IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Security Attacks detection using Deep Learning Algorithms

G. Sankara Rao [1] , Dr. P. Krishna Subba Rao [2]
[1] Research Scholar, [2] Dean of Student Affairs
[2] Gayatri Vidya parishad college of Engineering (A), Visakhapatnam, India.

ABSTRACT:

Computer networks can be utilised efficiently for processing corporate data, teaching and learning, teamwork. The advancement of Internet technology has led to a variety of useful services being offered to users. We do, however, also face a number of security threats. An unauthorised infiltration into a computer in your company or an address in your designated domain is referred to as a network intrusion. There are several security breaches affecting computer and network systems today. These infractions may be the consequence of system compromise efforts by unauthorised external attackers or by inside privileged users abusing their power. No network is impermeable, and no firewall is error-proof. Attackers often create new exploits and attack methods intended to get past your defences. Network-borne information is more useful. An increasingly popular distribution method for attackers is the WWW. Attacks can now be launched with ease. Such attacks can occasionally cause a significant increase in network traffic. The five categories of network traffic are normal, denial of service assaults, user to root attacks, and probe (Probing attacks). As a result, the challenge of attacks classification can be thought of as involving intrusion identification. Accuracy of intrusion detection can be significantly increased by enhancing classifier performance in accurately recognising malicious traffic. To train and develop the model that can distinguish between attack-type network traffic and regular type network traffic, several deep learning techniques are used. To determine the optimum model for recognising network intrusion detection, the test accuracy of this model is compared with that of other machine learning techniques.

Keywords: NSL-KDD, packet traffic, machine learning, Dataset.

I. INTRODUCTION

The use of communication systems is essential to the daily lives of most people. Computer networks can be utilised efficiently for processing corporate data, teaching and learning, teamwork, acquiring large amounts of data, and entertainment. The current computer network protocol stack was created with the goal of making it transparent and user-friendly. As a result, a strong communication protocol stack was created. The protocol's adaptability has left it open to attacks initiated by attackers. This necessitates the need for on-going security and monitoring of computer networks. Network security risks, such as unlawful denial of service, lack of authenticity, and confidentiality breach, can be divided into the aforementioned three categories. Different types of DoS are described by a variety of words for embranchment. DDoS is one acronym that suggests the attack is coming from a number of unaffiliated, independent sources. DoS attacks include DDoS attacks as well. The ICMP (Ping) Flood, TCP-SYN Flood, and UDP Flood are the DDoS attack categories. The five categories of network traffic are normal and denial of service (DoS) assaults, root-to-local (R2L) attacks, user-to-root (U2R) attacks, and probe (Probing attacks). Consequently, one may categorise intrusion detection as a classification issue. Enhancing classifier performance in reliably identifying malicious traffic can dramatically improve intrusion detection accuracy.

.DDoS attack Categories are :

A. **ICMP (Ping) Flood** attack:

An attempt is made to overwhelm a targeted device or server with ICMP echo requests (pings) in a sort of Distributed Denial-of-Service (DoS) attack known as an Internet Control Message Protocol (ICMP) flood DDoS attack.).

B. **TCP-SYN Flood attack**

It is a specific kind of DDoS attack that could harm every system-connected internet-connected device. The sender repeatedly sends SYN requests while disregarding the victim host's answer and continuing to submit SYN queries from a bogus IP address after receiving a SYN-ACK from the victim host. Every request made by a dependable customer is turned down. The illustration shows it below.

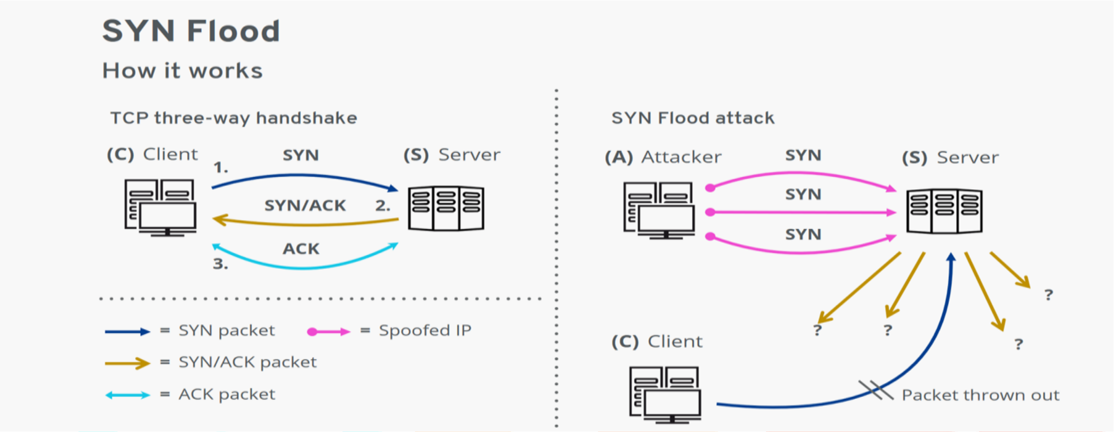


Figure 1) HOW SYN Flood attack happens

C. **UDP Flood attack**

In a volumetric Denial-of-Service (DDoS) attack known as a UDP flood, the victim is targeted by the attacker, who floods arbitrary ports with IP packets containing User Datagram Protocol (UDP) packets.

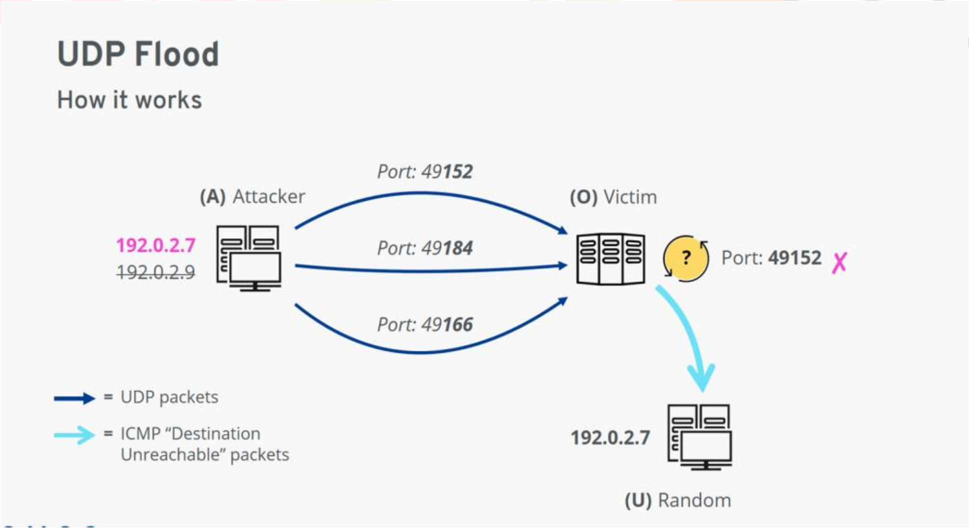


Figure 2) How UDP flood attack Happens

II. LITERATURE REVIEW

M. Zekri et al. [1] Proposed Attacks as DDoS (Distributed Denial of Service) are ones of the most frequent that inflict serious damage and affect the cloud performance. Thus, in this work, we designed a DDoS detection system based on the C.4.5 algorithm to mitigate the DDoS threat. This algorithm, coupled with signature detection techniques, generates a decision tree to perform automatic, effective detection of signatures attacks for DDoS flooding attacks. To validate our system, we selected other machine learning techniques and compared the obtained results. These 3 methodologies are used in this project they are DDoS Attack, Intrusion Detection Methodologies, and Machine Learning Techniques.

S. Wankhede [2] proposed The aim of this paper is to detect DoS attack effectively using Machine learning (ML) and Neural Network (NN) algorithms. The detection is specifically focused on application layer DoS attack detection rather than at transport and network DoS attack detection. The latest DoS attack dataset CIC IDS 2017 dataset is used in the experiment. The experimentation has divided the dataset into different splits and the best split is found for each algorithm i.e., RF and MLP. Results of RF and MLP are compared and it is shown that RF provides better results than MLP. Dataset used is CIC IDS 2017 DATASET.

YANG Lingfeng et al. [3] proposed DDoS attack detection system, the controller extracts the network traffic characteristics through statistical flow table information and uses the support vector machines (SVM) method to identify the attack traffic. The experiment is conducted using KDD99 dataset. The experiment results show the effectiveness of the DDoS attack identification method.

Shadman Latif, et al.[4] In this paper, the best machine learning algorithm, among the popular ones, is determined for a popular cyber security dataset (NSL-KDD).

Darsh Patel et al. [5] this article proposes a hybrid anomaly detection approach that detects irregularities in the network traffic implicating compromised devices by using only elementary network information like Packet Size, Source, and Destination Ports, Time between subsequent packets, Transmission Control Protocol (TCP) Flags, etc.

Mehdi Barati et al. in [6] this paper, the new hybrid detection method using Genetic Algorithm and Artificial Neural Network was proposed. And concluded that result of this study is very promising compared to previous studies.

Swain Sunita et al. [7] implemented an off-line intrusion detection system using Multi-Layer Perceptron (MLP) artificial neural network. And concluded that implemented system solved classification problem

Sagar Dhanraj Pande et al. [8] in this paper Primary focus was given to machine learning and deep learning technique for Detection of DDOS Attack.

Mavra Mehmood et al. [9] In this research, a detection system is projected on the NSLKDD dataset by applying data transformation and maximization and minimization method.

Xiaoyong Yuan et al. [10] proposed a deep learning based DDoS detection approach and found out that results shows that DeepDefense reduces error reate by 39.69%.

Sanjeev Kumar et al. [11] In this paper, different components used in Smurf-attack have

Been presented, and how the attack traffic is amplified towards the victim computer.

Baojun Zhou et al. [12] Proposed a ML based online internet traffic monitoring system using spark streaming to detect real time DDoS attacks and compared this approach with other methods Naïve Bayes, Logistic Regression and Decision Tree.

Ahmad Riza'ain Yuso et al. [13] have presented a feature selection algorithm for effective intrusion detection and found that his model has better accuracy and performance compared to other techniques.

Obaid Rahman et al. [14] Proposed J48, RF, SVM, KNN to detect and block the DDoS attack in an SDN network and their results showed that J48 performs better than other evaluated algorithms in terms of training and testing time.

Pourya Shamsolmoali et al. [15] presented a statistical technique to detect and filter DDoS attack and concluded that overall performance of C2DF in detection accuracy and time consumption is higher than existing models.

Yuan Tao et al. [16] focused on detecting DDoS flooding attacks in local area networks, and found that The proposed method does not have the pressure of storage for past packets analysis, nor is costly to the computing power of the routers.

Suman Nandi et al. [17] used a Hybrid approach that selects the top most important features , and concluded that this Hybrid approach gives the better DDoS detection rate compared to other methods.

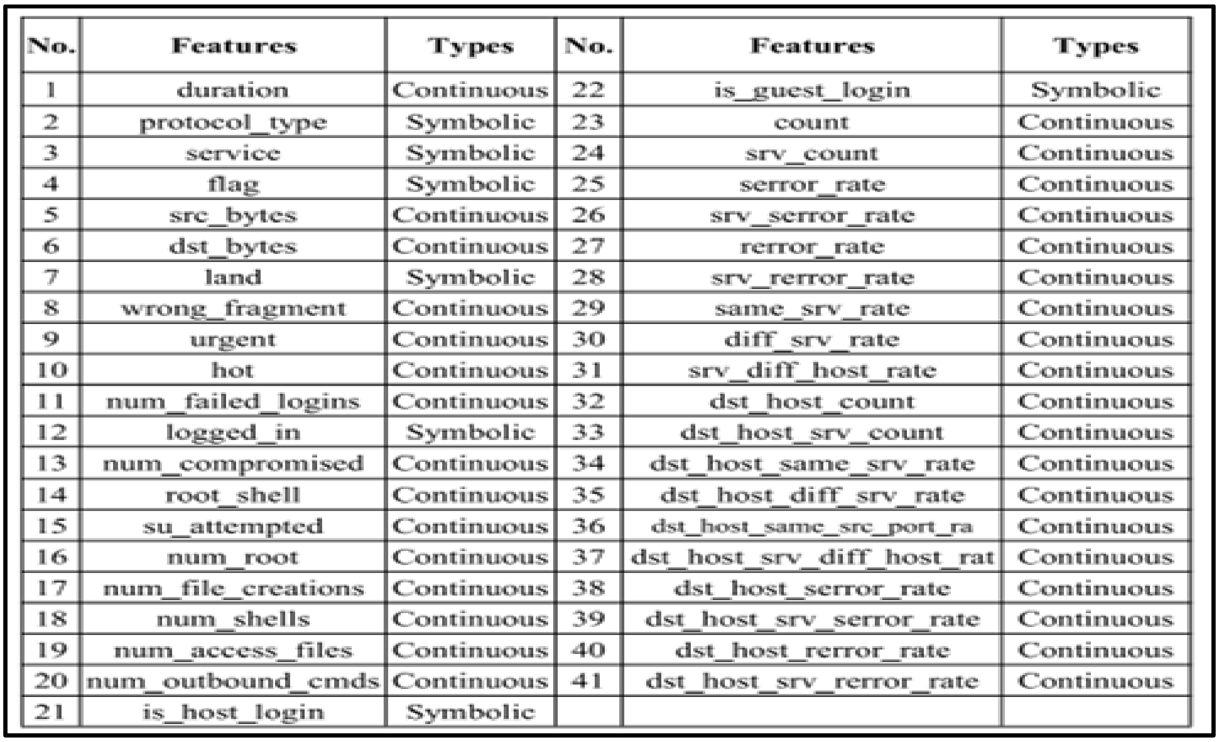
Chunyuan WU et al. [18] Focused on a specific type of network security attack – DDoS attack and visualizations in this class explore the pattern of the multi-dimensional data. It is very helpful in DDoS attack detection since it helps to decide the type of attack

Francisco Sales de Lima Filho et al. [19] presented the smart detection system an online approach to dos/ddos attack detection used random forest tree algorithm to classify network traffic and delivered improved DR,FAR,PREC.

B.S. Kiruthika Devi et al. [20] The HCF-SVM algorithm is employed to weed out spoofed traffic operating at victim end found The detection accuracy is high with reduced false positive.

III. DATASET DESCRIPTION

NSL-KDD dataset: NSL-KDD dataset is the benchmark dataset for modern day internet traffic. It is considered for my experiment. The NSL-KDD train dataset consists of 125,973 records and the test dataset contains 22,544 records. It does not include redundant records and the number of records in test and train sets are reasonable and make it affordable to run the experiments. NSL-KDD dataset consists of 42 features which contain information to predict the class of DDoS attack. The label marked for normal traffic is = 0 and the label for attack traffic is = 1. NSL-KDD dataset attributes are shown in below table 4.1.



No.	Features	Types	No.	Features	Types
1	duration	Continuous	22	is_guest_login	Symbolic
2	protocol_type	Symbolic	23	count	Continuous
3	service	Symbolic	24	srv_count	Continuous
4	flag	Symbolic	25	serror_rate	Continuous
5	src_bytes	Continuous	26	srv_serror_rate	Continuous
6	dst_bytes	Continuous	27	rerror_rate	Continuous
7	land	Symbolic	28	srv_rerror_rate	Continuous
8	wrong_fragment	Continuous	29	same_srv_rate	Continuous
9	urgent	Continuous	30	diff_srv_rate	Continuous
10	hot	Continuous	31	srv_diff_host_rate	Continuous
11	num_failed_logins	Continuous	32	dst_host_count	Continuous
12	logged_in	Symbolic	33	dst_host_srv_count	Continuous
13	num_compromised	Continuous	34	dst_host_same_srv_rate	Continuous
14	root_shell	Continuous	35	dst_host_diff_srv_rate	Continuous
15	su_attempted	Continuous	36	dst_host_same_src_port_ra	Continuous
16	num_root	Continuous	37	dst_host_srv_diff_host_rat	Continuous
17	num_file_creations	Continuous	38	dst_host_serror_rate	Continuous
18	num_shells	Continuous	39	dst_host_srv_serror_rate	Continuous
19	num_access_files	Continuous	40	dst_host_rerror_rate	Continuous
20	num_outbound_cmds	Continuous	41	dst_host_srv_rerror_rate	Continuous
21	is_host_login	Symbolic			

Table 4.1) NSL-KDD attributes

IV. METHODOLOGY

Our work plan for this paper can be visualized in below Figure 4.1). And the steps are explained as follows. The proposed system is used to classify the IP packet is Benign or DoS attack. Table 1) shows detection accuracy results.

`

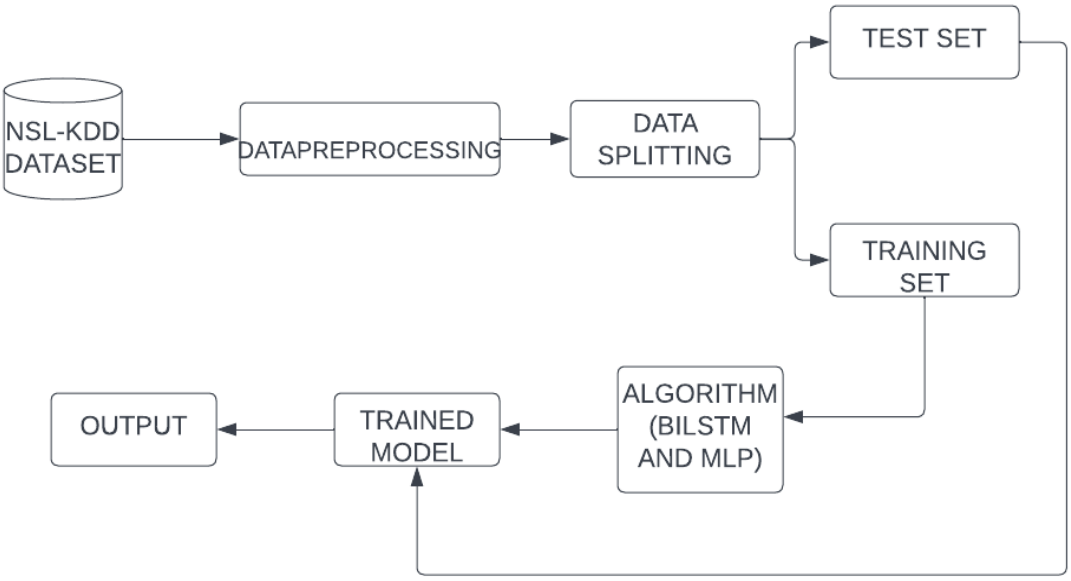


Figure 4.1) Proposed Model Architecture

Steps performed:

Dataset collection step:

First the NSL-KDD dataset is taken as input. On this NSL-KDD dataset, pre-processing techniques such as normalization and feature extraction were performed. Then the dataset will be splitted into training and testing data. Bidirectional LSTM and MLP models were applied to train the data to teach the machine. The testing set is then used to test the accuracy of the above trained model. Then it classifies as the normal class or attacker class

Pre-processing step:

The steps involved, with respect to processing the data are: pre-processing of data and normalization of data. In pre-processing, data from the dump area or from real-time environment which may consists of noise, inconsistent, incomplete, missing value, numeric and non-numeric data must be cleaned here. Data normalization is a method to convert the data vector into a new data vector where numeric values fall within a specified range, such as scaling values between [0,1]. There are many types of normalization such as min-max, z-score and decimal scaling normalization.

Training data:

The entire dataset is splitted into two parts: Training set and testing set. Training data (or a training dataset) is the initial data used to train machine learning models. Training datasets are fed to machine learning algorithms to teach them how to make predictions or perform a desired task. For supervised ML models, the training data is labelled. If it has poor training performance or poor validation performance then it goes back to the training dataset again.

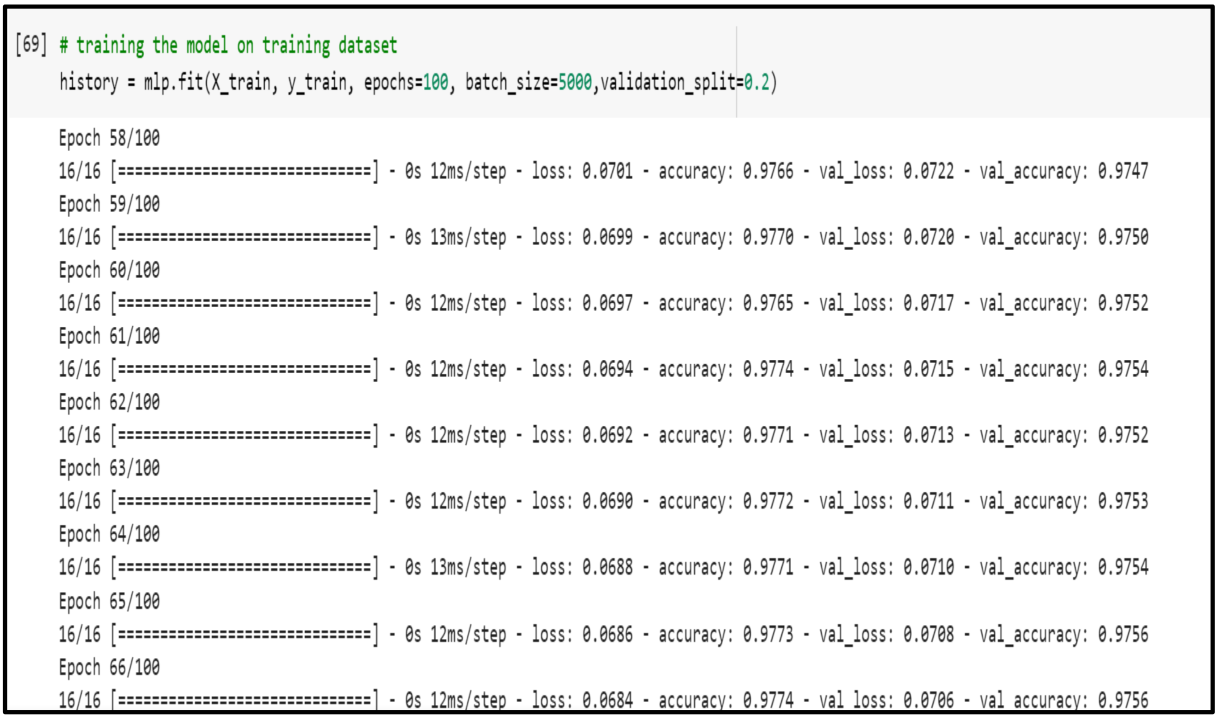
Testing data:

Once your machine learning model is built (with your training data); you need unseen data to test your model. This data is called testing data. The test set is a set of observations used to evaluate the performance of the model using some performance metric. It is important that no observations from the training set are included in the test set. If the test set does contain examples from the training set, it will be difficult to assess whether the algorithm has learned to generalize from the training set or has simply memorized it.

Classifies as Normal class or Attacker class:

If the training data contains poor performance then it increases complexity, if not then it goes to poor testing performance. If training data poor testing performance then it decreases complexity if not the data is tested again and then the input has to be uploaded. After that it classifies as normal or attacker class. If it is normal class it gives result .If it is attacker class then it classifies the type of attack and then returns result.

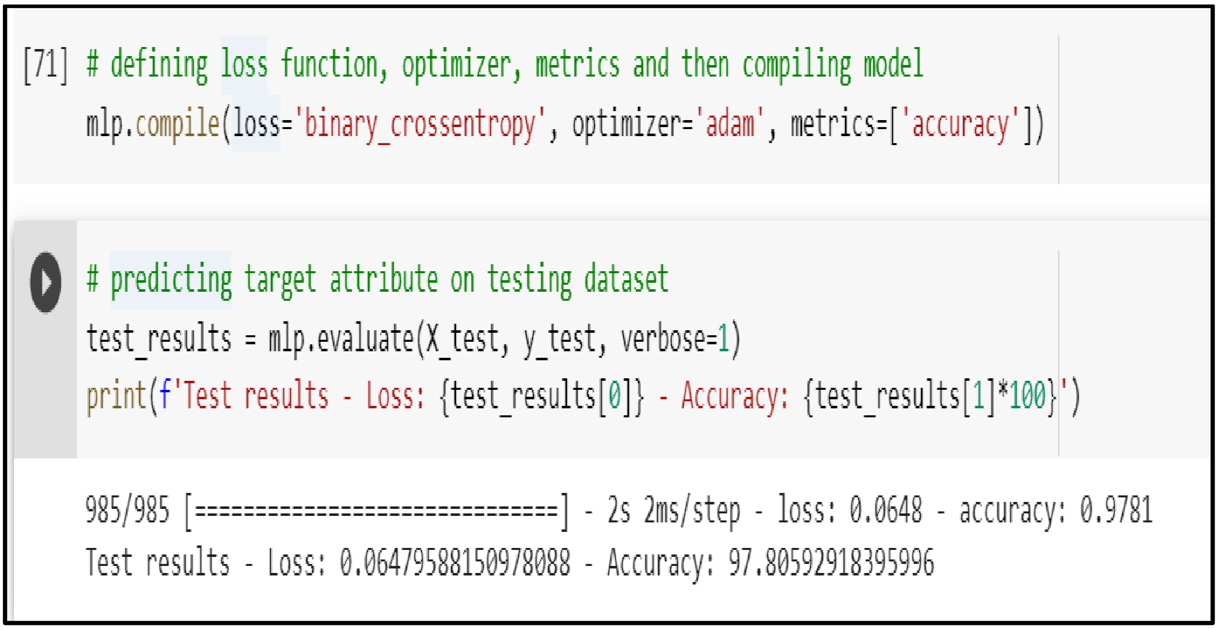
V. EVALUATION RESULTS



```
[69] # training the model on training dataset
     history = mlp.fit(X_train, y_train, epochs=100, batch_size=5000,validation_split=0.2)

     Epoch 58/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0701 - accuracy: 0.9766 - val_loss: 0.0722 - val_accuracy: 0.9747
     Epoch 59/100
     16/16 [==============================] - 0s 13ms/step - loss: 0.0699 - accuracy: 0.9770 - val_loss: 0.0720 - val_accuracy: 0.9750
     Epoch 60/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0697 - accuracy: 0.9765 - val_loss: 0.0717 - val_accuracy: 0.9752
     Epoch 61/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0694 - accuracy: 0.9774 - val_loss: 0.0715 - val_accuracy: 0.9754
     Epoch 62/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0692 - accuracy: 0.9771 - val_loss: 0.0713 - val_accuracy: 0.9752
     Epoch 63/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0690 - accuracy: 0.9772 - val_loss: 0.0711 - val_accuracy: 0.9753
     Epoch 64/100
     16/16 [==============================] - 0s 13ms/step - loss: 0.0688 - accuracy: 0.9771 - val_loss: 0.0710 - val_accuracy: 0.9754
     Epoch 65/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0686 - accuracy: 0.9773 - val_loss: 0.0708 - val_accuracy: 0.9756
     Epoch 66/100
     16/16 [==============================] - 0s 12ms/step - loss: 0.0684 - accuracy: 0.9774 - val_loss: 0.0706 - val_accuracy: 0.9756
```

Figure 4.2) Multilayer Perceptron training Results



```
[71] # defining loss function, optimizer, metrics and then compiling model
     mlp.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

     # predicting target attribute on testing dataset
     test_results = mlp.evaluate(X_test, y_test, verbose=1)
     print(f'Test results - Loss: {test_results[0]} - Accuracy: {test_results[1]*100}')

     985/985 [==============================] - 2s 2ms/step - loss: 0.0648 - accuracy: 0.9781
     Test results - Loss: 0.06479588150978088 - Accuracy: 97.80592918395996
```

Figure 4.3) Multilayer Perceptron Accuracy Result

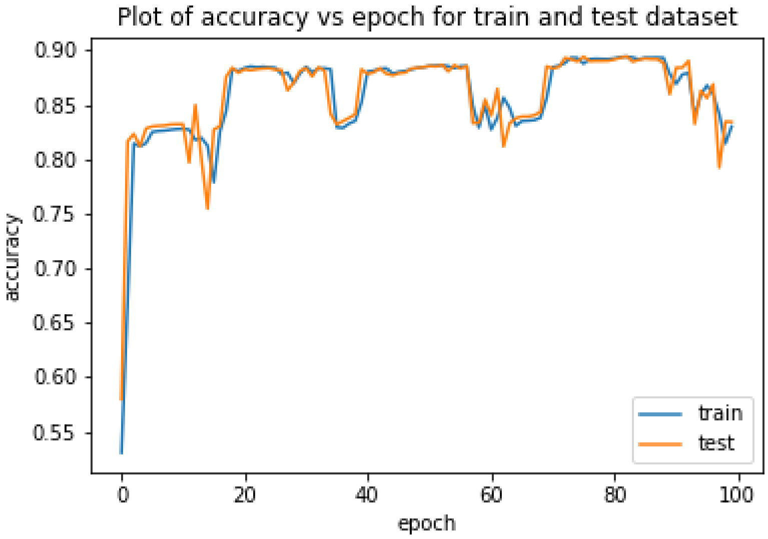


Figure 4.4) LSTM Binary Classification Accuracy

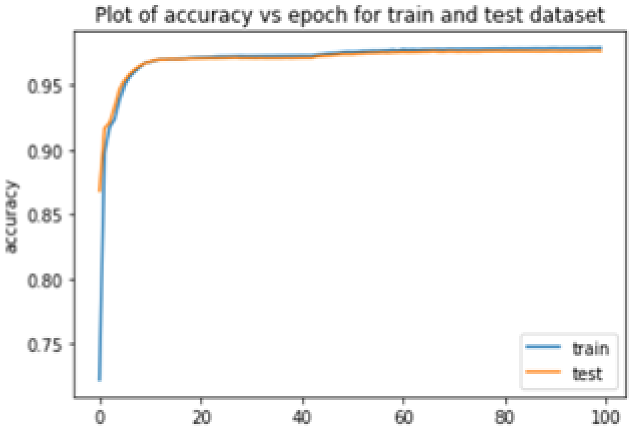


Figure 4.5) **Plotting accuracy vs. epoch graph**

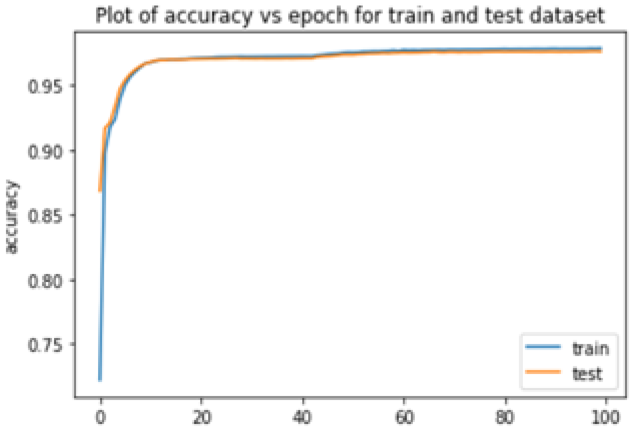


Figure 4.6) Plotting loss vs. epoch graph

Analysis: The performance of the proposed system is obtained using different classifier algorithms. The results achieved shown in above Figure 4.1, Figure 4.2, Figure 4.3, Figure 4.4, Figure 4.5, and Figure 4.6. It shows 97.8% accuracy in Multilayer Perceptron and LSTM shows 83.1% accuracy. Thus the proposed approach can satisfactorily detect network attacks like dos attack to protect the networks to make the networks more safeguard.

CONCLUSION

Instead of utilising machine learning rules or signatures, we suggested the model employing deep learning approaches for detecting network security attacks. In order to create an effective and adaptable network attack detection system, we proposed a deep learning-based technique. The NSL-KDD data set's processing of the data is extensive in order to reduce the number of false alarms. In order to choose this optimum model for network attack detection, the test accuracy of this model is compared with that of other machine learning techniques.

VI. REFERENCES

[1] M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, "DDoS attack detection using machine learning techniques in cloud computing environments," *Proc. 2017 Int. Conf. Cloud Comput. Technol. Appl. CloudTech 2017*, vol. 2018-Janua, pp. 1–7, 2018, doi: 10.1109/CloudTech.2017.8284731.

[2] S. Wankhede and D. Kshirsagar, "DoS Attack Detection Using Machine Learning and Neural Network," *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, 2018, doi: 10.1109/ICCUBEA.2018.8697702.

[3] L. Yang and H. Zhao, "DDoS attack identification and defense using SDN based on machine learning method," *Proc. - 2018 15th Int. Symp. Pervasive Syst. Algorithms Networks, I-SPAN 2018*, pp. 174–178, 2019, doi: 10.1109/I-SPAN.2018.00036.

[4] S. Latif, F. F. Dola, M. M. Afsar, I. Jahan Esha, and D. Nandi, "Investigation of Machine Learning Algorithms for Network Intrusion Detection," *Int. J. Inf. Eng. Electron. Bus.*, vol. 14, no. 2, pp. 1–22, 2022, doi: 10.5815/ijieeb.2022.02.01.

[5] D. Patel, K. Srinivasan, C. Chang, and T. Gupta, "Network Anomaly Detection inside Consumer Networks — A Hybrid Approach," pp. 1–12, doi: 10.3390/electronics9060923.

[6] M. Barati, A. Abdullah, N. I. Udzir, and ..., "Distributed Denial of Service detection using hybrid machine learning technique," *Biometrics ...*, pp. 268–273, 2014, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7013133/

[7] S. Sunita, B. J. Chandrakanta, and R. Chinmayee, "A Hybrid Approach of Intrusion Detection using ANN and FCM," *Eur. J. Adv. Eng. Technol.*, vol. 3, no. 2, pp. 6–14, 2016.

[8] S. D. Pande and A. Khamparia, "A Review on Detection of DDOS Attack Using Machine Learning and Deep Learning Techniques," *Think India J.*, vol. 22, no. 16, pp. 2035–2043, 2019, doi: 10.13140/RG.2.2.33777.63848.

[9] M. Mehmood *et al.*, "A hybrid approach for network intrusion detection," *Comput. Mater. Contin.*, vol. 70, no. 1, pp. 91–107, 2021, doi: 10.32604/cmc.2022.019127.

[10] X. Yuan, C. Li, and X. Li, "DeepDefense: Identifying DDoS Attack via Deep Learning," *2017 IEEE Int. Conf. Smart Comput. SMARTCOMP 2017*, pp. 1–8, 2017, doi: 10.1109/SMARTCOMP.2017.7946998.

[11] S. Kumar, "Smurf-based Distributed Denial of Service (DDoS) attack amplification in internet,"

Second Int. Conf. Internet Monit. Prot. ICIMP 2007, no. 0521585, 2007, doi: 10.1109/ICIMP.2007.42.

[12] B. Zhou, J. Li, J. Wu, S. Guo, Y. Gu, and Z. Li, "Machine-learning-based online distributed denial-of-service attack detection using spark streaming," *IEEE Int. Conf. Commun.*, vol. 2018-May, 2018, doi: 10.1109/ICC.2018.8422327.

[13] A. R. A. Yusof, N. I. Udzir, A. Selamat, H. Hamdan, and M. T. Abdullah, "Adaptive feature selection for denial of services (DoS) attack," *2017 IEEE Conf. Appl. Inf. Netw. Secur. AINS 2017*, vol. 2018-Janua, pp. 81–84, 2017, doi: 10.1109/AINS.2017.8270429.

[14] O. Rahman, M. A. G. Quraishi, and C. H. Lung, "DDoS attacks detection and mitigation in SDN using machine learning," *Proc. - 2019 IEEE World Congr. Serv. Serv. 2019*, vol. 2642–939X, pp. 184–189, 2019, doi: 10.1109/SERVICES.2019.00051.

[15] P. Shamsolmoali and M. Zareapoor, "Statistical-based filtering system against DDOS attacks in cloud computing," *Proc. 2014 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2014*, pp. 1234–1239, 2014, doi: 10.1109/ICACCI.2014.6968282.

[16] Y. Tao and S. Yu, "DDoS attack detection at local area networks using information theoretical metrics," *Proc. - 12th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. Trust. 2013*, pp. 233–240, 2013, doi: 10.1109/TrustCom.2013.32.

[17] S. Nandi, S. Phadikar, and K. Majumder, "Detection of DDoS Attack and Classification Using a Hybrid Approach," *ISEA-ISAP 2020 - Proc. 3rd ISEA Int. Conf. Secur. Priv. 2020*, pp. 41–47, 2020, doi: 10.1109/ISEA-ISAP49340.2020.234999.

[18] S. Sheng, C. Wu, and X. Dong, "Research on Visualization Systems for DDoS Attack Detection," *Proc. - 2018 IEEE Int. Conf. Syst. Man, Cybern. SMC 2018*, pp. 2986–2991, 2019, doi: 10.1109/SMC.2018.00507.

[19] F. S. De Lima Filho, F. A. F. Silveira, A. De Medeiros Brito Junior, G. Vargas-Solar, and L. F. Silveira, "Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning," *Secur. Commun. Networks*, vol. 2019, 2019, doi: 10.1155/2019/1574749.

[20] B. S. Kiruthika Devi, G. Preetha, G. Selvaram, and S. Mercy Shalinie, "An impact analysis: Real time DDoS attack detection and mitigation using machine learning," *2014 Int. Conf. Recent Trends Inf. Technol. ICRTIT 2014*, 2014, doi: 10.1109/ICRTIT.2014.6996133.