Intelligent & Predictive Security Deployment in IOT Environments

Abdul ghani ansari, Dr. Irfana Memon, Dr. Fayyaz Ahmed, Dr. Majid Hussain Memon, Dr. Kelash Kanwar, Dr. fareed Jokhio

ag.ansari19@gmail.com, irfanahameed@quest.edu.pk, Engr_Fayaz@quest.edu.pk, majidhussain@quest.edu.pk, kelashkanwar@quest.edu.pk, fajokhio@quest.edu.pk

Associate Professor, QUEST, Nawabshah, Pakistan

Abstract: The Internet of Things (IoT) has become more and more widespread in recent years, thus attackers are placing greater emphasis on IoT environments. The IoT connects a large number of smart devices via wired and wireless networks that incorporate sensors or actuators in order to produce and share meaningful information. Attackers employed IoT devices as bots to assault the target server; however, because of their resource limitations, these devices are easily infected with IoT malware. The Distributed Denial of Service (DDoS) is one of the many security problems that might arise in an IoT context. DDOS attempt involves flooding a target server with irrelevant requests in an effort to disrupt it fully or partially. This worst practice blocks the legitimate user requests from being processed. We explored an intelligent intrusion detection system (IIDS) using a particular sort of machine learning, such as Artificial Neural Networks, (ANN) in order to handle and mitigate this type of cyber-attacks. In this research paper Feed-Forward Neural Network (FNN) is tested for detecting the DDOS attacks using a modified version of the KDD Cup 99 dataset. The aim of this paper is to determine the performance of the most effective and efficient Back-propagation algorithms among several algorithms and check the potential capability of ANN- based network model as a classifier to counteract the cyber-attacks in IoT environments. We have found that except Gradient Descent with Momentum Algorithm, the success rate obtained by the other three optimized and effective Back- Propagation algorithms is above 99.00%. The experimental findings showed that the accuracy rate of the proposed method using ANN is satisfactory.

Keywords:

Distributed Denial of Service (DDoS) Attacks, Knowledge-Discovery-Dataset(KDD), Artificial Neural Network (ANN),Traincgb,Trainoss,Trainrp,Traingdb

1. Introduction

The IoT has grown-up speedily in recent years, making our everyday lives more convenient. The IoT is a network where lots of network devices are linked to exchange data. Smart devices, such as sensors and actuators, are network device connections. Smart devices are used in a range of applications, including smart buildings, health, cities, grids, transportation, energy storage, and waste management [1]. It has the ability to attach a variety of items, including automobiles, home appliances, wearable, and electronic devices. The connected devices will identify, control, and track the location of the connected devices [2]. Cyber-attacks are more concentrated on them because of the limited resources available to these machines. Since these devices have limited computational processing memory, implementing an efficient protection mechanism is difficult. The hijacked devices conscripted into botnet attacks are one of the most popular IoT challenge attacks [3]. These botnet attacks will use infected IoT devices operated by the C&C server to launch DDoS attacks against the target host. As a result, an effective mechanism to detect this type of attack is needed to secure IoT devices and networks. One of the systems used to detect cyber threats is IDS (intrusion detection system). Recent years have seen a sharp increase in DDOS attacks, which have disrupted numerous IoT networks and caused significant losses. The communal detection systems, like Snort [4] and Suricata [5], are misused-based detection systems. These systems were mostly reliant on the conventional network, despite their prominence for identifying cyber-attacks. Additionally, because misused-based detection systems rely on the signatures of prior assaults to impose the type of intrusion detection system,

Manuscript received December 5, 2022
Manuscript revised December 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.12.24

attackers will get around them. For attack detection, the anomaly-based device used benign traffic data to align with the incoming traffic pattern. These systems are capable of detecting unknown threats, but their implementation in the IoT setting is difficult due to the diverse nature of IoT devices. We implemented the attack detection method using an ANN special form of machine learning since it can detect variants of attack signatures. We used a modified version of the KDDcup99 training dataset to create machine learning-based detection system in this research. In the IoT environments, three-layer architecture is specified [6] i.e, perception layer, network layer and applications layer is illustrate in Figure 1.

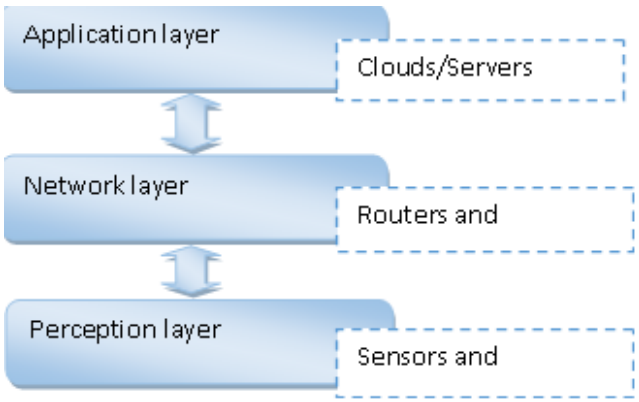

Figure 1: Architecture of IoT layers

The perception layer uses sensing devices such as smart controllers, sensors, and radio-frequency identification (RFID) readers to collect real-time data. This layer is information-free and can only access actual data sets. Bluetooth, wifi, and ZigBee are examples of wireless technologies that can be used to communicate data collected by the perception and device levels. The network layer is in charge of these interactions. The application layer is responsible for processing the provided data in order to achieve the desired result. Due to the heterogeneity of connected objects, protection is a major challenge in the IoT setting. It might be challenging to protect IoT devices from many types of threats. Since the IoT comprises various layers, each of which has weaknesses and restrictions. Because of this, the IoT environment must identify security by guarding against many sorts of assaults. The current protection framework needed to be balanced with a variety of functions. In the IoT environment, a DDoS attack is a common security problem.

1.1 DDOS (DISTRIBUTED DENIAL OF SERVICE)

Securing IoT against a wide range of threats is a difficult challenge. A DDoS attack involves an effort to completely or partially destroy a targeted server by flooding it with internet traffic. The key objective of a DDoS attack is to disrupt the victim's network or server device [7]. Economic losses, downtime, and short and long-term consequences were all caused by DDoS attacks on the victim server. Since the attacker generates a defensive force in the form of zombies or bots, a DDoS attack is more effective. All of these bots have been programmed to target the victim and disable their features. Web cameras, CCTV, and other smart devices are among the non-legacy IoT devices targeted by the intruder Detecting DDoS attacks in the IoT can thus avoid floods from unknown attackers and improve the efficiency of linked resources [8]. Several research projects relating to IoT have recently attracted increased interest from academics, researchers, and industry. The purpose of this work is to address some machine learning methods or techniques for detecting DDoS attacks in an IoT environment.

1.2 IOT BOTNETS' CONCEPT IN DDOS ATTACK

IoT, which used to be considered favorable for technology, is now evolving into a nightmare by fostering significant DDoS attacks. The probability of falling victim to a DDoS assault rises as the number of IoT devices increases. The method used in these attacks is to infect various IoT devices with malware, which is subsequently sent through the network. The number of compromised IoT devices that the attacker employs to launch the attack increases the frequency of this type of attack. These compromised IoT devices are referred regarded as "bots" in the hacking community [9]. When malicious software (malware) is placed on the victim device, it overcomes its security measures without the user's knowledge, turning the victim IoT device into a bot. In addition, the server in charge of these bots is referred to as the "Master Bot Controller." The master bots can communicate with the infected systems over HTTP, HTTPS, or IRC (Internet Relay Chat). While botnets based on HTTP employ the HTTP protocol, which operates at the bit level of sent data and is therefore more difficult to monitor and identify, botnets based on IRC have a client-server architecture with default communication channels. A "Botnet" is a network made up of millions of bots that are organised under the Master Bot as the infection is repeated. On the other hand, a traditional botnet network comprises a master bot controller that can be used to spawn new bots and differs slightly in reach and scope from current IoT botnets (malware).

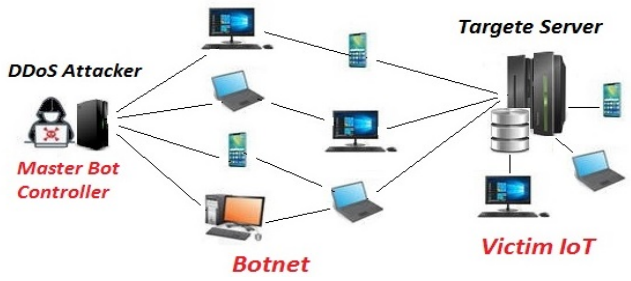

Figure 2: DDOS Attack in IoT

Traditional botnets can only compromise a small number of computer systems. IoT botnets are significantly more sophisticated in terms of targeting a larger number of IoT devices because IoT devices are often on and connected to the internet for much longer periods of time (nearly 24/7/365). Once the botnet has been established, as shown in Figure 2, all of the bots are treated as slaves of the attacker's master bot controller, and each bot is given instructions to send bogus packets to the targeted web server while also preventing more legitimate packets from reaching the targeted server system. As a result, botnets are the primary reason why such attacks have become so common [10]. How IoT devices can fall into such traps so swiftly is now the subject in discussion. The explanation for this is that we have been negligent in securing simple IoT products. We continue to be concerned about the security of more expensive, more important electronics in our daily lives, such as the lock systems in our vaults, vehicles, and other valuable items [11]. When it comes to protecting these products, we sometimes overlook the small and significant devices that are less expensive and don't draw our attention to ensuring their basic level of protection, such as web cameras, smart TVs, and music systems. Our lack of understanding of how to secure these devices is enough to draw the attacker's attention. The majority of these devices either has no protective mechanism at all or has a poor one, which allows the attacker's malware to easily compromise them by brute-forcing all potential username and password combinations. In the corporate world of today, this is a question of business. The bot masters benefit from the sale of their attack services [12]. New botnet versions have been developed in the digital market as a result of such competition among various IT and commercial rivalries.

1.3 DDOS ATTACK DETECTION IN IOT

In an IoT system when resources are shared, DDoS is a challenging security issue that interrupts traffic. Therefore, identifying DDoS is a crucial task in order to give end users more efficient resource sharing. There have been reports of a number of widespread DDoS assaults in the IoT context, including HTTP GET/POST attacks and attacks using the TCP, ICMP, UDP, and GET/POST commands of the TCP/TCP protocols. Taxonomy of DDoS attacks shown in Figure 3.

Anomaly-based and signature-based detection [13, 14] techniques can be separated into two groups. The signature-based detection method compares the network traffic it has captured to known attack patterns, like packet sequences or bytes. This type of detection strategy is easier to comprehend, extend, and produce more significant findings than anomaly-based detection schemes. On the other hand, the signature-based detection system can only distinguish between known attacks with a pre-defined pattern.

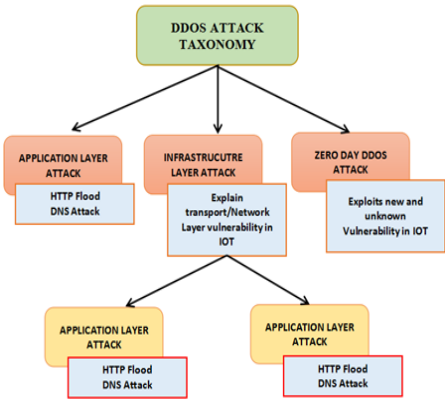

Figure 3: Taxonomy Of DDoS Attack In IoT Environment

Utilizing patterns of behavior, the anomaly-based detection approach is employed to identify the attack. The unknown attack will be classified by this type of detection model. But it doesn't have poor precision. A variety of real-time issues can be solved using machine learning (ML) methods. Machine learning algorithms' primary objectives are to identify patterns in given data samples and create predictions in order to address business issues. Machine learning algorithms use a range of methods [13-19]. A subfield of artificial intelligence (AI) known as machine learning allows computer programmers to learn from examples, analogies, and prior experiences [13]. The capabilities included into the software get smarter as it learns, and the programmer is able to make better decisions. Two of the most popular methods in machine learning are genetic algorithms and Artificial Neural Networks (ANN).

1.4 ARTIFICIAL NEURAL NETWORKS (ANN)

ANNs are artificial neural networks that imitate the brain's neurons and synapses to transfer data for communication, learning, and decision-making [20]. Inside IoT systems, ANNs are used to track the status of

IoT devices and make informed decisions [21]. They have only recently become a significant component of artificial intelligence. There are various forms of Artificial Neural Networks used in machine learning today, but this study used a feed-forward neural network to detect DDoS attacks. The three basic layers of an artificial neural network are in Figure 4.

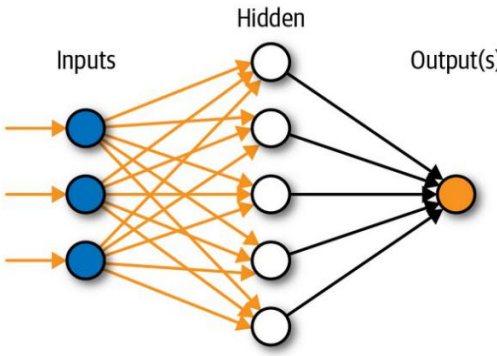

Figure 4: Artificial Neural Network

- **Input layer in a neural network:** The artificial input neurons that make up the input layer of the neural network send data from the initial neuron layers to the plan for additional processing. The workflow is launched by the neural network's input layer.

- **Hidden layer in a neural network:** The hidden layer of an artificial neural network is composed of a number of layers between the input and output layers, where the input and output of the artificial neurons are weighted according to the number of inputs.

- **Output layer in a neural network:** An artificial neural network's output layer is the last layer of neurons that produces certain programme outputs. Neurons at the output layer can be formed and handled differently than other artificial neurons in the neural network since they are the final "performer" nodes in the network [22].

The purpose of this study is to determine whether it is feasible to accurately identify DDoS traffic on a changing collection of data utilising derived parameters from collected traffic and artificial neural networks (ANN). In order to increase the accuracy of identifying particular types of DDoS traffic, the goal of this study effort is to design an IDS based on ANN and assess and check the effectiveness of a variety of machine learning algorithms for detecting and classifying DDoS traffic. The telecommunication industry, as well as the domains of traffic and transportation technology, have benefited from the growing use of ANN as an expert systems tool in numerous areas and fields. Artificial intelligence (AI) has been extensively studied in recent years with the goal of identifying and categorising unwanted DDoS traffic. In order to lessen the detrimental impacts of DDoS attacks in various information and communication contexts, numerous approaches were researched, put forth, and tested. The study states how to build an ANN model that can quickly identify both known and unidentified DDoS attacks. In order to distinguish between DDoS assaults and normal traffic samples, the attacks were located by extracting pertinent characteristics (such as source and destination IP addresses, packet length, destination port, and a sequential number of packets). The created ANN model was trained using parameter values. The proposed model was used to identify attacks based on the TCP, UDP, and ICMP protocols. The inability to distinguish the exact form of DDoS attack is the reason for this research's deficiency. DDoS attacks were detected with 99 percent accuracy using the assessment model. We proposed IDS that uses ANN to learn a system's and connected devices' healthy states.

1.5 FEED-FORWARD NEURAL NETWORK (FFNN)

A FFNN is a type of ANN in which connections between nodes do not form a loop. A FFNN, in which some paths are cycled, is the polar opposite of a recurrent neural network (RNN). The feed-forward model is the simplest kind of neural network because input is only processed in one direction. The data never travels backwards, regardless of how many hidden nodes it passes through [23]. FFNN is shown in Figure 5.

1.6 RUNNING OF FEED-FORWARD NEURAL NETWORKS

A single layer perceptron is a Feed- Forward Neural Network in its most basic form. Consider the ANN model mentioned above, The layer receives a number of inputs and multiplies them by the weights. The sum of the weighted input values is then calculated. The output value is typically 1 if the sum of the values exceeds a predefined threshold, which is typically set at zero, and typically -1 if the sum is less than the threshold. A popular feed-forward neural network model for classification is the single-layer perceptron. Additionally, single –layer perceptoron can use machine learning functions. The delta rule is a characteristic of the neural network that allows it to compare the outputs of its nodes with the intended values and adjust its weights over training to create output values that are more accurate. There is a gradient descent as a result of this teaching and learning process. Although the procedure for updating weights in multi-layered perceptrons is essentially the same, it is known as back-

propagation. In such cases, each hidden layer in the network is modified using the output values given by the final layer. Feed-forward neural networks, which have a simple design as shown in Figure 5, can be useful in a variety of machine learning applications. To operate several feed forward neural networks separately but with a light intermediary for moderation, for instance, may be the setup. This process, like the human brain, depends on a huge number of individual neurons to organize and process larger tasks. Since each network completes its task independently, the final outputs can be combined to produce a synthesized and coherent output.

The following is how the rest of the paper is organized: in section II, will discuss about modified data-sets. In section III, the history approach, from which we adapted the methods and techniques, will be presented. In section IV, the proposed method will be defined, and in section V, the evaluation results will be discussed. Finally, segment will bring this paper to a close.

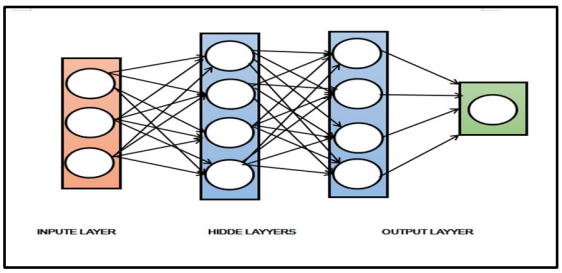

Figure 5: Feed Forward Neural Network

1.7 DATA SETS DESCRIPTION

Here we'll look at some attacks that can cause slower network performance, device crashes, or device failures for legitimate users. For instruction detection, a modified version of the KDD cup 1999 is used. Both the training and test datasets include 41 features that are classified as common traffic or special attack forms. KDD data labels or classes are further divided into two groups, each representing an attack or no attack. The following four attack groups are included in the KDD Cup 99 updated dataset as shown in Table 1.

1) **DOS (Denial of Service),** to flood a specific system or resource, the intruder would need a laptop, smart devices, and an internet connection. An attacker tries to keep legitimate customers away from a service e.g. TCP, SYN, Flood, Smurf, Neptune, back, Pod, tear drop [24].
2) **Probe,** An attacker tries to figure out what data is on the target host. For example, by scanning victims for knowledge of services available attack types upsweep, nmap, portsweep, satan using the Operating System [25].

3) **R2L (Remote to Local),** an intruder attempts to gain access to a local host account even though he does not have one [26].
4) **U2R (User to Root),** an intruder uses a local host account to try to gain root privileges. Overflowing buffer [27].

Table1: Data set description

Name of the files	Features Description
KDD_DDoS.csv	duration, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, num_compromised, root_shell, su_attempted, num_root, num_file_creations, num_shells, num_access_files, num_outbound_cmds, is_host_login, is_guest_login,count, srv_count ,serror_rate, srv_serror_rate, rerror_rate, srv_rerror_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, dst_host_srv_rerror_rate, protocol_type,service,flag
KDD_Probe.csv	
KDD_R2L.csv	
KDD_U2R.csv	

Using deep learning and machine learning techniques, a number of researchers are focusing on detecting the most common DDoS attacks that have the greatest effect on the internet and IoT. The following are some examples of recent work in this area:

2. RELATED WORK

Pande, S et al. (2021) [28] discussed numerous ongoing DDoS attack detection techniques with an emphasis on machine learning techniques. There is also a discussion of a list of freely available DDoS resources. The DDoS attack was carried out using a command-based ping of death technique. The model was trained using the random forest algorithm, which resulted in 99.76 percent of instances being correctly classified.

Y. Jia et al. (2020) [29] designed a new ML model to detect assaults in an IoT environment.The creation of new DDoS defence strategies for an IoT setting, together with FlowGuard, an edge-focused IoT defence system for DDoS detection, categorization, and mitigation. Two machine learning algorithms for DDoS detection and classification are developed, together with a novel DDoS assault detection model based on traffic changes. Two DDoS simulators, Slow HTTP Test and BoBeSi, produce

two sizable datasets to show the effectiveness of the two machine learning algorithms.

M. Gajewski et al. (2020) [30] developed a new technique for anomaly detection of network traffic attacks in the IoT environment, which provides shared accountability between a network provider and a service client. In order to identify the monitoring data, the established scheme employs a machine learning technique.

Dong et al. (2020) [31] developed a brand-new extreme learning machine-based bot-net identification method .The suggested detection method may quickly learn without data processing by simply obtaining network stream files and removing botnet traffic characteristics.

N. A. de Souza et al. (2020) [32] investigated k-Nearest Neighbor (kNN) and Deep Neural Networks (DNN), a two-stage hybrid binary classification technique, that combined DNN and kNN algorithm.

N. F. Syed et al. (2020) [33] created a new-fangled framework for detecting application layer DoS attacks that is designed to interact with the MQTT (Message Queuing Telemetry Transport) protocol. The MQTT information busted is protected by using the ML algorithm to detect the attack.

Cvitić et al. (2019) [34] developed an innovative detection method for DDoS attacks generated by IoT devices. The conceptual framework that is presented is built on system groups, where each apparatus relies on the distinctiveness of its own traffic.

Alsamiri, J et al. (2019) [35] proposed to evaluate several machine learning methods in order to quickly and effectively detect IoT network attacks.A recent dataset called Bot-IoT is being used to test several detection methods. The implementation procedure involved the employment of seven different machine learning algorithms, and the bulk of them worked successfully. Additional features were taken from the Bot-IoT dataset during installation and compared to research from the literature; the new features produced superior results.

Roopak, M., et al. (2019) [36] suggested the deep learning models for DDoS attack detection and they were tested on the most recent CICIDS2017 datasets and obtained the maximum accuracy of 97.16 percent. Proposed models and machine learning algorithms are frequently contrasted. All other deep learning models and machine learning algorithms are beaten by the CNN+LSTM hybrid model in performance.

Ge, M., et al. (2019) [37] presented a novel traffic flow classification approach to intrusion detection for IoT. The author developed generic characteristics from field data at the packet stage using a recently made available IoT dataset. For binary and multi-class classification attacks against IoT devices, such as denial of service, distributed denial of service, reconnaissance, and data theft, the author developed a feed-forward neural network model. The evaluation of the suggested scheme using the processed dataset shows that it has a good level of classification accuracy.

Bhuvaneswari Amma N. G. et al. (2019) [38] applied a technique, deep intelligence. The intelligence was derived using a radial base function with different levels of abstraction. The experiment was conducted on the recognized NSL KDD and UNSW NB15 datasets, with 27 features taken into account. In comparison to other existing methods, the author believed that his method was more accurate.

Muhammad Aamir et al (2019) [39] used a clustering approach to introduce a feature selection process. Five separate machine learning algorithms were used to compare algorithms. For training, random forest (RF) and support vector machine (SVM) was used. The highest level of accuracy reached by RF was about 96 percent.

Narasimha et al. (2019) [40] employed anomaly detection besides the ML algorithms intended to detect the normal and attacked traffics. Real-time datasets were used in the experiment. For classification, the well-known naive Bayes ML algorithm was used. The results were contrasted with other algorithms like J48 and random forest (RF).

Doshi, R., et al. (2018) [41] demonstrated that normal and DoS attack traffic from consumer IoT devices can be successfully distinguished by machine learning-based packet-level DoS detection. For real-time classification and middlebox deployment, the author used a minimal feature set to reduce computational overhead. The attributes were chosen under the presumption that user IoT device network traffic patterns differ from well-researched non-IoT networked devices. The author examined five different ML classifiers on a dataset of regular and DoS attack traffic that was obtained from an experimental consumer IoT system network. All five algorithms' test set accuracy values above 0.99. Further research into machine learning anomaly detection is required in light of these preliminary findings in order to safeguard networks from weak IoT devices.

Doshi et al. (2018) [42] was developed a new model for detection which is based on Artificial Neural Network (ANN). A new router for home gateways or middleboxes has been built that can detect local IoT device sources of DDoS attacks.

Y. Meidan et al. (2018) [43developed the most recent N-BaIoT network-based anomaly detection techniques designed for the IoT context. The built-in technique uses deep autoencoders to spot unusual network activity emanating from compromised IoT devices.

Abeshu et al. (2018) [44] suggested a deep learning-based technique for identifying distributed assaults in fog-to-thing computing. Because cloud computing provides centralised processing, which is useless for big IoT networks that require cyber-security processing at the network's edge, this work exemplifies the drawbacks of cloud computing in IoT networks. An extensive IoT network that produces a lot of data can benefit from a fog-to-node strategy because deep learning has been demonstrated in the realm of big data. On NSLKDD datasets, this study compares a stacked autoencoder with softmax as classifier to a shallow learning model based on performance criteria like accuracy, false alarm rate, and detection rate. The author asserts that using distributed parallel computing to the fog to node model increased the accuracy and effectiveness of attack detection.

A. V. K. Rahul et al. (2018) [45] suggested a deep neural network and contrasted it to a shallow one. The proposed system, which has a learning rate of 0.1, was trained and tested using the KDDCup-'99 dataset. Using performance parameters including accuracy, precision, and recall, the results were contrasted with those of other machine learning techniques. According to the author, three-layer deep neural network models outperformed other models in the research, and deep learning is a promising technology for cybersecurity.

B. Nathan Shone et al. (2018) [46] used a DL algorithm for the identification of the attack. It also used the non-symmetric deep autoencoder (NDAE) function of unsupervised learning. On the well-known KDD-Cup 99 and NSL-KDD datasets. TensorFlow was used to implement the proposed technique on a graphics processing unit (GPU). The author believed that he was able to get more precise detection results.

Olivier Brunetal. (2018). [47] The author employed one of the most well-known deep learning approaches, the random neural network (RNN) method, which is operated in the field of IoT for detecting a DDoS attack for network detection. This deep learning-based technology effectively generates more promising findings as compared to conventional methods.

Ahanger, T. A. (2017) [48] proposed a DDoS detection system based on the LVQNN principles. The most recent dataset is used to train the neural network, and the results of the proposed system are compared to those of the BPNN to demonstrate that the former is more effective in detecting DDoS attacks, with 99.8% detection accuracy as opposed to 89.8% detection accuracy. The mechanism also showed that the performance of DDoS attack detection may be enhanced by adopting ANNIDS based on host anomaly detection.

Aljumah, A et al. (2016) [49] proposed an unique technique for successfully identifying DDoS attacks using artificial neural networks and chaos theory. The author started the learning process by simulating a real network environment. Various DDoS assaults were launched while this study's network was being used by genuine traffic. Using supervised and unsupervised approaches, the author separated DDoS attacks from legitimate traffic. To differentiate between legitimate traffic and DDoS attacks, the author used the Lyapunav coefficient. The author used updated datasets and trained artificial neural networks with these two learning methods to detect DDoS attacks with greater than 95% accuracy.

3. The Proposed Architecture's Design

The proposed approach employs an Artificial Neural Network (ANN) classifier. The system's performance is assessed using a modified KDD Cup99. The picture depicts the system's step by step flow, which comprises steps like KDD data collecting, data cleaning, neural network use, and neural network training shown in Figure 6.

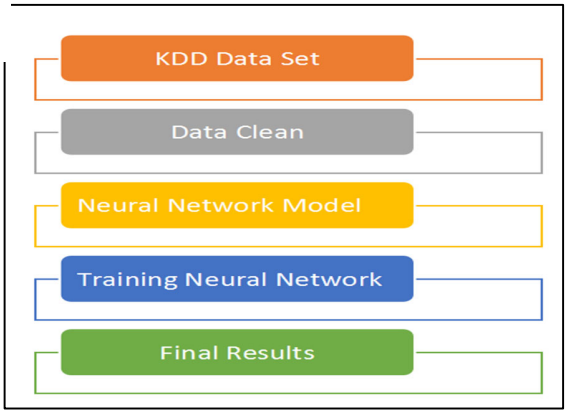

Figure 6: Basic Flow of research designing ANN Model

A. Knowledge Discovery in Databases (KDD) Data Set:

In additional steps, the first KDD data set from a trusted source is obtained for processing. The Knowledge Discovery of Databases (KDD) 1999 data collection, which was launched for the KDD Cup competition 1999, was developed using a variety of pattern recognition and machine learning algorithms. In an IoT network, the KDD-Cup 1999 dataset links records of attacks and intrusion. The framework employs KDD data for dataset testing and training.

B. Data Preparation

Clean up the KDD data set and put a value on protocol, attacks, and flags.

C. Model of a Neural Network

Following the set of desired data, we must construct the proposed network model using the data collected.

D. Neural Network Training

Using a feed-forward network, train the network model to compare and analyze the performance of several algorithms. Which result is the best for detecting a DDoS attacks?

E. Result

After the network has been trained, we can decide which algorithm is the most effective at detecting attacks.

4. EXPERIMENTAL RESULTS

For this research work, we used the MATLAB simulator, specifically ANN toolbox. First, we clean the KDD data set and assign Protocol, Attacks, and Flags values. Then we build a neural network model and use it to train the modified KDD data set. We received the results of DDoS attack detection after the training has been completed. The authors have converted the feature variables **"Protocol type"** with values like tcp=1, SMTP=2, http=3, UDP=4, Urp_i=5, ftp_data=6, Finger=7, FTP=8, Domain=9, ntp_u=10, **"Flag"** with corresponding values S0=0, RSTR=1, S1=2, REJ=3, S2=4, SO=5, SH=6, RSTO=7, SH=8, S1=9, OTH=10 The **"Attacks"** in the data set have been categorized as normal=0, warezclient=1, back=2, smurf=3, teardrop=4, ipsweep=5, multihop=6, neptune=7, ftp_write=8, nmap=9, warezmaster=10, pod=11, buffer_overflow=12, and so on.

This segment proves all of the experiments, performance, and dialogue throughout the experiment, a feed-forward neural network was used. For the entire experiment following neural network model (Figure 7) was used.

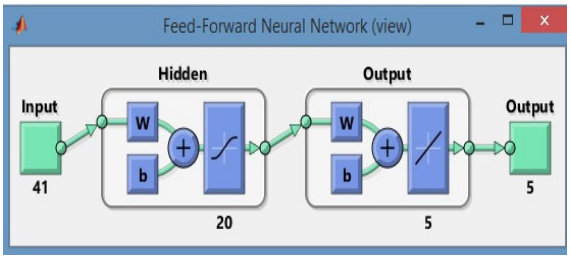

Figure 7: Artificial Neural Networks

We cleaned the data, prepared it ahead of time, and passed it into the DDOS feed-forward system. During the training of the neural network, the total time taken was 52 minutes and 17 seconds. The basic framework needed 243 emphases to plan. Total 10 algorithms trained but in this research work we used "scaled **Conjugate** gradient Back- Propagation with Powell-Beale Restarts", "One-step Secant Back-propagation", Resilient Back-Propagation", and Gradient descent with momentum and "Adaptive learning rate Back-Propagation" for training towards performance evolution. The system proposed used MATLAB (version R2018a 7.14.0.739, 32 bit (Win 32 bits)) to implement the optimized machine learning algorithms. In order to implement these algorithms, the framework is expected to first create the designed neural network using training data and then evaluate the network performance using testing data.

4.1 CONJUGATE GRADIENT BACKPROPAGATION WITH POWELL-BEALE RESTARTS

The path of the gradient to the set must be stated at specified intervals in conjugate gradient algorithms. This approach suggests restarting only when the gradient's orthogonality changes only slightly [50, [51]. The performance of the feed-forward neural network when using this algorithm demonstrates that validation is based on threshold optimization. The performance curve is created during network training, testing, and validation. At 1000 epochs, we got 0.00057684, the Best Validation Performance. Figure 8 depicts the confusion matrix after applying this technique.

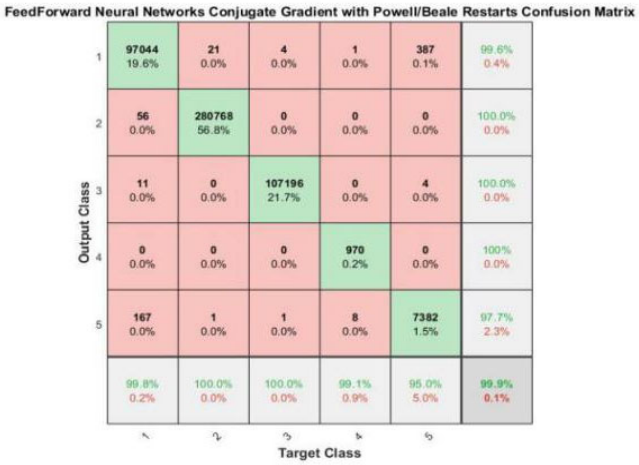

Figure 8: Confusion Matrix

The success rate in comparison to the error rate at every stage, including training, validation, and level 1 testing. The outcome is listed in TABLE 2.

Table 2: Showing the result towards performance evolution.

S.No	Back-propagation algorithm	Success rate /Accuracy	Training time	Mean squared error at epoch 1000
1.	Conjugate Gradient with Powell Beale Restarts	99.9%	21:09	0.00057684
2.	One Step Secant	99.5%	19:49	0.0015587
3.	Resilient Back propagation	99.4%	5:46	0.0023303
4.	Gradient Descent with Momentum	96.2%	5.33	0.021533

4.2 ONE-STEP SECANT BACK-PROPAGATION

It is an extension of the BFGS algorithm that, in order to save space and cut down on computation, assumes that the hessian matrix is an identity matrix and employs a secant approximation. It combines a quasi-Newton algorithm with a conjugate gradient algorithm [52].

$$dX = -gX + a\Delta X_{K-1} + \beta \Delta gX_{K-1}$$

where ΔX_{K-1}= change in weights in a previous iteration, ΔgX_{K-1}= change in gradient in a previous iteration

at 1000 epochs, we have got the best validation performance as is 0.0015587. The Figure 9 shows the confusion matrix when applied to this algorithm. The success rate versus the error rate at every stage, including training, validation, and level 1 testing.
The outcome is listed in TABLE 2.

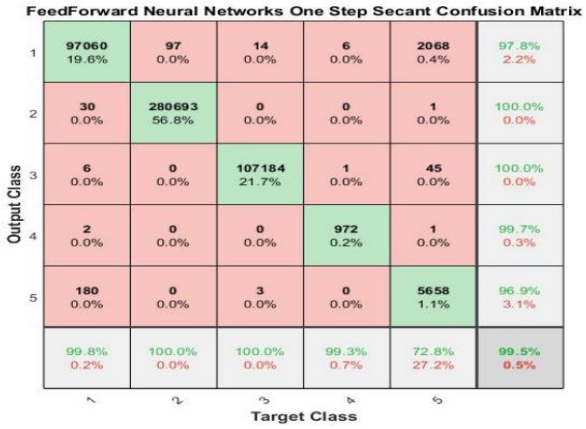

Figure 9: Confusion Matrix of one step secant

4.3. RESILIENT BACKPROPAGATION

Sigmoid transfer functions are used by multilayer neural networks to condense the inputs into a finite output set.Partial derivatives prevent the network from converging because they cause the weights to change only little. By selecting only the direction of weight update via partial derivative, resilient back-propagation eliminates this and uses less memory. [53].

$$dX = deltaX .* sign(gX)$$

where $deltaX$ = derivative of performance, gX= gradient,
dX = change n weight

We got the best validation performance i.e 0.0023303 at 1000 epochs. While the following figure 10 shows the confusion matrix when employed in this algorithmThe success rate vs. mistake rate for each stage, including level 1 testing, validation, and training. The outcome is listed in TABLE 2.

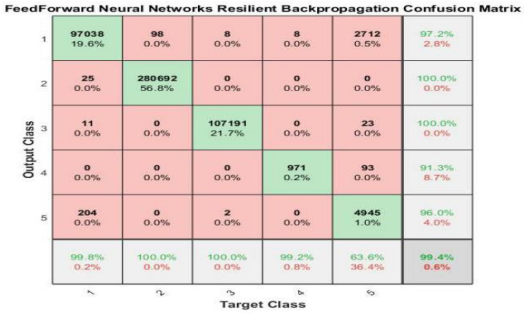

Figure 10: Confusion Matrix of Resilient Back-Propagation

4.4. GRADIENT DESCENT WITH MOMENTUM

This algorithm uses gradient descent with momentum to keep networks from being trapped in shallow local minimums [54].

$$dX = m_c \, \Delta X_{K-1} + lr * m_c * \frac{deltaX}{dX}$$

Where m_c =momentum coefficient, lr =learning rate, $deltaX$ = derivative of performance

We have got the best validation performance after training neural network which is 0.021533 at 1000 epochs. The figure 11 shows the confusion matrix when in used this algorithm. The success rate against error rate for every stage, including training, validation, and level 1 testing. The outcome is listed in TABLE 2.

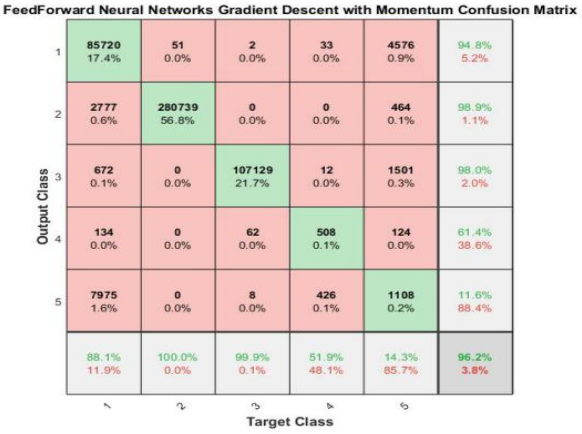

Figure 11: Confusion Matrix

5. CONCLUSION AND FUTURE WORK

IoT devices' security breaches develop a significant challenge because of their resource constraints. The attackers are concentrating their efforts on infecting these devices with a botnet, which is then controlled and managed by a C&C server to launch a DDoS attack against the target source. In this context we considered an intelligent IDS using ANN for DDOS attacks detection in IoT environments. This paper has aimed to evaluate the performance of the most optimized and effective Back-Propagation algorithms and these machine learning algorithms can identify DDOS attacks more effectively, although some machine learning algorithms are constrained by computing complexity. In this work, the tailored version of KDDCUP99 was used as a dataset because of its regular updates. We have found that except Gradient Descent with Momentum algorithm, the success rate obtained by other three optimized and effective Back-Propagation algorithms is above of 99.00%. The experimental findings showed that the accuracy rate of the proposed method using ANN is satisfactory. Our model was strengthened by including more invalid data points and retraining it to recognise both valid and invalid data points. We found that in IoT situations, our technique may be able to identify false data points. Building a sizable testbed and gathering additional data will be the next step, and we are certain that it will allow the system to efficiently detect different forms of assaults in IoT contexts.

References

[1] A. Whitmore, A. Agarwal, and L. Da Xu, "The Internet of Things—A survey of topics and trends," Information Systems Frontiers, vol. 17, no. 2, pp. 261-274, 2015.

[2] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the IoT and industrial IoT: A review," Internet of Things, p. 100081, 2019.

[3] "The 7 Most Common IoT Security Threats in 2019 | IoT For All," 2019. [Online]. Available: https://www.iotforall.com/7-most-common-iot-securitythreats-2019/. [Accessed: 13-Jul-2019].

[4] A. R. Baker and J. Esler, Snort IDS, IPS Toolkit. Syngress Publishing, Inc. Elsevier, Inc. 30 Corporate Dr. Burlington, MA 01803, 2007.

[5] S. A. R. Shah and B. Issac, "Performance comparison of intrusion detection systems and application of machine learning to Snort system," Futur. Gener. Comput. Syst., vol. 80, no. November 2017, pp. 157–170, 2018.

[6]R. Vishwakarma and A. K. Jain, "A survey of DDoS attacking techniques and defence mechanisms in the IoT network," Telecommunication Systems, vol. 73, no. 1, pp. 3-25,

[7] M. M. Salim, S. Rathore, and J. H. Park, "Distributed denial of service attacks and its defenses in IoT: a survey," The Journal of Supercomputing, pp. 1-44, 2019.

[8] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "DDoS attacks in cloud computing: Issues, taxonomy, and future directions," Computer Communications, vol. 107, pp. 30-48, 2017.

[9]Mendez,D.M.,Papapanagiotou,I.,&Yang,B.(2017).Internetof things: Survey on security and privacy. arXiv preprint arXiv:1707. 01879.

[10] McDermott, C. D., Petrovski, A. V., & Majdani, F. (2018, June). Towards situational awareness of botnet activity in the internet of things. In 2018 International conference on cyber situational awareness, data analytics and assessment (Cyber SA) (pp. 1–8). IEEE.

[11] Bertino, E., & Islam, N. (2017). Botnets and internet of things security. Computer, 2, 76–79.

[12] Jerkins, J. A. (2017, January). Motivating a market or regulatory solution to IoT insecurity with the Mirai botnet code. In *2017 IEEE 7th annual computing and communication workshop and conference (CCWC)* (pp. 1-5). IEEE.

[13] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, "Distributed denial of service (DDoS) resilience in cloud: Review and conceptual cloud DDoS mitigation framework," Journal of Network and Computer Applications, vol. 67, pp. 147-165, 2016.

[14] A. Rawashdeh, M. Alkasassbeh, and M. Al-Hawawreh, "An anomaly-based approach for DDoS attack detection in cloud environment," International Journal of Computer Applications in Technology, vol. 57, no. 4, pp. 312-324, 2018.

[15] H. Polat, O. Polat, and A. Cetin, "Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods and Machine Learning Models," Sustainability, vol. 12, no. 3, p. 1035, 2020.

[16] M. Idhammad, K. Afdel, and M. Belouch, "Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest," Security and Communication Networks, vol. 2018, 2018.

[17] A. Saied, R. E. Overill, and T. Radzik, "Detection of known and unknown DDoS attacks using Artificial Neural Networks," Neurocomputing, vol. 172, pp. 385-393, 2016.

[18] A. Amjad, T. Alyas, U. Farooq, and M. Tariq, "Detection and mitigation of DDoS attack in cloud computing using machine learning algorithm," EAI Endorsed Transactions on Scalable Information Systems, vol. 6, no. 23, 2019.

[19] M. Ghanbari and W. Kinsner, "Detecting DDoS Attacks Using Polyscale Analysis and Deep Learning," International Journal of Cognitive Informatics and Natural Intelligence (IJCINI), vol. 14, no. 1, pp. 17-34, 2020.

[20] Michael Negnevitsky. Artificial Intelligence: A Guide to Intelligent Systems. Pearson, 2011.

[21] I. Kotenko, I. Saenko, F. Skorik, and S. Bushuev. Neural network approach to forecast the state of the internet of things elements. In Soft Computing and Measurements (SCM), 2015 XVIII International Conference on, pages 133–135, May 2015.

[22] Dongare, A. D., Kharde, R. R., & Kachare, A. D. (2012). Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology (IJEIT)*, *2*(1), 189-194.

[23] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. E. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," Heliyon, vol. 4, no. 11. Elsevier Ltd, p. e00938, 2018.

[24] M. S. Galina Mikhaylova, "The 'Anonymous' Movement: Hacktivism as an Emerging Form of Political Participation," Graduate Council of Texas State University, 2014

[25] S. Paliwal and R. Gupta, "Denial-of-Service, Probing & Remote to User (R2L) Attack Detection using Genetic Algorithm," Int. J. Comput. Appl., vol. 60, no. 19, pp. 975–8887, 2012.

[26] M. Sabhnani and G. Serpen, "KDD feature set complaint heuristic rules for R2L attack detection," Proc. Int. Conf. Secur. Manag., vol. 1, pp. 310–316, 2003.

[27] F. Mozneb and A. Farzan, "The Use of Intelligent Algorithms to Detect Attacks In," vol. 3, no. 9, pp. 579– 584, 2014.

[28] Pande, S., Khamparia, A., Gupta, D., & Thanh, D. N. (2021). DDOS Detection Using Machine Learning Technique. In *Recent Studies on Computational Intelligence* (pp. 59-68). Springer, Singapore.

[29] Jia, Y., Zhong, F., Alrawais, A., Gong, B., & Cheng, X. (2020). Flowguard: an intelligent edge defense mechanism against IoT DDoS attacks. *IEEE Internet of Things Journal*, *7*(10), 9552-9562.

[30] Gajewski, M., Mongay Batalla, J., Mastorakis, G., & Mavromoustakis, C. X. (2020). Anomaly traffic detection and correlation in Smart Home automation IoT systems. *Transactions on Emerging Telecommunications Technologies*, e4053.

[31] Dong, X., Dong, C., Chen, Z., Cheng, Y., & Chen, B. (2020). BotDetector: An extreme learning machine-based Internet of Things botnet detection model. *Transactions on Emerging Telecommunications Technologies*, e3999.

[32] de Souza, C. A., Westphall, C. B., Machado, R. B., Sobral, J. B. M., & dos Santos Vieira, G. (2020). Hybrid approach to intrusion detection in fog-based IoT environments. *Computer Networks*, *180*, 107417.

[33] Syed, N. F., Baig, Z., Ibrahim, A., & Valli, C. (2020). Denial of service attack detection through machine learning for the IoT. *Journal of Information and Telecommunication*, *4*(4), 482-503.

[34] Cvitić, I., Peraković, D., Periša, M., & Botica, M. (2019). Novel approach for detection of IoT generated DDoS traffic. *Wireless Networks*, 1-14.

[35] Alsamiri, J., & Alsubhi, K. (2019). Internet of Things cyber attacks detection using machine learning. *Int. J. Adv. Comput. Sci. Appl*, *10*(12).

[36] Roopak, M., Tian, G. Y., & Chambers, J. (2019, January). Deep learning models for cyber security in IoT networks. In *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)* (pp. 0452-0457). IEEE.

[37] Ge, M., Fu, X., Syed, N., Baig, Z., Teo, G., & Robles-Kelly, A. (2019, December). Deep learning-based intrusion detection for iot networks. In *2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC)* (pp. 256-25609). IEEE.

[38] NG, B. A., & Selvakumar, S. (2019). Deep radial intelligence with cumulative incarnation approach for detecting denial of service attacks. *Neurocomputing*, *340*, 294-308.

[39] Aamir, M., & Zaidi, S. M. A. (2019). Clustering based semi-supervised machine learning for DDoS attack classification. *Journal of King Saud University-Computer and Information Sciences*.

[40] Mallikarjunan, K. N., Bhuvaneshwaran, A., Sundarakantham, K., & Shalinie, S. M. (2019). DDAM: Detecting DDoS attacks using machine learning approach. In *Computational Intelligence: Theories, Applications and Future Directions-Volume I* (pp. 261-273). Springer, Singapore.

[41] Doshi, R., Apthorpe, N., & Feamster, N. (2018, May). Machine learning ddos detection for consumer internet of things devices. In *2018 IEEE Security and Privacy Workshops (SPW)* (pp. 29-35). IEEE.

[42] Doshi, R., Apthorpe, N., & Feamster, N. (2018, May). Machine learning ddos detection for consumer internet of

things devices. In *2018 IEEE Security and Privacy Workshops (SPW)* (pp. 29-35). IEEE.
[43] Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., & Elovici, Y. (2018). N-baiot—network-based detection of iot botnet attacks using deep autoencoders. *IEEE Pervasive Computing*, *17*(3), 12-22.
[44] Abeshu, A., & Chilamkurti, N. (2018). Deep learning: The frontier for distributed attack detection in fog-to-things computing. *IEEE Communications Magazine*, *56*(2), 169-175.
[45] Vigneswaran, R. K., Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2018, July). Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security. In *2018 9th International conference on computing, communication and networking technologies (ICCCNT)* (pp. 1-6). IEEE.
[46] Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE transactions on emerging topics in computational intelligence*, *2*(1), 41-50.
[47] Brun, O., Yin, Y., Gelenbe, E., Kadioglu, Y. M., Augusto-Gonzalez, J., & Ramos, M. (2018, February). Deep learning with dense random neural networks for detecting attacks against iot-connected home environments. In *International ISCIS Security Workshop* (pp. 79-89). Springer, Cham.
[48] Ahanger, T. A. (2017, March). An effective approach of detecting DDoS using Artificial Neural Networks. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (pp. 707-711). IEEE.
[49] Aljumah, A., & Ahamad, T. (2016). A novel approach for detecting DDoS using artificial neural networks. *International Journal of Computer Science and Network Security*, *16*(12), 132-138.
[50]"Powell, M.J.D., 'Restart procedures for the conjugate gradient method,' Mathematical Programming, Vol. 12, 1977, pp. 241–254."

[51]"Beale, E.M.L., 'A derivation of conjugate gradients,' in F.A. Lootsma, Ed., Numerical methods for nonlinear optimization, London: Academic Press, 1972."
[52] "Battiti, R., 'First and second order methods for learning: Between steepest descent and Newton's method,' Neural Computation, Vol. 4, No. 2, 1992, pp. 141–166."
[53] "Riedmiller, M., and H. Braun, 'A direct adaptive method for faster backpropagation learning: The RPROP algorithm,' Proceedings of the IEEE International Conference on Neural Networks, 1993."
[54] Bharadwaj, H., & Kumar, V. (2018). Comparative study of neural networks in path planning for catering robots. *Procedia computer science*, *133*, 417-423.