Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 218 (2023) 2420–2429

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

DDoS Detection using Deep Learning

Deepak Kumar[1] , R.K.Pateriya[2], Rajeev Kumar Gupta[3], Vasudev Dehalwar[4], Ashutosh Sharma[5]

[1, 2, 4] *Computer Science and Engineering Department, Maulana Azad National Institute of Technology, Bhopal, India, 462003*

[3] *Computer Science and Engineering Department, Pandit Deendayal Energy University, Gandhinagar, India, 382007*

[5] *Department of Computer Science & Engineering, University of Petroleum & Energy Studies, Dehradun, 248007, India*

Abstract

The network's infrastructure becomes more vulnerable to cyber-attacks as the number of services offered through the internet expands. The complexity of "Distributed Denial-of-Service (DDoS)" threats on the internet has recently increased, posing a challenge to typical protection systems. As a result, early identification and separation of network data is the most crucial part of protecting against DDoS threats. A "Long Short-Term Memory (LSTM)" based model is created in this study to identify DDoS threats on a sample of network traffic packets. LSTM is a deep learning technique that includes a feature selection and extraction algorithm. When trained, it updates itself; Even with a smaller number of data points, LSTM functions swiftly and correctly. Using the "CICDDoS2019 dataset" for training and testing, the suggested LSTM model can achieve an accuracy of up to 98 percent in the current work, and Deep learning exceeds machine learning on the CICDDoS2019 dataset.

Keywords: *DDoS; Long Short-Term Memory (LSTM); CICDDoS2019, Classification, Deep Learning*

1. Introduction

We live at a time when the "Internet of Things (IoT) " [1] affects almost every area of modern life. The Internet of Things is made up of a wide range of devices (things) with varying technological backgrounds, all of which are vulnerable to security threats. The security fundamentals and attributes of each entity differ from one another and finding a common solution that can securely fix all of the problems has become tough. Because of inadequate security systems, attackers can target IoT devices. Furthermore, banking and financial transactions, communication, e-commerce, shopping, online payment, healthcare, and education are all carried out online through the provision of services on the Internet [2]. Because of the widespread usage of the aforementioned services, it is more vulnerable to cyber-attacks. DDoS [3] are the most dangerous and frequent type of cyber-attack. It is interrupt many services

10.1016/j.procs.2023.01.217

which are possible through the internet.

DoS is an abbreviation for denial of service [4], which occurs when a system sends malicious communication to a server. A DDoS attack happens when a large number of computers or compromised systems, i.e. bots, do DoS attacks on one application. The targeted network is then bombarded with packets from all around the world.DDoS attacks are changing and expanding in terms of volume and complexity as disruptive Internet technologies proliferate [5]. Outages, data theft, and even ransom demand from an attacker are all instances of potential cyber threats that could have major ramifications for an organization's operations.

DDoS attacks should be avoided by taking action as soon as they occur. Because of the growing usage of the internet, cyber-attacks on internet-connected equipment have become an enticing target. Industry and Academics are researching the idea to put in ML and DL for DDoS detection as ML and DL [6] release their immense potential in several sectors. In risk identification, traditional methods have less accuracy and respond slowly. Threats can be detected using ML approach such as Random Forest [7], KNN [8], and Naive Bayesian [9] more efficiently and precisely. In machine learning, characteristics for categorization must be chosen by the subject expert. Feature selection happens inside the deep learning model. DL approach such as ANN [10], DNN, and RNN learn multiple patterns from a large number of labelled samples using a sequence of nonlinear types of layers. As a result, DL can be an effective DDoS detection method. In Section 2, we'll look at several examples of successful ML/DL DDoS detection.

After evaluating several choices, we opted to employ the LSTM Deep Learning architecture in our experiment. The time domain correlation is one of the key properties of DDoS traffic that LSTM can capture. Experiments indicate that it performs admirably for the intended function.
The goal of this paper was to detect network traffic using LSTM, a deep learning model, use to detect DDoS attacks on the CICDDoS2019 dataset. The following are the contributions that our research will make.

- It was attempted to develop a deep learning model with a possible higher accuracy than ML.
- It was found that the proposed model worked nicely with CICDDoS2019.
- For usage in DDoS intrusion detection systems, deep learning-based support is offered.
- We compare and evaluate the performance of ML and DL techniques for cyber defense.

The following is the paper's layout. The second section examines the related works. The deployment of IDSs is discussed in Section 3. A comparison between Deep Learning-based IDS and Machine Learning-based IDS is shown in Section 4. Lastly, in Section 5, we come to some conclusions.

2. Related Work

There are several machine learning techniques used for DDoS threat categorization, including K-Nearest Neighbor (KNN), Logistic regression, Random forest (RF), Support Vector Machine (SVM) and Naive-based classifiers. KNN finds the k nearest neighbors of the incoming data. The voting technique is used to categorize test data that has a significant number of votes of a similar type. N.H. employed KNN [11] to categorize the network's DDoS assault with excellent results. SVM creates a hyperplane in the transform domain based on labelled training data to classify unseen data. Cheng utilized SVN to distinguish between benign and malicious traffic based on the "IP Address Interaction Feature." This model does a good job of distinguishing between malicious and benign traffic [12]. Random Forest is a decision tree collection. Random forest categorization is performed on the basis majority vote of a vast number of decision trees. Zheng et al. employed Random Forest [13] to classify DDoS attacks and obtained acceptable classification performance given the correct feature set. A classification algorithm is known as a Naive Bayes classifier. The Bayes theorem underpins it. It worked perfectly when the functions are self-contained. The occurrence of one feature in a category is assumed to be independent of the presence of any other feature by a Nave Bayes classifier. Riadi used the NB approach [14] to find the DDoS attack using mean difference and standard deviation and they came up with a good result.

Dincalp discovers and extends clusters from high-density core samples. It's very well suited to data with comparable density clusters. Dincalp employed the DBSCAN clustering approach to deal with the variety of attack vectors. [15].

In their experiments, the recommended approach performed effectively with certain characteristics. The biological neural network is comparable to the ANN. Back-propagation is one of the approaches used by ANN to learn function mapping. Ahanger [16] develops ANN for detection of a DDoS attack.

In a study by Zubair Hasan, Sattar, and Zahid Hasan [17], the "Deep Convolution Neural Network (DCNN)" model is employed to fight DDoS attacks. Because machine learning algorithms are incapable of analyzing traffic with fewer data points, the DCNN model is appropriate in this situation. DCNN outperforms shallow machine learning algorithms like Naive Bayes, SVM, and KNN, which have accuracy ratings of 80%, 87 percent, and 92 percent, respectively, according to the data.

To assure SDN security, Krishnan [18] offers an attack detection system architecture that combines a deep learning model called Non-symmetric Deep Autoencoder (NDAE) and a shallow machine learning technique called Random Forest (RF). In contrast to the traditional autoencoder's encoder-decoder arrangement, NDAE simply has an encoder. To overcome the issues caused by the shallow machine learning model taking more training durations, and more memory and processing requirements, the usage of a deep learning model was selected. As a result, NDAE was taken since it has more accuracy while using less memory of CPU and taking less time to train. The NSL-KDD and CIC-IDS2017 datasets were used to assess the efficacy of the model employed to detect DDoS attacks. Using the NDAE model to the CIC-IDS2017 and NSL-KDD datasets yielded accuracy values of 99.60 % and 99.24 %, respectively, indicating NDAE is fit for usage in attack detection.

According to Zhu, Ye, and Xu [19], the FNN and CNN models, are deep learning techniques, that should be used to analyze traffic on the network and detect DDoS intrusions. In trials on the NSL-KDD dataset, deep learning models were found to have superior accuracy in distinguishing anomaly types and network anomaly identification than RF, RT, and SVM, these are machine learning techniques.
In the IDS that monitors abnormal activity on the network, Alzahrani and Hong [20] advocate using an ANN to detect DDoS attacks. When the Artificial Neural Network (ANN) was evaluated independently in the studies, find that using the ANN technique can find more accuracy. It may infer that the deep learning model is highly successful at analyzing network data and detecting DDoS attacks based on the findings of the above literature review.

3. Background and proposed work

The mathematical backgrounds of the suggested model as well as the structure of the deep learning model are discussed in this part.

A. Algorithm I

The DNN is made up of neural networks that don't have any feedback connections. The input and output layers, as well as the hidden layer (which can be several), are the main components of the DNN. Each layer contains units with weights. The activation procedures of the units from the previous layer are made by these units.

Each hidden layer having an activation function which is represented by equation 1

$$d(x)=f(x^t w + b) \tag{1}$$

The mathematical formula for the nested hidden layer is given by equation 2.

$$d(x) = d_i(d_{(i-1)}(d_{(i-2)}(\ldots.(d_1)))) \tag{2}$$

in equation 1 where d_i is a hidden layer,f is the activation function, x, y, and w are the input vector, output vector, and w weight vector respectively. Every hidden layer in the DNN model is composed of sigmoid activation functions [22], the mathematical formula for which is given in equation 3

$$\text{Sigmoid}(x) = \frac{1}{1+e^{-x}} \tag{3}$$

The outer layer of DNN has a softmax activation function [23], which is given in equation 4

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \quad (4)$$

The equation 5-7 represents the binary cross entropy [24] loss function,

$$q_0 = 1-\hat{y}\ ,\ q_1 = 1-\hat{y} \quad (5)$$

$$p_0 = 1-y\ ,\ q_1 = y \quad (6)$$

$$L = \sum_i p_i \log q_i = -y\log\hat{y} - (1-y)\log(1-\hat{y}) \quad (7)$$

Where L is the Loss function, pi is an actual probability, and qi is the predicted probability.

In our research, AdaMax was employed as an optimizer. The infinite norm is used to develop AdaMax [15]. The equation 8-10 shows the mathematical formula for AdaMax's optimization algorithm

$$m_t = \beta_1 m_{t-1} + (1-\beta_t) g_t \quad (8)$$

$$u_t = \max(\beta_2 u_{t-1}, |g_t|) \quad (9)$$

$$\theta_t = \theta_{t-1} - \left(\frac{\alpha}{1-\beta_1^t}\right)\frac{m_t}{v_t} \quad (10)$$

Where t is time, d_t is the gradient at time t, α is learning rate, m_t is a first-moment vector, β_1 β_2 ε [0,1] are exponential decay rate,ut is the exponentially weighted infinity norm, θ_t is revised value are denoted. At the starting time t=0, initially, mt and ut are zero.

B. Algorithm II

Recurrent neural networks are types of neural networks that process sequential input. It uses hidden states to allow prior predicted values to be used as inputs.RNN is a multilayer perceptron. It is acyclic. It is made up of input, output, and a hidden layer. The n layers that make up a multilayer perceptron are specified by the order in which they are formed. The Keras framework [9] was used to create our deep learning model. In order to manage data, how data is entered in a model, retained by the model and exit from the model LSTMs employ three gates named forget gate, an input gate, and an output gate. These three gates are the main component of the LSTM model and are responsible for its overall control. Based on the previously hidden state as well as the data point that is currently being processed by the sequencer, the forget gate will determine which portions of the LSTM should now be forgotten, according to eq11 (have less weight). The eq 12 represent the mathematical formula for the input gate which is responsible to determines which kinds of data are permitted to enter the network. This equation aims to compute what relevant information should be introduced to the network's LSTM (cell state) given the previous hidden state and fresh input data. The output gate is responsible for establishing the new hidden state. This choice is based on the newly updated cell state, the previous concealed state, and the new input data. by eq 13. "Wf, Wi, and Wo are the weight of forget gate, input gate, and output gate respectively". The information that the sigmoid function uses comes from the most recent input as well as the hidden layer that came before it, as indicated in Equation (14). The (tanh) function [25] is used to convey the input gate, current source, and prior hidden state data (15). The range of the sigmoid (σ) function varies from 0 to 1 and tanh function range varies from -1 to 1. If the sigmoid function value lies close to 1 then it keeps the data if it is close to 0 it removes the data.

The architecture of LSTM shown in Fig.1

$$F(t) = \sigma(w_f[h_{(t-1)}, X_t] + b_f \quad (11)$$

$$I(t) = \sigma(w_i[h_{(t-1)}, X_t] + b_i \quad (12)$$

$$O(t) = \sigma(w_o[h_{(t-1)}, X_t] + b_o \quad (13)$$

$$f(x) = \frac{1}{1 + e^{axt}} \quad (14)$$

$$\tanh(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (15)$$

$$\widehat{C_t} = \tanh\ (w_c, [h_{t-1}, X_t] + b_c) \quad (16)$$

$$c_t = F_t . C_{(t-1)} + I_t . \widehat{C_t} \quad (17)$$

$$h_t = O_t . \tanh\ (C_t) \quad (18)$$

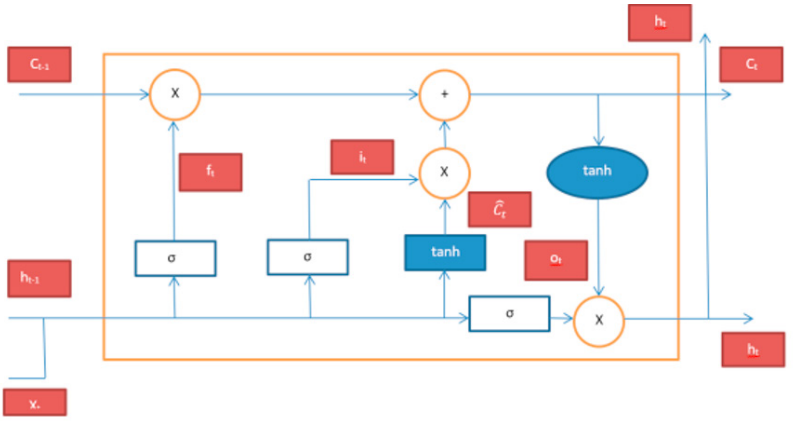

Fig 1. Architecture of LSTM

where $h_{(t-1)}$ represents the output of the previous hidden state, α is the learning rate, x_t is current input and b_f is the bias for forget, b_i is the bias for input gate, b_o is the bias for output gate.

4. Experiments and result

A. Testing Environment

Python 3.9, Tensorflow, and Sklearn were utilized in this experiment, and the machine employed had 400 CPU cores and 2 GPU nodes, providing the performance of 25.6 telops and 29.6 teflops Rpeak.

B . Dataset

Canadian Institute for Cybersecurity collected the CICDDoS2019 datasets [27] using Wireshark in emulated environments. They're made up of two sorts of usage profiles, as well as multistage attacks like Heartbleed and various DoS and DDoS attacks. The CICFlowMeter [28] is used to pre-process the collected traffic. It has 88 network traffic features, to generate a wide range of DoS and DDoS traffic statistics. The generated data collection is in CSV format and contains records of traffic features. The CIC-DDoS2019 dataset [29] contains approx fifty million sixty-three thousand one hundred twelve records, with fifty million six thousand two hundred forty-nine rows dedicated to DDoS attacks and approx. fifty-six thousand eight hundred sixty-three rows for benign. There are 88 features in each row. Network Time Protocol (NTP), Microsoft SQL Server (MSSQL), Domain Name System (DNS), The 12 DDoS Cyber-attacks in the training dataset are UDP-Lag, LDAP, NetBIOS, SSDP, SNMP,

SYN, UDP, WebDDoS, and TFTP. In the test, the dataset has seven different types of attacks, i.e MSSQL, SYN, PortScan, LDAP, NetBIOS, UDP-Lag, and UDP.

NTP-based attack: In this attack, the attacker uses the server capabilities of the "Network Time Protocol (NTP)" to flood a specific target or even other networks with UDP internet traffic. As a result of this attack, the target and its network infrastructure may become inaccessible to normal traffic.

A DNS-based DDoS attack is a reflection-based DDoS activity in which a person uses a Botnet to send a high number of resolve requests to a certain IP. An attacker makes queries to a publicly available susceptible server to create large replies, which are subsequently replicated to a targeted system in an LDAP-based assault.

MSSQL-based DDoS attack: In this, an attacker uses the “Microsoft SQL Server Resolution Protocol” to make programmed queries seem to come from the target server by using a falsified Internet Protocol address.

A complete reversal DDoS technique in which an attacker sends faked "Name Release" or "Name Conflict" signals to a target system to disable all network connectivity. SNMP-based attack: It uses the "Simple Network Management Protocol (SNMP)" to build attack quantities of tens of gigahertz to jam the target's networking pipes. SSDP-based attack: Using "Universal Plug and Play networking protocols”, the attacker transmits a large quantity of traffic to a chosen target.

UDP-Lag-based attack: In this sort of attack, the attacker overflows arbitrary channels on the target server using a Data packet containing UDP packets of data.

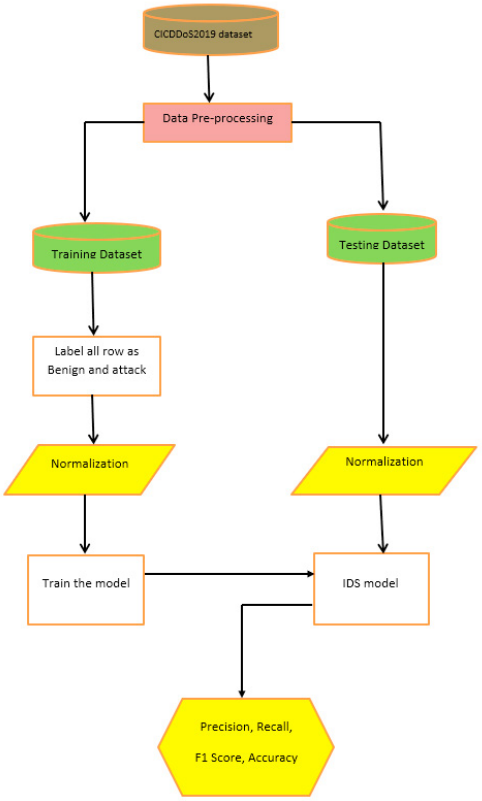

Fig 2. Functional Block Diagram

WebDDoS-based attack: This threat compromises a Web server or application by exploiting valid HTTP GET or

POST requests.

SYN-based assault: This is a form of attack in which an attacker leverages the Web's TCP/IP network connectivity to flood a target network with SYN requests, causing connection queues to overflow and the system to become unresponsive to user inquiries.

TFTP-based attack: In a TFTP-based assault, the attacker sends a default file demand to the victim TFTP server, which subsequently sends the data to the inquiring target host.

Port scanning-based attack: In a port scanning attack, network security is checked by scanning all access ports throughout the whole network. Find out which services are running on the host server via scanning.

C. Prepossessing

The dataset for CICDDoS2019 is in csv format. It is made up of a vast number of data packets. So, while importing the data, a random sampling technique is used to verify that the sample is random. We remove some rows from the dataset because they contain infinity rather than a numerical value. Seventeen features that are not must show a significant effect on accuracy remove from the dataset. For DDoS attack classification, the dataset is separated into two benign and attacks. In the dataset developed to detect a network attack, 'BENIGN' is set as '0′ while other attacks are set as '1′.

The attack types are grouped into two sections for classification purposes: exploitation and reflection-based attacks. To bring the data in the range of 0 to 1 normalization is done and to send the data to LSTM reshape is done.

D. Performance Matric

In order to properly evaluate machine learning and deep learning algorithms, it is essential to select the appropriate performance criteria. Precision (P), accuracy (A) recall (R) and F1-score (F1) are the performance indicators that we primary used in this work. Equation (19) calculates the precision, which shows out of positive predictions how many are correct. Its range lies between 0 to 1.

$$\text{Precision} = \frac{True\ \ Positive(TP)}{True\ Positive(TP)+False\ Positive(FP)} \tag{19}$$

The recall value is calculated using Equation (20), which displays how many true positives are accurately anticipated.

$$Recall = \frac{True\ \ Positive(TP)}{True\ Positive(TP)+False\ Negative(FN)} \tag{20}$$

Equation (21)'s accuracy reveals the model's correct prediction rate.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{21}$$

The F1 score is a harmonic mean of recall and precision that is derived using Equation (22).

$$\text{F1 score} = 2 * \frac{Recall*Precision}{Recall+Precision} \tag{22}$$

E. Result

We make use of the CICDDoS2019 Dataset, which is split into two portions training portion and a testing portion. The suggested LSTM model is evaluated using the train and test datasets. Table 1 shows the findings of KNN, DNN, and the suggested model LSTM, which are assessed for precision, recall, F1 score, and accuracy. The table 1 shows the comparative analysis of KNN, ANN and proposed approach.

Table 1: Performance analysis of KNN, ANN and proposed approach.

Model	P	R	F_1	A

KNN	0.89	0.91	0.92	0.89
ANN	0.94	0.93	0.94	0.93
Proposed Model	0.98	0.97	0.97	0.98

The performance experimental results of KNN, DNN, and proposed LSTM in term of Accuracy is shown in the bar graph.

When compared to KNN and DNN, the experimental trial using the suggested LSTM model with a learning rate of 0.0001 and epoch value 25 yields the greatest accuracy, 98 percent. Figure 3 depicts the relationship between training and validation accuracy, with epoch on the x-axis and accuracy on the y-axis. The graph shows that when the epoch value increases, the accuracy increases as well, proving that the suggested model is effective. Similarly, Figure 4 depicts the relationship between training and validation loss, with epoch on the horizontal axis and loss on the vertical axis. The graph shows that as the epoch number grows, the losses decrease, proving that the suggested model is effective.

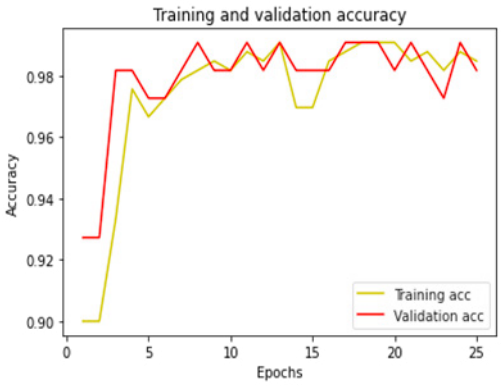

Fig 3. Plot between Accuracy and Epochs

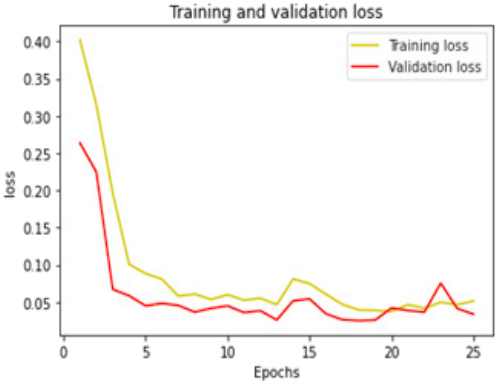

Fig 4. Plot between Loss and Epochs

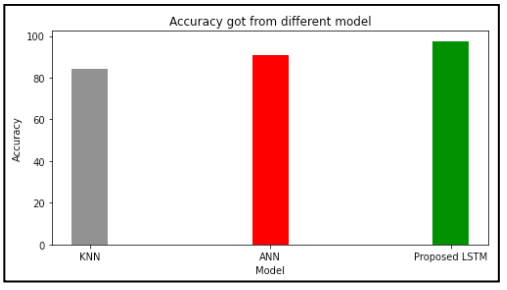

Fig 5. Comparison of different model

5. Conclusion and Future work

In this study, it is proposed that a DL model be used to classify DDoS attacks on the network, which is likely to be more effective than the machine learning model. The LSTM model was chosen as a viable model for this investigation as it encompasses both feature selection and extraction in its model, which makes it superior to shallow

machine learning methods. In the present research, the LSTM model has been used for the classification of benign and threats on the CICDDoS2019 dataset, the LSTM model, which is employed as a deep learning model, has approximately 98.6% for DDoS attacks classification which is large as compared to KNN and ANN model. Furthermore, using the CICDDoS2019 dataset with LSTM to detect DDoS attacks provides direction for other DDoS intrusion detection research. Because of its great accuracy in attack detection, it appears that incorporating the LSTM model into the software-based networks is a good option.

For future work, By capturing network traffic incorporates incremental learning. So the machine can update with a new type of attack.

References

[1] Kumar, Sachin, Prayag Tiwari, and Mikhail Zymbler. "Internet of Things is a revolutionary approach for future technology enhancement: a review." Journal of Big data 6.1 (2019): 1-21.

[2] Snehi, Manish, and Abhinav Bhandari. "Vulnerability retrospection of security solutions for software-defined Cyber–Physical System against DDoS and IoT-DDoS attacks." Computer Science Review 40 (2021): 100371.

[3] Chang, Rocky KC. "Defending against flooding-based distributed denial-of-service attacks: A tutorial." IEEE communications magazine 40.10 (2002): 42-51.

[4] Divyang Dave, Meet Kava, R. K. Gupta and Kaushal Shah, "Deep Learning approach for Intrusion Detection System, IEEE International Conference on Technology, Research, and Innovation for Betterment of Society (TRIBES), 2022. Doi: 10.1109/TRIBES52498.2021.9751643.

[5] R. K. Gupta et al., "An Improved Secure Key Generation Using Enhanced Identity-Based Encryption for Cloud Computing in Large Scale 5G", Wireless Communications and Mobile Computing 2022.

[6] Khuphiran, Panida, et al. "Performance comparison of machine learning models for ddos attacks detection." 2018 22nd International Computer Science and Engineering Conference (ICSEC). IEEE, 2018.

[7] Farnaaz, Nabila, and M. A. Jabbar. "Random forest modeling for network intrusion detection system." Procedia Computer Science 89 (2016): 213-217.

[8] Li, Yang, and Li Guo. "An active learning based TCM-KNN algorithm for supervised network intrusion detection." Computers & security 26.7-8 (2007): 459-467.

[9] Panda, Mrutyunjaya, and Manas Ranjan Patra. "Network intrusion detection using naive bayes." International journal of computer science and network security 7.12 (2007): 258-263.

[10] Yong, Li, and Zhang Bo. "An intrusion detection model based on multi-scale CNN." 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). IEEE, 2019.

[11]- Vu, N.H. DDoS attack detection using K-Nearest Neighbor classifier method. In Proceedings of the International Conference on

Telehealth/Assistive Technologies, Baltimore, Maryland, USA, 16–18 April 2008; IEEE: Piscataway Township, NJ, USA, 2008;

pp. 248–253.

[12]- Cheng, J.; Yin, J.; Liu, Y.; Cai, Z.;Wu, C. DDoS attack detection using IP address feature interaction. In Proceedings of the IEEE

International Conference on Intelligent Networking and Collaborative Systems, Thessalonika, Greece, 24–26 November 2010;

IEEE: Piscataway Township, NJ, USA, 2009; pp. 113–118.

[13] Wang, C.; Zheng, J.; Li, X. Research on DDoS attacks detection based on RDF-SVM. In Proceedings of the 10th International Conference on Intelligent Computation Technology and Automation, Changsha, China, 9–12 October 2017.

[14]- Fadlil, A.; Riadi, I.; Aji, S. Review of detection DDoS attack detection using Naïve Bayes classifier for network forensics. Bull.

Electr. Eng. Inform. 2017, 6, 140–148. [CrossRef]

[15] Dincalp, U. Anomaly based distributed denial of service attack detection and prevention with machine learning. In Proceedings of

the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies, Ankara, Turkey, 19–21 October 2018.

[16]- Ahanger, T.A. An effective approach of detecting DDoS using artificial neural networks. In Proceedings of the 2017 International

Conference onWireless Communications, Signal Processing and Networking, Chennai, India, 22–24 March 2017; IEEE: Piscataway Township, NJ, USA, 2017; pp. 707–711.

[17]- Zahid Hasan, Md., Zubair Hasan, K. M., & Sattar, Abdus (2018). Burst header packet flood detection in optical burst switching network using deep learning model. Procedia Computer Science, 143, 970–977.

[18]- Krishnan, Prabhakar, Duttagupta, Subhasri, & Achuthan, Krishnashree (2019). VARMAN: Multi-plane security framework for software defined networks. Computer Communications, 148, 215–239.

[19]- Zhu, M., Ye, K., & Xu, C. Z. (2018). Network anomaly detection and identification based on deep learning methods. In M. Luo, & L. J. Zhang (Eds.), Cloud Computing – CLOUD 2018. CLOUD 2018. Lecture Notes in Computer Science. Cham: Springer.

[20]- Alzahrani, S., & Hong, L. (2018). Detection of distributed denial of service (DDoS) attacks using artificial intelligence on cloud. In 2018 IEEE World Congress on Services (SERVICES), San Francisco, CA (pp. 35–36).

[21]- Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection system: A survey. Applied Sciences, 9, 4396. Beijing, China

[22] Pratiwi, Heny, et al. "Sigmoid activation function in selecting the best model of artificial neural networks." Journal of Physics: Conference Series. Vol. 1471. No. 1. IOP Publishing, 2020.

[23] Dunne, Rob A., and Norm A. Campbell. "On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function." Proc. 8th Aust. Conf. on the Neural Networks, Melbourne. Vol. 181. Citeseer, 1997.

[24] Murphy, Kevin (2012). Machine Learning: A Probabilistic Perspective. MIT. ISBN 978- 0262018029.

[25] Kingma, D. P., & Ba, J. (2015). Adam: a method for stochastic optimization. 3rd International Conference for Learning Representations, San Diego. Sharafaldin, I., Lashkari, A. H., Hakak, S., & Ghorbani, A. A. (2019).

[26] Chollet F., “Keras: Python Deep Learning Library,” https://keras.io, Last Visited, 2022.

[27] University of New Brunswick. DDoS Evaluation Dataset (CIC-DDoS2019). 2019. Available online: https://www.unb.ca/cic/

datasets/ddos-2019.html (accessed on 20 december 2021).

[28] Canadian Institute for Cybersecurity. CICFlowMeter (4.0) [Source Code]. 2016. Available online: https://github.com/

CanadianInstituteForCybersecurity/CICFlowMeter (accessed on 20 december 2021).

[29] Sharafaldin, I.; Lashkari, A.H.; Hakak, S.; Ghorbani, A.A. Developing realistic distributed denial of service (DDoS) attack dataset

and taxonomy. In Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India,

1–3 October 2019; pp. 1–8

[30] Powers, D. M. W. (2011). Evaluation: From precision, recall and fmeasure to roc, informedness, markedness and correlation. Journal of Machine Learning Technologies, 2(1), 37–63M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.