect the classification result, others could not have any effect at all. The system will operate faster and have greater value if features that don't affect classification are used. In order to make possible more highly effective features, characteristics which are of no impact on classification are to be removed.

3.1 Feature Selection Methods

As function filtering tools, the filter, attached, and wrapper methods were included. The structure of feature selection strategies is indicated in Figure 3.2. The filter approach is more involved with features' inherent characteristics, while the wrapper method is more concerned with features' usefulness dependent on classifier performance. By combining the advantages of the filter and wrapper methods, the embedded research aims to improve the classification framework or system's efficiency.

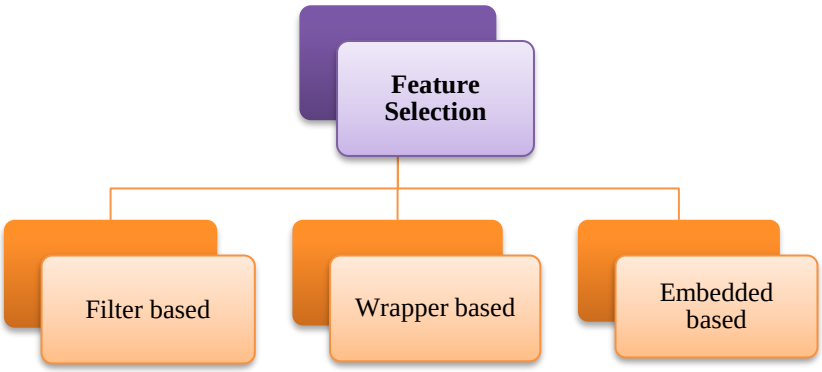

Fig. 3.2. Methods of Feature Selection

To accomplish their objectives, each feature selection process uses a different algorithm. Our classification technique is a binary classification problem with no incomplete data in our dataset. In order to collect filter-based features, the instance-based learning-recommended relief filtering strategy was chosen. The greedy search-based Sequential Forward Selection method has been chosen to be one of the wrapper-based feature selection methods due to its high effectiveness in identifying an optimal function subset. The Lasso or L1 technique was chosen as the embedded feature selection strategy because it applies an advantage against complexity to avoid over-fitting and enhances the algorithm's optimisation efficiency.

3.1.1. Filter based Feature Selection

Statistical approaches are used to measure how much the filter-based feature selection model achieves its goals of the feature outcomes. As shown in Figure 3.3, the optimal feature set is chosen from the scored features in order to develop a reduced feature set. The Relief method is used in the research, which is a filter-based feature selection technique. The Relief method predicts the distance between classes and the interaction between every set of data in the database and other samples in the same class. It generates a framework and goes through the feature selection as a part of this estimation [30].

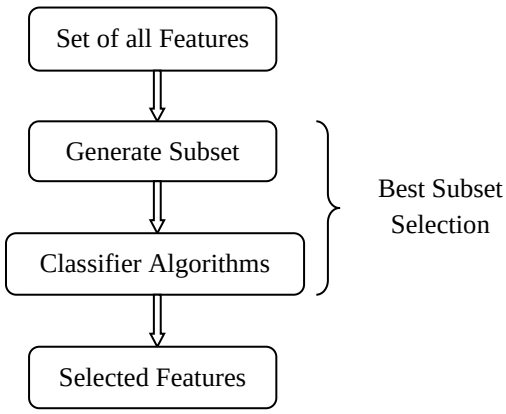

Fig. 3.3. Filter based Feature Selection

3.1.2. Wrapper based Feature Selection

The wrapper approach tries all features in the classification algorithm to find the most optimal characteristics. The process is completed when the optimal subset of features is found, and a smaller dataset is generated (Figure 3.4). It should be more effective than statistical techniques, but depends on the classification model, it may be one of the wrapper-based function selection techniques, sequential forward floating selection, is implemented in this research. Prior to the classifying performance reaching a greater value, the same direction is employed to select the function [30].

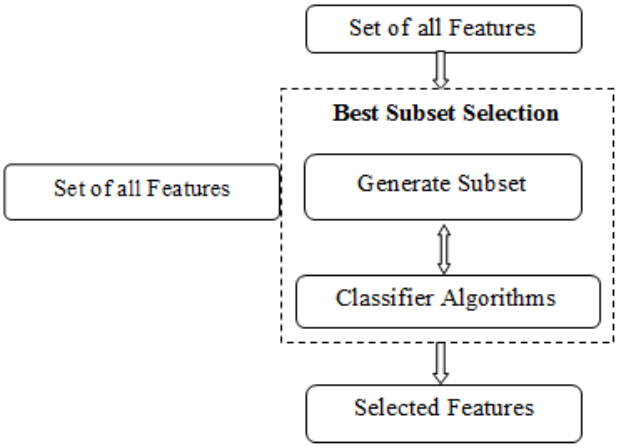

Fig. 3.4. Wrapper based Features Selection

3.1.3. Embedded based Feature Selection

Feature selection techniques that are embedded in the classification models are known as embedded-based feature selection techniques. These approaches identify features by determining which ones would have the greatest impact on the model's accuracy (Figure 3.5). They were created to bring together the benefits of filter and wrapper-based approaches. In this method, feature selection and classification are performed out simultaneously by a learning methodology utilising its variable selection mechanism [30]. This analysis applies

the Lasso algorithm, a few of the embedded-based feature selection methods.

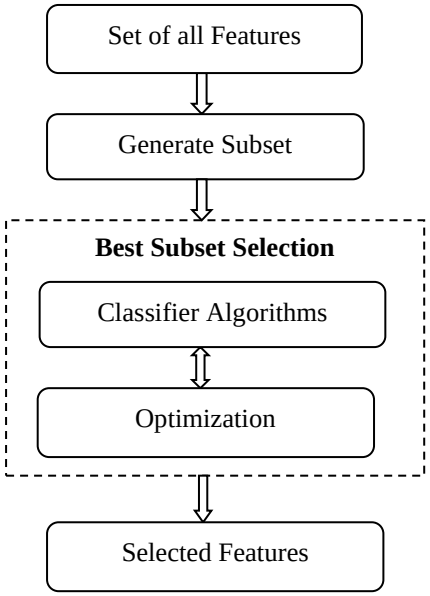

Fig. 3.5. Embedded based Feature Selection

3.2 Artificial Neural Network (ANN)

Artificial neural networks, which behave similarly to neurons in our central nervous system, are crucial networks in nearly every data analysis and computational use. ANNs have strong integrated elements in complete synchronization to meet an objective purpose, similar to biological connectivity between neurons. They are used in a variety of applications, such as pattern recognition and classification, identification problems, adaptation and regulation, and so on. Figure 3.6 shows a standard artificial neural network system.

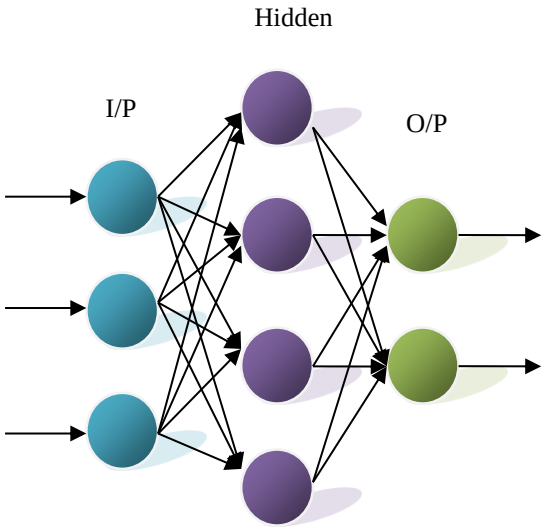

Fig. 3.6. Artificial Neural Network Architecture

The input, hidden, and output layers of an ANN are described in the above architecture. The number of nodes in the input layer of a given issue statement is referred to as the function vector. There are approximately as many groups or classes where the optimal performance can be assigned as there are nodes in an output layer. Based on where the brain tumour is malignant or benign, the output nodes for a brain tumour diagnosis and classification question could vary from two to three. An attack detection issue can have a maximum of two output layers, each containing a genuine or contaminated packet of data. A randomised load is allocated to each node at the initial phase of the updating procedure, and techniques connect the input and output layers with the hidden layer. In order to lower the error at the output (2), the weights have been changed by the weight update equation given in (1) once the first iteration is done. Figure 3.7 shows the workflow of proposed system.

$$H(a, b) + \beta \quad (1)$$

$$M\{m^2[i]\} = M\{(a[i] - g[i])\} \quad (2)$$

where $M\{m^2[i]\}$, stands for the expected mean squared error function, a[i] stands for the intended output, and g[i] stands for the received output. This iteration procedure is driven by a classification algorithm that can use a propagation rule function. Training is a method of becoming familiar with the applied neural network in order to create feature vector structures and, ultimately, classify the inputs into a predetermined output class. Neural network learning aims to minimise the loss function by applying a minimization law, as seen in figure 3.6. Equations (3) and (4) illustrate methods to calculate the first and second derivatives of the provided loss function in order to determine the approach to the specified criterion objective. The term H* denotes the minimum dimension of the given loss function.

$$\int t\,(h) = at/ah_x \ (y = 1,\ldots,n) \quad (3)$$

$$\int t\,(h) = a^2t/ah_x \,.\, ah_y \ (x, y = 1,\ldots,n) \quad (4)$$

Between the points 1 and 2 in the above diagram, the minimum feature of loss is current. For one-dimensional loss minimization functions, the golden segment and Brent's approach are widely used algorithms. However, most real-time difficulty descriptions, like the problem goal of this study, necessitate a multi-dimensional search and reduction approach. The following paragraph delves into multi-dimensional optimization approaches and methodologies in sufficient complexity, which will significantly help in the implementation of the proposed framework for attack defence.

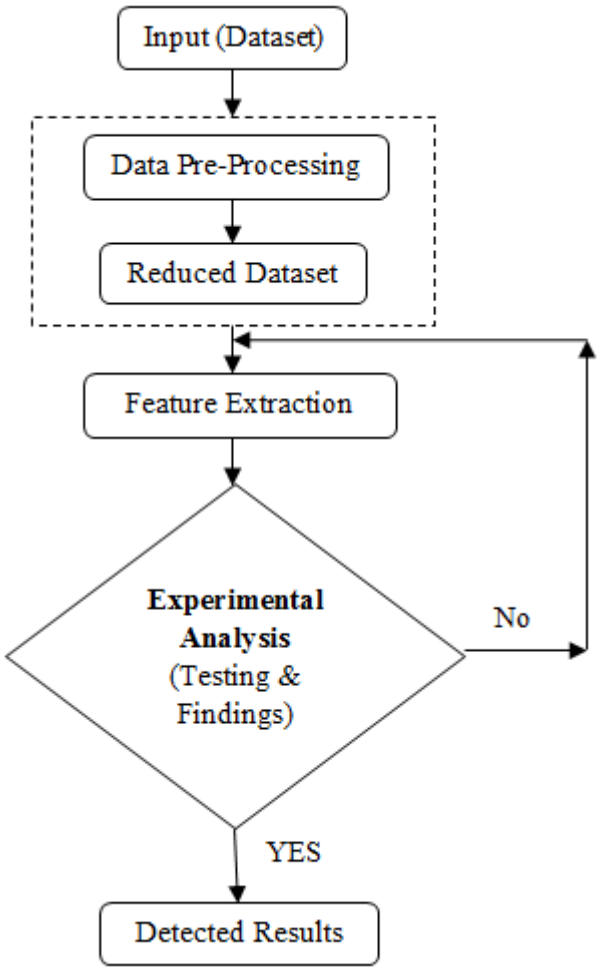

Fig. 3.7. Flow Chart of Proposed System

It depends on the average of squares iteration and is also referred to as damped least squares approach. This algorithm also avoids interfering with the Hessian matrix and its inverse results, instead focusing on the Jacobian Matrix. The Jacobian matrices are characterised by an error rate of the type.

$$t = \sum p_q^2,\ q = 0,1,2, \dots ,z \qquad (5)$$

This approach is useful for training moderate neural methods since the damping f ratio is reduced or improved to get to the convergence rate, and the acceleration to gradient is very high. As the length of the Jacobian matrix doubles, the condition occurs, increasing the expense and difficulty of large networks.

This work proposes and implements a three layer ANN feed forward system using a form of principle component analysis (PCA) for dimensionality reduction. The approach depends on research and evaluations seen regarding ANN usage architectures and their benefits. When extracting Jacobian matrices for massive neural network techniques such as the kinds applied in this research, the goal of lowering the dimensionality is to minimise the memory and computing responsibilities of an LM learning algorithm.

4. Experimental Results and Analyses

The topology includes multiple virtual machines (VMs) with "Ubuntu 18.04, 1 GB RAM, and 1 CPU configuration, two VM switches, one Open vSwitch, sFlow, and InfluxDB docker files, and a Phyton-based open access OpenFlow/SDN controller". The OVS switch was connected by a VM switch that connected sFlow and InfluxDB, all of which were installed in the same virtual machine. A bridge was used to link the VM switch to the OVS switch from the users' perspective (br0). Network studies from the OVS switch were gathered, and a sFlowDocker image was used during both the attack and regular traffic to evaluate the impact of the attack traffic. In order to achieve this, an open-source time-series framework called InfluxDB was implemented to log network statistical data into an object with a timestamp using JavaScript.

Dataset Description

A protocol-based attack scenario was used. Using a "hping3" packet generator, a DDoS attack dataset was generated using 3 categories of flooding attacks: "Transmission Control Protocol, User Datagram Protocol, and Internet Control Protocol". The Hping3 tool was built on the network's PC1 and designated as the intruder, with PC6 designated as the victim. For each protocol-based flooding attack, PC6 has the IP address 10.0.0.6, with a payload size of 512 bytes and constant packet speeds of 2108 packets per second. As a consequence, the number of flow modulation messages received from a controller reaches 1527 per second, resulting in 7254 flow table entries. The processing, bandwidth, CPU power, and communication capacity of the controller were all overloaded as a result of the generated DDoS attack packets. Because of the unsustainable flows that the controller produces as a result of assaulting packets, the switch's flow table entries are complete. Any of the submitted TCP, UDP, and ICMP packets was subjected to a 15-minute simulation. This approach ended in the generation of a dataset with 73,027 samples linked to network traffic flow during a DDoS attack. PC2 and PC5 had their TPC, UDP, and ICMP traffic data registered in order to collect standard traffic data for training and testing. This method yielded a 64,000-sample dataset containing information about natural network traffic flow. As a result, a new dataset was developed, which included 12 features and 129,000 samples. The dataset's class and twelve attributes are listed below in table 4.1, along with their descriptions.

Using feature selection techniques, the dataset generated for the analysis was decreased, and classification was performed on these results. Classifiers also include several machine learning techniques. There were two phases of the

experimental analysis. Many of the features in the dataset were used to train and implement classifiers. To evaluate the efficiency of the chosen platform [31] in the first step, the dataset was trained and tested collections by the k-fold cross validation (k = 10) methodology. Feature selection strategies were used in the second level. Relief, sequential forward floating filtering, and Lasso algorithms were used as filter, wrapper, and embedded function selection approaches, respectively, to pick the most successful element in the entire dataset. These feature collection approaches resulted in three separate datasets. Machine learning algorithms were used to evaluate the best output combination between 1 and 12 features for each of the datasets.

Table 4.1. DDoS Attacks Dataset Description

Features	**Description**
Ifinpkt	The data plane sends packets into the device.
Ifoutpkt	The out packet regulates packets exiting the device based on the flow rule sent by the controller with the "Packet-out" code.
Hits	The no of packets that meet the controller's previously defined rules which are contained in the OVS switch's flow table.
Masks	The super flow mask stats are shown in the "masks" row. For data paths that do not use super flow, this row is excluded.
Miss	A match is found in the flow table when a new packet reaches at the transmitting devices (switch, modem, etc.). If a packet arrives that does not meet any flow feedback (flow miss), it is forwarded to the dispatcher, which would generate a new flow law.
OpenFlowfrom	The controller has received a flow adjustment message. The controller sends one of the most important signals. Switch state for the current flow is modified as a result of this post.
Cpu_util	The computer's central processing unit uses arithmetic operations to interpret and measure the data that comes in. The percentage of time this unit is used varies based on the computer's computing power. If the percentage is high, the machine is performing at or above its full capacity for the number of running applications.
Mngmnt_interface_inpkts:	The control plane receives a packet.
Mngmnt_interface_outpkts:	The control plane is losing packets.
OpenFlowto	Most of the controller's most significant messages are this one. Switch state for the current flow is modified as a result of this post.
OVS Flow	The total number of data flows in a data path.
OVS Lost	The number of packets that were sent for processing but were never processed.
Class	Traffic class. Labeled data were used in this analysis. In other words, supervised learning is targeted at four distinct forms of attacks: "natural traffic," "ICMP Flood," "TCP Flood," and "UDP Flood."

4.1. Performance Analyses

The proposed ANN model's efficiency is contrasted to that of other ML approaches. On the basis of cross validation, the efficient ML approach was chosen. The dataset is used to analyse the data and implement the performance of proposed ANN frameworks. Our suggested method operates well and is capable of correctly classifying and predicting threats.

Confusion Matrix: It is a simple reference guide that is used to demonstrate how classification algorithms work. The confusion matrix format is defined in the following matrix table 4.2. The ANN confusion matrix is illustrated in Figure 4.1.

Table 4.2. Confusion Matrix

Classification	**Class**	**Predicted**	
		Positive	**Negative**
Attack	**True**	TP	TN
Normal	**False**	FP	FN

True positives and false positives are the terminology used to characterize true positives and false positives, respectively. Similarly, FN emerges for false negative and TN emerges for true negative.

Accuracy: Accuracy is the frequency at which information is correctly classified, or the circumstances in which legitimate attack data is identified as such and legitimate data as normal.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Precision: The percentage of correctly identified positive information to all positive information is known as precision.

$$\text{Precision} = \frac{TP}{TP+FP}$$

Recall (or) Sensitivity: Recall is the ratio of correctly classified positive information to total attention in a class.

$$\text{Recall} = \frac{TP}{TP+FN}$$

F1-Score: The weighted average of precision and recall is used to calculate the F1-Score.

$$\text{F1-Score} = \frac{2*Recall*precision}{Recall+Precision}$$

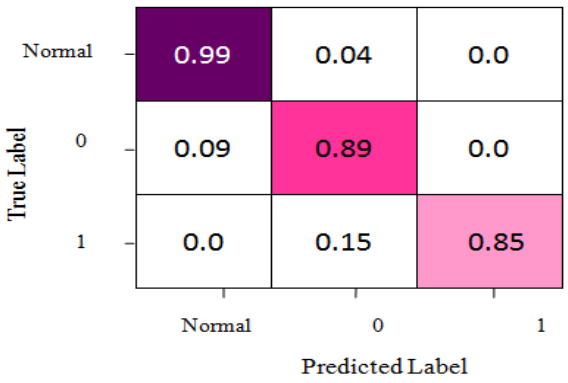

Fig. 4.1. Confusion Matrix of ANN Classifier

For the dataset in this paper, data from both normal traffic and DDoS attack traffic were initially investigated on SDN controllers. At this point in the analysis, no feature selection techniques were used. Both dataset functionality were used in the analysis, and two separate reduction approaches were used (Table 4.3). Although the classification results obtained using all features are appropriate, the efficiency of the system improved when feature selection approaches were used.

Table 4.3. Comparison of Feature Selection Methods with ANN Classifier

Classifier	Feature Selection Method	Accuracy (%)	Recall (%)	Precision (%)	F1-Score (%)
Artificial Neural Network (ANN)	Filter	95.14	90.13	93.78	91.43
	Wrapper	89.37	85.91	89.05	89.54
	Embedded	93.46	89.25	90.13	90.63

When the wrapper function selection system was used, the KNN classifier had the highest accuracy rate (98.3 percent) when the test data was extracted to the ML techniques (see Table 4.3). This accuracy rating was obtained after six features were qualified and tested. After using feature selection algorithms, it found that the output ratios obtained by training classifiers over a dataset with 12 features improved. It created a comparative table to compare our results to those of other researchers. Figure 4.2, 4.3 and 4.4 illustrates the analyses as well as those mentioned in the table.

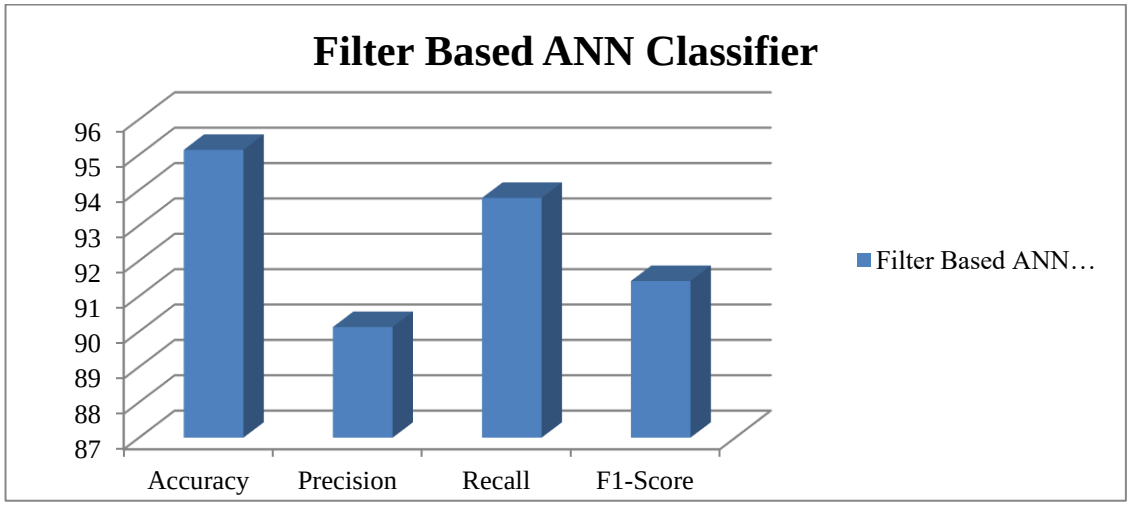

Fig. 4.2. Performance metrics of Filter Based ANN Classifier

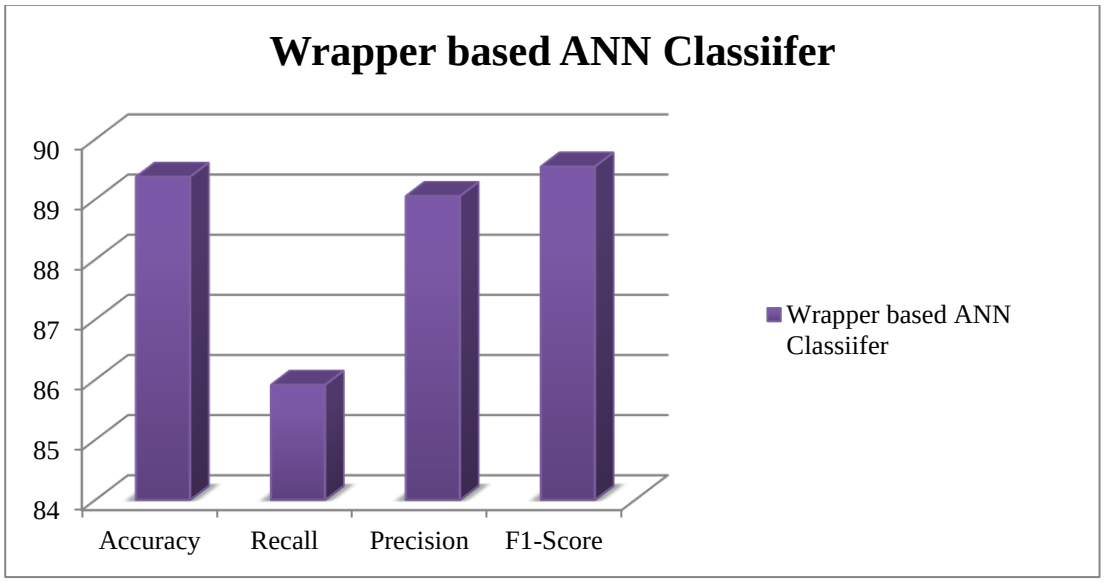

Fig. 4.3. Performance metrics Analysis of Wrapper Based ANN Classifier

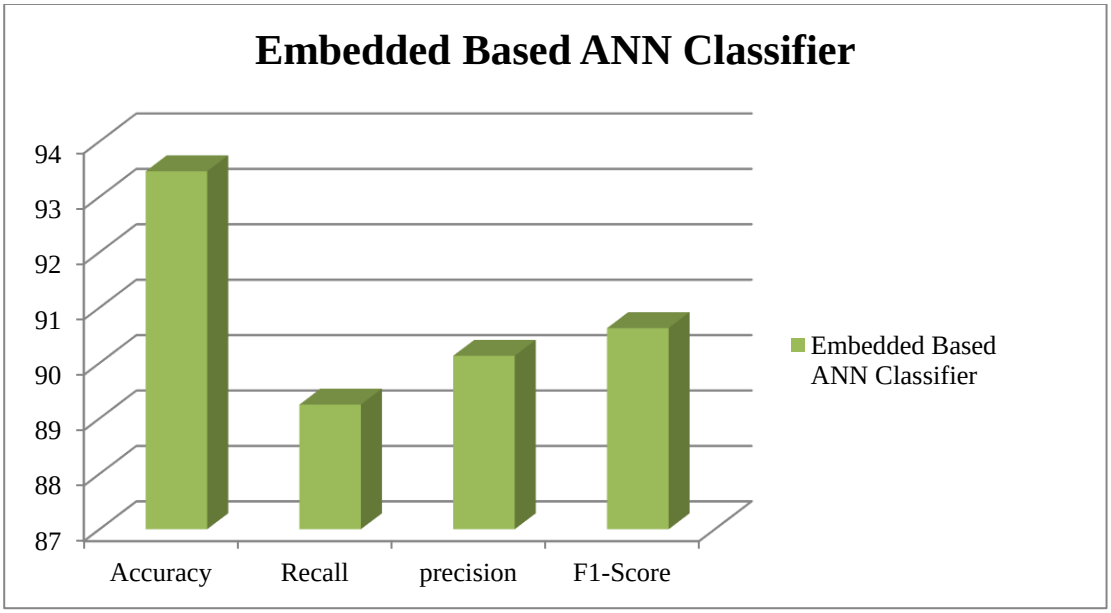

Fig. 4.4. Performance metrics Analysis of Embedded Based ANN Classifier

The findings show that using machine learning methods, the SDN structure was effective in detecting DDoS attacks. A stable and powerful network framework could be built using the strategies to be designed on SDN architecture. On wide networks, the controllers' efficiency can be enhanced by capturing packets in the traffic and comparing them to previously learned packet flow data. Another choice for detecting DDoS attacks on the dataset is to use a two-stage method. While the controller processes the incoming flow data, it is copied to another unit and evaluated there. To get some network output, the module must tell the controller whether or not there is an issue with the data coming in. As a consequence, while the controller manages only network operations, the other module takes on the function of an attack detection system.

In an SDN topology, several SDN controllers may also be arranged in a hierarchical structure. In an SDN node with a multi-controller setup, DDoS attack traffic identification is faster than with a single controller since the network traffic load is distributed among controllers. At this stage, the position of the controllers in the network is also crucial. Through maximizing resource usage, the controllers' efficiency on network components in the data plane will improve and delays will be minimized until they are correctly positioned in the network. However, for network operation reliability, data flow between controllers cannot be disrupted in a network with multiple controllers.

In DDoS attacks, the attacker can help to avoid controllers from communicating with each other. If contact between controllers is disrupted, the SDN architecture can crash. In our research, DDoS attack traffic emerged by applying a

simple SDN architecture with a single controller, and a dataset was created. ML techniques were then employed to categorise DDoS attacks. The use of a single-controller or multi-controller setup in the network topology does not lead to significant improvements in technique when extracting a dataset from SDN.

5. Conclusion

Machine learning techniques were used to evaluate SDN-based detection systems developed for DDoS attacks in this research. Through processing flow data, the first proposed solution ensures that attacks are detected with 99.1 percent accuracy, regardless of the type of traffic. The second solution, among the suggested schemes, separates DDoS attacks into two categories: normal traffic and attack traffic. ANN architectures will execute this monitoring with 95.14 percent sensitivity, reducing the controller's workload. Using the feature selection approaches used in the analysis, a subset of 12 features was chosen and trained using classifiers. The algorithm or the threshold value given to the algorithm defined the number of features selected. Various types of features may be chosen and trained by the classifier by adjusting the threshold value. The accuracy can vary based on the number of features. In general, this proposed system found that all network performs well above 90% of the time, and the algorithms used for this dataset work well. At the same time, this technique will detect network surfing, attacks within layers, and malicious software on SDN. The following methodology should be used to preserve and strengthen the SDN framework.

References

[1] Cheng, J., Zhang, C., Tang, X., Sheng, V. S., Dong, Z., & Li, J. (2018). Adaptive DDoS attack detection method based on multiple-kernel learning. *Security and Communication Networks*, *2018*.

[2] Cui, J., Zhang, Y., Cai, Z., Liu, A., & Li, Y. (2018). Securing display path for security-sensitive applications on mobile devices. *Computers, Materials and Continua*, *55*(1), 17.

[3] Pimpalkar, A. S., & Patil, A. B. (2015, March). Detection and defense mechanisms against DDoS attacks: A review. In *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)* (pp. 1-6). IEEE.

[4] Zeb, K., Baig, O., & Asif, M. K. (2015, March). DDoS attacks and countermeasures in cyberspace. In *2015 2nd World Symposium on Web Applications and Networking (WSWAN)* (pp. 1-6). IEEE.

[5] Shen, J., Gui, Z., Ji, S., Shen, J., Tan, H., & Tang, Y. (2018). Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks. *Journal of Network and Computer Applications*, *106*, 117-123.

[6] Lin, W., Xu, S., He, L., & Li, J. (2017). Multi-resource scheduling and power simulation for cloud computing. *Information Sciences*, *397*, 168-186.

[7] Jiang, W., Wang, G., Bhuiyan, M. Z. A., & Wu, J. (2016). Understanding graph-based trust evaluation in online social networks: Methodologies and challenges. *ACM Computing Surveys (CSUR)*, *49*(1), 1-35.

[8] Peng, T., Liu, Q., Meng, D., & Wang, G. (2017). Collaborative trajectory privacy preserving scheme in location-based services. *Information Sciences*, *387*, 165-179.

[9] Nunes, B. A. A., Mendonca, M., Nguyen, X. N., Obraczka, K., & Turletti, T. (2014). A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications surveys & tutorials*, *16*(3), 1617-1634.

[10] Scott-Hayward, S., O'Callaghan, G., & Sezer, S. (2013, November). SDN security: A survey. In *2013 IEEE SDN For Future Networks and Services (SDN4FNS)* (pp. 1-7). IEEE.

[11] Meng, R., Rice, S. G., Wang, J., & Sun, X. (2018). A fusion steganographic algorithm based on faster R-CNN. *Computers, Materials & Continua*, *55*(1), 1-16.

[12] Cui, Q., Zhou, Z., Yuan, C., Sun, X., & Wu, Q. J. (2018). Fast American Sign Language Image Recognition Using CNNs with Fine-tuning. *Journal of Internet Technology*, *19*(7), 2207-2214.

[13] Li, Y., Wang, G., Nie, L., Wang, Q., & Tan, W. (2018). Distance metric optimization driven convolutional neural network for age invariant face recognition. *Pattern Recognition*, *75*, 51-62.

[14] Saied, A., Overill, R. E., & Radzik, T. (2014, June). Artificial Neural Networks in the detection of known and unknown DDoS attacks: Proof-of-Concept. In *International Conference on Practical Applications of Agents and Multi-Agent Systems* (pp. 309-320). Springer, Cham.

[15] Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2015). An empirical evaluation of information metrics for low-rate and high-rate DDoS attack detection. *Pattern Recognition Letters*, *51*, 1-7.

[16] Tan, Z., Jamdagni, A., He, X., Nanda, P., & Liu, R. P. (2013). A system for denial-of-service attack detection based on multivariate correlation analysis. *IEEE transactions on parallel and distributed systems*, *25*(2), 447-456.

[17] Yu, S., Zhou, W., Jia, W., Guo, S., Xiang, Y., & Tang, F. (2011). Discriminating DDoS attacks from flash crowds using flow correlation coefficient. *IEEE*

transactions on parallel and distributed systems, *23*(6), 1073-1080.

[18] Wang, A., Chang, W., Chen, S., & Mohaisen, A. (2018). Delving into internet DDoS attacks by botnets: characterization and analysis. *IEEE/ACM Transactions on Networking*, *26*(6), 2843-2855.

[19] Kumar, G. D., Rao, C. G., Singh, M. K., & Ahmad, F. (2014, June). Using jpcap api to monitor, analyze, and report network traffic for ddos attacks. In *2014 14th International Conference on Computational Science and Its Applications* (pp. 35-39). IEEE.

[20] Johnson Singh, K., Thongam, K., & De, T. (2016). Entropy-based application layer DDoS attack detection using artificial neural networks. *Entropy*, *18*(10), 350.

[21] Rukavitsyn, A., Borisenko, K., & Shorov, A. (2017, February). Self-learning method for DDoS detection model in cloud computing. In *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)* (pp. 544-547). IEEE.

[22] Zhang, H., Cai, Z., Liu, Q., Xiao, Q., Li, Y., & Cheang, C. F. (2018). A survey on security-aware measurement in SDN. *Security and Communication Networks*, *2018*.

[23] Ashraf, J., & Latif, S. (2014, November). Handling intrusion and DDoS attacks in Software Defined Networks using machine learning techniques. In *2014 National Software Engineering Conference* (pp. 55-60). IEEE.

[24] Mihai-Gabriel, I., & Victor-Valeriu, P. (2014, November). Achieving DDoS resiliency in a software defined network by intelligent risk assessment based on neural networks and danger theory. In *2014 IEEE 15th International Symposium on Computational Intelligence and Informatics (CINTI)* (pp. 319-324). IEEE.

[25] Yan, Q., Gong, Q., & Yu, F. R. (2017). Effective software-defined networking controller scheduling method to mitigate DDoS attacks. *Electronics Letters*, *53*(7), 469-471.

[26] Chin, T., Mountrouidou, X., Li, X., & Xiong, K. (2015, June). Selective packet inspection to detect DoS flooding using software defined networking (SDN). In *2015 IEEE 35th international conference on distributed computing systems workshops* (pp. 95-99). IEEE.

[27] Dayal, N., & Srivastava, S. (2017, January). Analyzing behavior of DDoS attacks to identify DDoS detection features in SDN. In *2017 9th International Conference on Communication Systems and Networks (COMSNETS)* (pp. 274-281). IEEE.

[28] Ye, J., Cheng, X., Zhu, J., Feng, L., & Song, L. (2018). A DDoS attack detection method based on SVM in software defined network. *Security and Communication Networks*, *2018*.

[29] Kim, D. E., & Gofman, M. (2018, January). Comparison of shallow and deep neural networks for network intrusion detection. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 204-208). IEEE.

[30] Venkatesh, B., & Anuradha, J. (2019). A review of feature selection and its methods. *Cybernetics and Information Technologies*, *19*(1), 3-26.

[31] Stamp, M. (2017). *Introduction to machine learning with applications in information security*. CRC Press.