**NETWORK INTRUSION DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES**

*A*

*Report*

*Submitted in partial fulfilment of the*

*Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

**SUSHMITHA ADABOINA <1602-18-737-108>**

**UDAY KIRAN.S <1602-18-737-114>**

**VINUTHNA TATIKONDA <1602-18-737-118>**

**KUSUMA KODURU <1602-18-737-312>**

Under the guidance of

R. Dharma Reddy



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University) Ibrahimbagh, Hyderabad-31.**

**2020-2021**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**

****

**DECLARATION BY THE CANDIDATE**

I, **SUSHMITHA ADABOINA, UDAY KUMAR.S, VINUTHNA TATIKONDA, KUSUMA KODURU** bearing hall ticket number **1602-18-737-108, 1602-18-737-114, 1602-18-737-118, 1602-18-737-312** hereby declare that the project report entitled **“NETWORK INTRUSION DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES”** under the guidance of **R.DHARMA REDDY**, Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology.**

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Sushmitha Adaboina 1602-18-737-108

Uday Kumar.S 1602-18-737-114

Vinuthna Tatikonda 1602-18-737-118

Kusuma koduru 1602-18-737-312

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**

****

**BONAFIDE CERTIFICATE**

This is to certify that the project entitled “NETWORK INTRUSION DETECTION SYSTEM USING DEEP LEARNING TECHNIQUES” being submitted by SUSHMITHA ADABOINA, UDAY KUMAR.S, VINUTHNA TATIKONDA, KUSUMA KODURU bearing 1602-18-737-108, 1602-18-737-114, 1602-18-737-118, 1602-18-737-312 in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by him/her under my guidance.

**R. Dharma Reddy Dr. K. Ram Mohan Rao**

**Professor HOD, IT**

**Internal Guide**

**ACKNOWLEDGEMENT**

The satisfaction that accompanies that the successful completion of the project would not have been possible without the kind support and help of many individuals. We would like to extend my sincere thanks to all of them.

We would like to thank our coordinator Mr. Dharma Reddy sir who kept us motivated throughout to complete the project successfully. His ideas and insights helped us to make this project more operative.

We would also use this opportunity to thank our senior Sai Sri Vasthav Kalluri 1602-17-737-036. We are grateful to his guidance, and constructive suggestions that helped us in the preparation of this project. His constant guidance and willingness to share his vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks. We would like to thank all faculty members and staff of the Department of Information Technology for their generous help in various ways for the completion of this project.

Finally, yet importantly, we would like to express our heartfelt thanks to our HOD Dr. K. Ram Mohan Rao Sir and classmates for their help and wishes for the successful completion of this project.

**ABSTRACT**

Technology is rapidly evolving in a world driven by networks, online transactions and automated processes. The immense evolution in the technologies over the last decade has resulted in a large expansion in the network size, and the number of applications handled by the network nodes. As a result, a huge amount of important data is being generated and shared across different network nodes. The security of the this data and network nodes has become a challenging task due to the generation of a large number of new attacks either through the mutation of an old attack or a novel attack. Almost every node within a network is vulnerable to security threats. So there arises a need to detect and prevent these attacks. Both ML and DL are powerful tools in learning useful features from the network traffic and predicting the normal and abnormal activities based on the learned patterns.

The ML-based IDS depends heavily on feature engineering to learn useful information from the network traffic. While DL-based IDS do not rely on feature engineering and are good at automatically learning complex features from the raw data due to its deep structure. The decision can be made by using the data from the classification part achieved through the Sequential model. There are different malware datasets available publicly for further research by cyber security community. Appropriate data from network is captured and it is stored as a ‘.csv’ file and is fed to the implemented Deep learning model to predict the attack in a real time manner, thus detection is achieved. The internal behavior of this network is carefully tracked and tuned by using plotting and exploring codes until it reaches a functional peak in intrusion prediction accuracy. This process is done using real-time traffic monitoring to find out if any unusual behavior is present in the network or not. Monitoring data and analyzing it over time are essential to the process of predicting future events, such as risks, attacks and diseases.

**TABLE OF CONTENTS**

1.Introduction ..............................................................................................8

a.Information about the problem domain..........................................8

b.Priority of the project……………………………………………..9

2. Related work ..........................................................................…….........10

3. Proposed Work…………………………………………………………12

a.Use Cases…………………………………………………………12

b.UI Screenshots…………………………………………………....12

c.Architecture and technology used………………………………..13

d.Design………………………………………………………….....15

e.Implementation…………………………………………………...16

a. Description of main modules and functions………….…...16

b.Algorithm………………………………………………….18

c.Github link………………………………………………....20

f.Testing……………………………………………………………..21

g.Failure Analysis………………………………………………..…24

4.Results…………………………………………………………..……….25

5.Conclusion and Future Scope………………………………………..….28

6.References…………………………………………………………….…29

**LIST OF FIGURES:**

|  |  |
| --- | --- |
| **Figure Numbering** | **Figure description** |
| Fig 1 | Sample Network Traffic Capture |
| Fig 2 | Cicflowmeter |
| Fig 3 | Architecture of the neural network |
| Fig 4 | Internal nodes of the network |
| Fig 5 | Flow of the entire project |
| Fig 6 | Flowchart showing the development of model |
| Fig 7 | Code snippet of label encoding |
| Fig 8 | Bar graph showing frequency of label values |
| Fig 9 | Code snippet of applying Normalizer |
| Fig 10 | Code snippet of applying to\_categorical |
| Fig 11 | Code snippet of the Model |
| Fig 12 | Code snippet of training the model |
| Fig 13 | Evaluation on test data(25% data) |
| Fig 14 | Prediction of classes |
| Fig 15 | Confusion matrix |
| Fig 16 | roc\_auc score calculation |
| Fig 17 | Plot of roc curve |
| Fig 18 | Plot showing accuracy |
| Fig 19 | Plot showing loss |
| Fig 20 | Wireshark tool capturing live data |
| Fig 21 | Cicflowmeter converting pcap to csv |
| Fig 22 | Loading real time traffic dataset |
| Fig 23 | Final results |

**LIST OF TABLES:**

|  |  |
| --- | --- |
| **Table Numbering** | **Table description** |
| Table 1 | NSL KDD Performance Scores |
| Table 2 | CicIds 2017 Performance Scores |

1. **INTRODUCTION:**
2. **Information about the Problem Domain**

The domain we have chosen is Cyber Security.

There may be many possibilities of a malicious attempt to damage or disrupt a computer network or system. Cyber security is the practice of defending computers, servers, mobile devices, electronic systems, networks, and data from malicious attacks. It is also known as information technology security or electronic information security.

With an increasing number of users, devices and programs in the modern enterprise, combined with the increased deluge of data -- much of which is sensitive or confidential -- the importance of Cyber Security continues to grow. The growing volume and sophistication of cyber attackers and attack techniques compound the problem even further. Cyber-attackers use illegal methods, tools and approaches to cause damages and disruptions or gain unauthorized access to computers, devices, networks, applications and databases.

Big organizations with many employees have a higher risk of getting attacked. This is because of two reasons. Firstly, because the data is of huge importance and secondly even if a single employee does not follow the security protocol the entire organization gets into trouble.

 A normal consumer, who is connected to the internet if not vigilant enough, can be trapped easily. Because it is just a matter of minutes to hack an internet-connected device and that is why Cyber Security matters.

The most difficult challenge in cyber security is the ever-evolving nature of security risks themselves. Cyber security is a critical business issue for every organization. Companies need cyber security to keep their data, finances, and intellectual property safe. Individuals need it for similar reasons, although intellectual property is less of a factor, and there is a higher risk of losing important files, such as family photos.

1. **Priority of the Project**

Our aim is to develop an Intrusion Detection System to monitor network traffic where the network administrator will come to know which packet flow is normal and which is an attack. It is done by analyzing data to detect any attack toward a system or a network.

Dataset used : CIC-IDS 2018

The traffic data for labeled dataset is collected in a confined, isolated and private network environment by Canadian Institute of Cyber Security.

They classified network traffic as an intrusion when there is a deviation from the normal traffic pattern.  There exist variant kinds of attacks for cyber systems, such as flooding, distributed denial of service, abnormal packet attack, and spoofing. These attacks are performed deliberately to prepare the dataset CIC-IDS. Each flow in the network classified as either bot(attack) or benign(normal).

We achieve to design a deep neural network model capable of distinguishing the network flow as either attack or normal. We would like to perform training of the dataset so that the model could predict the flow as attack or normal when used in real time analysis. Cicflowmeter and Wireshark are the open-source tools available for capturing real time network traffic flow and they store the collected information in .pcap format. It can be converted to .csv file later and then the developed model can be used on that data. With this, the results can be obtained.

It can be used in any organization involving devices connected to same network.

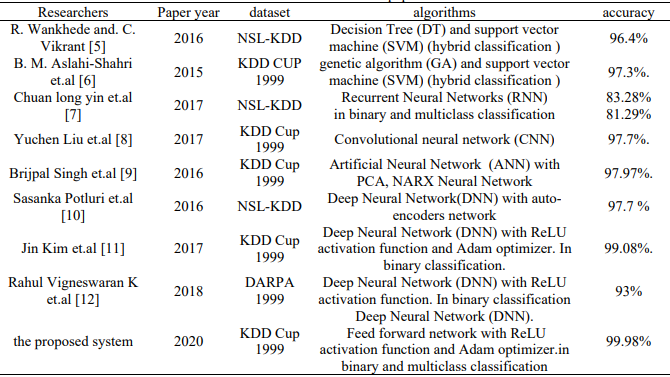
It is very similar to a house alarm system which will sound an alarm if an intruder attempts to break into a window or a door. For example, if a hacker attempts to gain access to your computer or network, the intrusion detection system will immediately notify the network administrator of the attempted security breach. Once reported, the manager can pinpoint the exact location of the suspicious activity and follow the proper safety protocols.

The network IDS monitors network traffic and triggers alerts when suspicious activity or known threats are detected, so IT personnel can examine more closely and take the appropriate steps to block or stop an attack.

**2 RELATED WORK**

There are many research trends to a topic of Intrusion Detection System using deep learning and machine learning techniques.

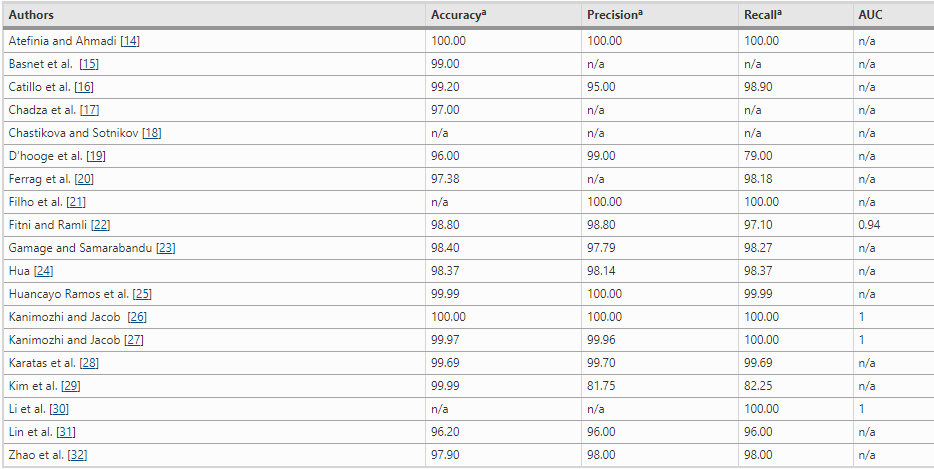
Binbusayyis and Vaiyapuri, mainly focused on creating an ensemble for feature selection using different evaluation measures, that can implement an intrusion detection system. Particularly, they proposed a set of feature selection and feature extraction and developed an IDS model by using the learning algorithm, Random Forest. The evaluation was done on various evaluation datasets, namely, KDDCup’99, in order to demonstrate the effectiveness of the proposed model. The results revealed that the specific subset of features is promising due to the final high performance metrics, achieving 99.98% accuracy, compared to other approaches. The dataset being used is from Network Security Laboratory. Table 1 shows work of Binbusayyis and Vaiyapuri with improved accuracy 99.98%.



**Table 1.** NSL KDD Performance Scores

NSL KDD trained model is rarely used in real time scenarios.

Therefore, many researchers started using a more realistic dataset which is CIC IDS.



# Table 2. CICIDS 2017: Performance scores

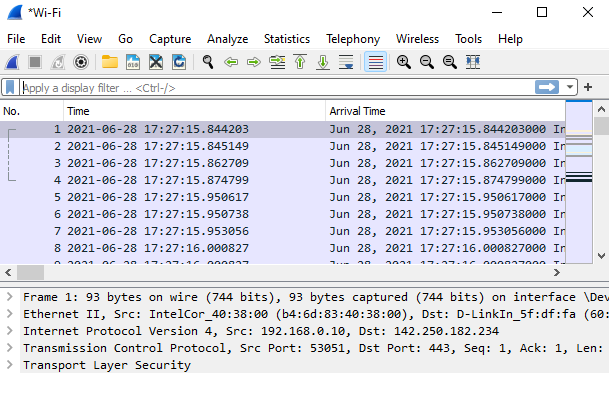
In recent literature, most of the studies in flow-based Intrusion Detection Systems based on machine-learning technologies are using the CICIDS2017 dataset, for the training and the evaluation. However, due to the new entrant dataset in the field of cyber security, there are limited published studies yet.

The exponential growth in computer networks and network applications worldwide has been matched by a surge in cyber attacks. For this reason, datasets such as CSE-CIC-IDS2018 were created to train predictive models on network-based intrusion detection. These datasets are not meant to serve as repositories for signature-based detection systems, but rather to promote research on anomaly-based detection through various machine learning approaches. CSE-CIC-IDS2018 contains about 16,000,000 instances collected over the course of ten days. It is the most recent intrusion detection dataset that is big data, publicly available, and covers a wide range of attack types. About 17% of the instances is attack traffic.

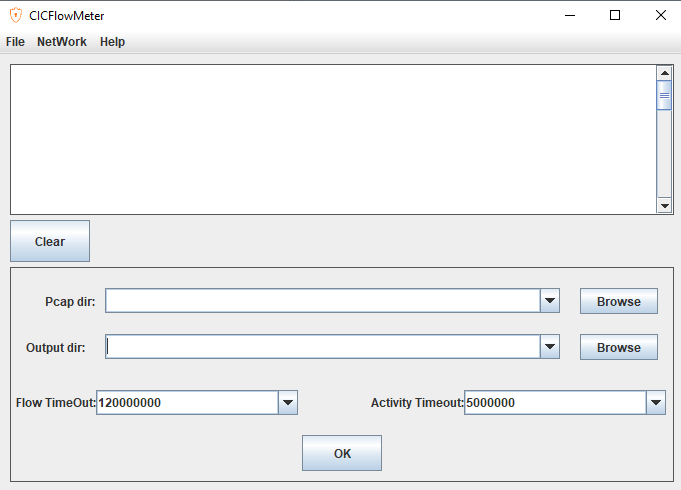
Researchers V.kanimozhi and T.Prem Jacob have used it and obtained accuracy of 99.95% but they dint mention about using it in real time. In the method we proposed, latest dataset CIC-IDS 2018 is being used and we are going to use a better model to achieve the desired accuracy and also we show the real time prediction.

**3 PROPOSED WORK:**

1. **Use Cases/Deliverables of the project:**
2. The developed neural network
3. Classification of the network flow as normal or attack.
4. Live Data must be obtained using Wireshark packet sniffing tool(open-source) and fed into the model.
5. **UI Prototypes/Screenshots:**



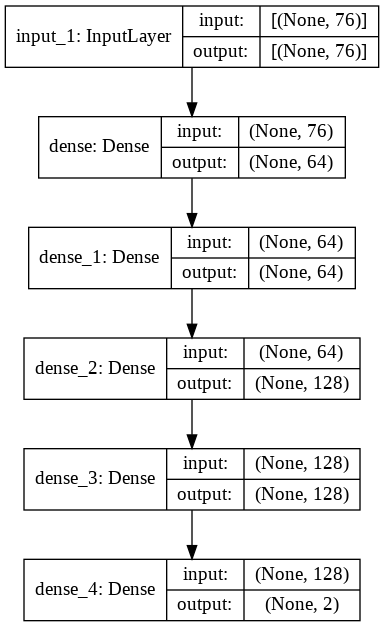
**Fig 1.**Sample Network Traffic Capture

** Fig 2.** Cicflowmeter

Ciciflowmeter is used to convert pcap file into csv with required features.

1. **Architecture and Technology used:**

Sequential network is built with input size same as number of columns. Output Layer has two neurons as binary classification is done.



**Fig 3**. Architecture of the neural network



**Fig 4.** Internal nodes of the network

**Hardware specifications:**

Executed on Intel® core™ i3-4030 CPU @ 1.90GHz, installed memory of 8GB RAM , 64-bit Operating system, x64 based processor.

**Technology used for developing model**:

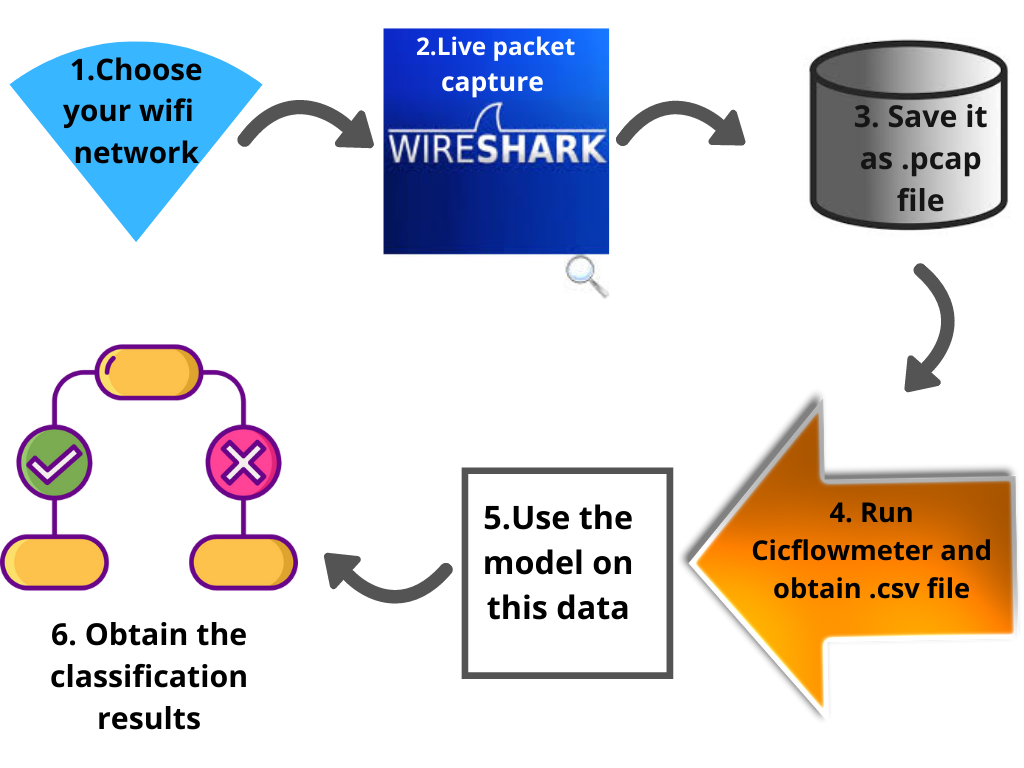
Python 3.9

Tensorflow version 2.5.0

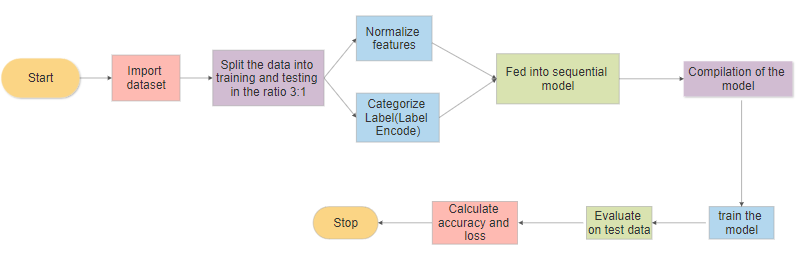
**Open-source tools used for live network traffic capture:**

Wireshark, Cicflowmeter 4.0

1. **Design:**



**Fig 5.** Flow of the entire project



**Fig 6.** Flowchart showing the development of model

1. **Implementation:**
2. Description of main modules and functions:

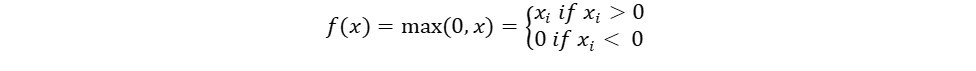
For preprocessing:

1. Label Encoder: It is available in ‘sklearn.preprocessing’ library. It Encode target labels with value between 0 and n\_classes-1. This transformer should be used to encode target values, i.e. y, and not the input X. As we are using binary classifier, the labels are encoded as 0 and 1. 0 for ‘Benign’ and 1 for ‘Bot’.

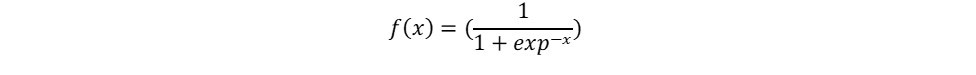
1. Normalizer : It is also available in ‘sklearn.preprocessing’ library. It Normalize samples individually to unit norm. Each sample (i.e. each row of the data matrix) with at least one non zero component is rescaled independently of other samples so that its norm (l1, l2 or inf) equals one. This transformer is able to work both with dense numpy arrays and scipy.sparse matrix (use CSR format if you want to avoid the burden of a copy / conversion). Here, we applied normalize for 76 features after removing redundant columns and columns having NaN values.
2. To\_Categorical : This module is available in tensorflow.keras.utils library.Using the method to\_categorical(), a numpy array (or) a vector which has integers that represent different categories, can be converted into a numpy array (or) a matrix which has binary values and has columns equal to the number of categories in the data.
3. Test\_train\_split: Split arrays or matrices into random train and test subsets. Here, the input data is divided in the ratio 3:1 i.e 75% training and 25% testing.

For the model:

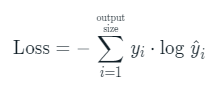
1. Sequential() : It the model which is being used to build the neural network and it is available in tensorflow.keras module. A Sequential model is appropriate for **a plain stack of layers** where each layer has **exactly one input tensor and one output tensor**.
2. Activation function ReLu: The Rectified linear unit (ReLu) activation function has been the most widely used activation function for deep learning applications. The main idea behind the ReLu activation function is to perform a threshold operation to each input element where values less than zero are set to zero. Mathematically it is defined by Eq(1):

 (1)

1. Activation function sigmoid: Using a mathematical definition, the sigmoid function takes any range real number and returns the output value which falls in the range of 0 to 1. Based on the convention, the output value is expected to be in the range of -1 to 1. Mathematically, sigmoid is represented by Eq(2):

(2)

1. Optimizer adam : Adam is an adaptive learning rate optimization algorithm that’s been designed specifically for training deep neural networks. Adam is a popular algorithm in the field of deep learning because it achieves good results fast. Adam is an adaptive learning rate method, which means, it computes individual learning rates for different parameters.
2. Categorical\_cross\_entropy: The categorical crossentropy loss function calculates the loss of an example by computing the following sum:The categorical crossentropy is well suited to classification tasks, since one example can be considered to belong to a specific category with probability 1, and to other categories with probability 0.

Mathematically represented as :  (3)

where *y*^​*i*​ is the i-th scalar value in the model output, *yi*​ is the corresponding target value, and output size is the number of scalar values in the model output.

1. Algorithm (for the model):

Step 1: Import the CiC Ids 2018 dataset containing the network flows collected over 10 days (available in kaggle.com)

Step 2: Encode the label column using LabelEncoder and separate it from the dataframe.

Step 3: Remove redundant columns having NaN and null values.

Step 4: Split the dataset in into training and testing data (3:1 ratio)

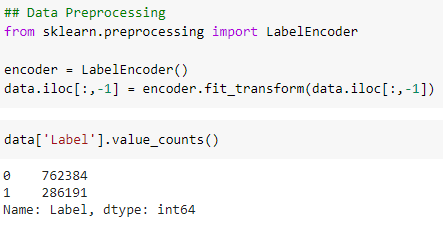
Step 5: Normalize the features and categorize the label using normalize and to\_categorical functions.

Step 6: Create sequential model with 1 input layer, 3 hidden layers and 1 output layer.

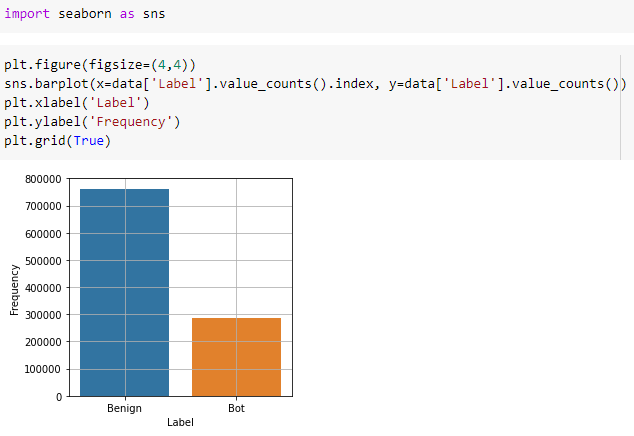
Step 7: Compile the model and evaluate on the test data.

Step 8: Calculate auc\_roc score and plot confusion matrix and other related graphs.

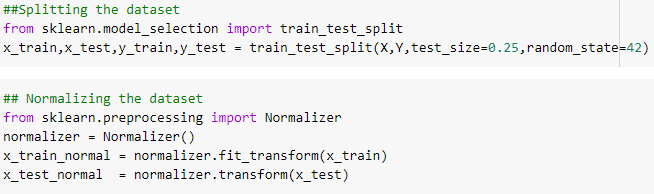
Important Code Snippets :



**Fig 7.** Code snippet of label encoding



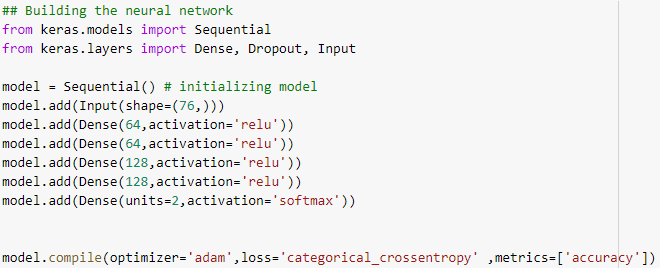
**Fig 8.** Bar graph showing frequency of label values



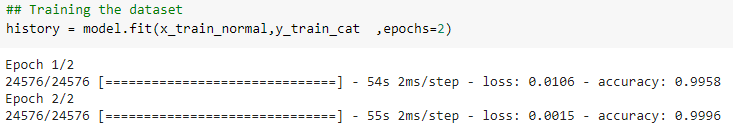
**Fig 9.** Code snippet of applying Normalizer



**Fig 10.** Code snippet of applying to\_categorical



**Fig 11.** Code snippet of the Model



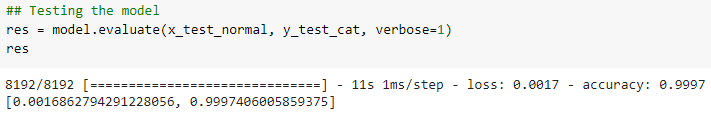
**Fig 12.** Code snippet of training the model

1. Github Link :

https://github.com/Vinuthna123/Network-Intrusion-Detection-System-using-Deep-Learning.git

**f. Testing :**

After evaluating the model on the 25% test data, accuracy obtained is 99.97 % and loss is 0.16 % as shown in figure



**Fig 13.** Evaluation on test data(25% data)

Prediction is done and true labels and predicted labels are compared.

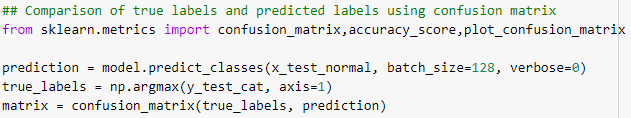
Here we consider ‘normal’ or ‘benign’ as negative class since there is no attack and ‘bot’ as positive class.

True Negatives(TN) –Samples correctly classified as ‘benign’ class

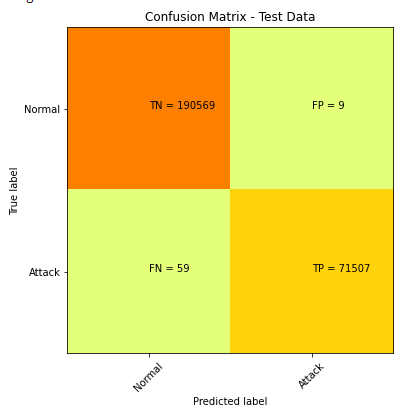
False positives –Samples misclassified as ‘bot’

False Negatives –Samples misclassified as ‘benign’

True Positives –Samples correctly classified as ‘bot’ class



**Fig 14.** Prediction of classes

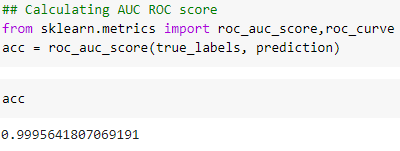


**Fig 15.** Confusion matrix

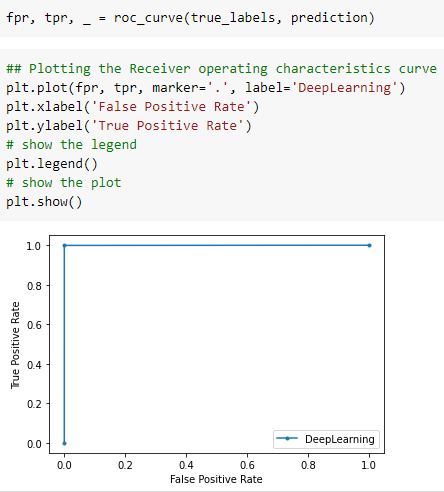
Receiver Operating Characteristics Curve(ROC) :

ROC curves typically feature true positive rate on the Y axis, and false positive rate on the X axis. This means that the top left corner of the plot is the “ideal” point - a false positive rate of zero, and a true positive rate of one.It is shown in figure

roc\_auc\_score - Compute Area Under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores.

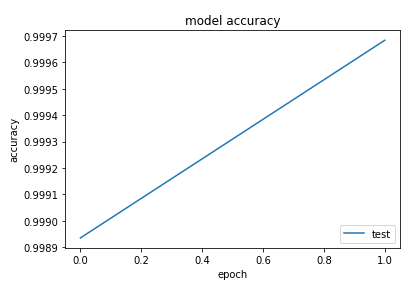


**Fig 16.** roc\_auc score calculation

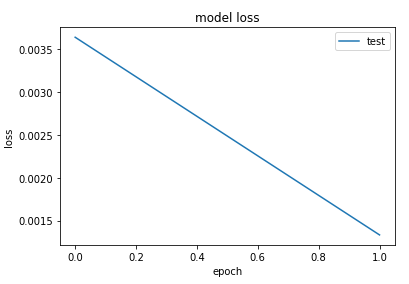


**Fig 17.** plot of ROC curve

Two epochs and in each of them 24576 steps were used, the accuracy gradually increased and real part remained 99 % from the beginning. Loss reduced from 0.35 to 0.15



**Fig 18.** Plot showing accuracy



**Fig 19.** Plot showing loss

1. **Failure Analysis:**

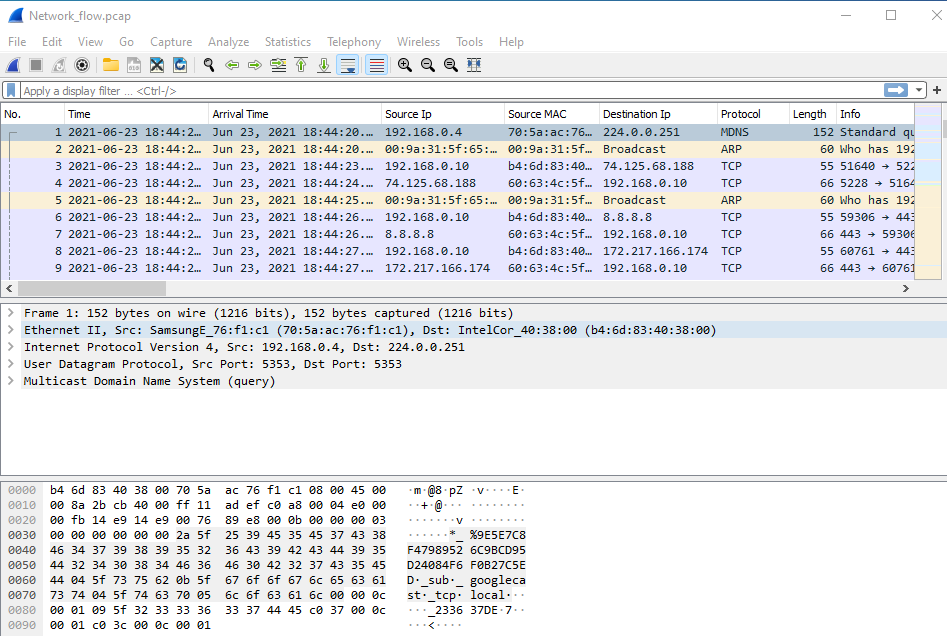
The model is unable to predict correct class for some rows.

9 rows which should be normal are classified as attack and 59 rows are misclassified as normal. According to our analysis from these rows, this happened because the values are different compared to some of the preceding and succeeding rows which are normal. Also, they were quite different from the pattern machine has already learnt during training, due to which overfitting happened and the model becamed biased hence, it was misclassified as normal.

**4 RESULTS:**

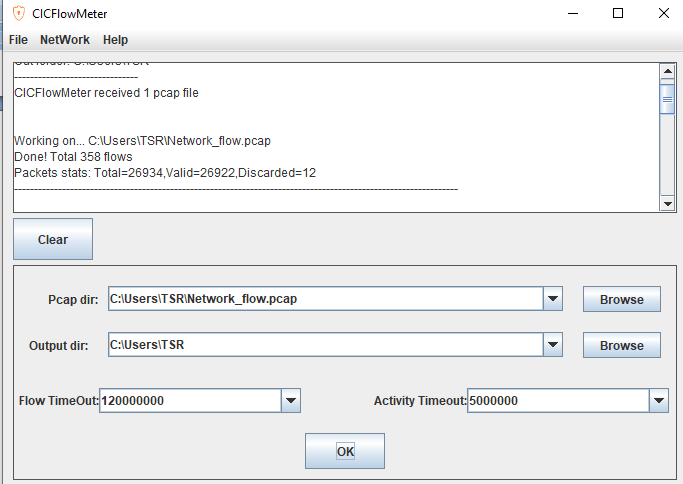
The motive of the project is to use it in real time and we tested our developed model on real time data.

1.Capture of live network traffic:



**Fig 20.** Wireshark tool capturing live data

2.Send this file to Cicflowmeter to obtain .csv file with required features

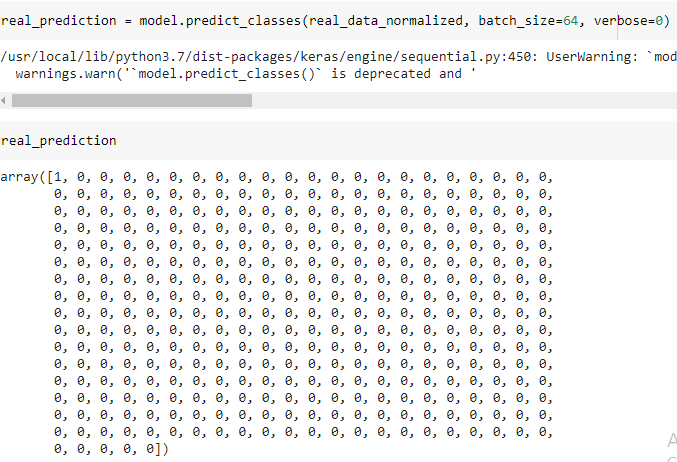


**Fig 21.** Cicflowmeter converting pcap to csv

3.Feed the data into the developed model and observe the prediction



**Fig 22.** Loading real time network traffic dataset



**Fig 23.** Final results

**5 CONCLUSION AND FUTURE WORK**

In our project, we proposed a sequential neural network model where deep learning techniques were used and we achieved accuracy of 99.97%. Also, this detection was executed on real network traffic analysis having the knowledge of packet payload information. But, the 0.03% failure cases must be considered where the model failed to predict correct classes due to overfitting and large bias in the dataset. The misclassification of attacks is more compared to normal flow which is a problem to be researched and solved. To move further in this direction, we propose to evaluate more machine learning algorithms to detect minority attacks efficiently with acceptable false positive rate and low cost of misclassification in future.

Also, while we are evaluating our model on real time dataset, the capture of network traffic and loading it into Cicflowmeter needs to be done manually. There is lot of research going on in this field and our work was restricted to testing our model on the network traffic and classifying it. We propose to develop an alert system which will be able to generate an alarm whenever there is an attack in the network traffic. This can be done using IoT and Raspberry pi. Automation is the need of the hour and therefore, the capture, loading data, alert system needs to be automated without manual work.

This model can be deployed in organizations and this activity can be done by the network administrator. The future scope of this project is to deploy the model and alert system in real time environment having thousands of devices connected to same network and which could prevent the entry of intruders and thereby achieving Cyber security.

**6 REFERENCES**

[1] TY - BOOK,AU - Kanimozhi, V.,AU - Jacob, Prem, PY - 2019/04/01,SP - 0033,EP - 0036,T1 - Artificial Intelligence based Network Intrusion Detection with Hyper-Parameter Optimization Tuning on the Realistic Cyber Dataset CSE-CIC-IDS2018 using Cloud Computing, DO - 10.1109/ICCSP.2019.8698029

[2] Ali A. Ghorbani, Wei Lu and M. Tavallaee, Network intrusion detection and prevention: Concepts and Techniques, Advances in

Information security, Springer, 2010.

[3] man Sharafaldin, ArashHabibiLashkari, and Ali A. Ghorbani, “Toward

Generating a New Intrusion Detection Dataset andIntrusion Traffic

Characterization”, 4th International Conference on Information Systems

Security and Privacy (ICISSP), Portugal, January 2018.

[4] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," in IEEE Access, vol. 7, pp. 41525-41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[5] A. O. Adetunmbi, S.O. Falaki, O. S. Adewale, and B. K. Alese, Network Intrusion Detection based on rough set and k-nearest neighbour,

Intl. Journal of computing and ICT research, 2(1) (2008), 60-66.

[6] C. Elkan, Results of the KDD’99 classifier learning. SIGKDD Explorations, 1(2) (2000), 63-64.

[7] C. Krugel and T. Toth, Using decision tree to improve signature based intrusion detection, in: Proceedings of RAID, 2003, G. Vigna, E.

Jonsson, and C. Kruegel, eds, Lecture Notes in Computer Science, Vol. 2820, 173-191.

[8] D. Barbara, J. Couto, S. Jajodia, L. Popyack, and N. Wu, ADAM: Detecting intrusions by data mining, in : Proceedings of 2nd Annual IEEE workshop on Infor. Assu. Secur., Jun 2001, New York , 11-16.

[9] G. Wang, J. Hao, J. Ma and L. Huang, A new approach to intrusion detection using artificial neural networks and fuzzy clustering, Expert system with applications, 37 (2010), 6225-6232, Elsevier.

[10] NSL KDD dataset, http://iscx.ca/NSL-KDD/ , accessed on 8-10-2010.

[11] N. B. Amor, S. Benferhat and Z. Elouedi, Naïve Bayes Vs Decision Trees in Intrusion detection system,2004, 420-424

[12] Buczak, A.L.; Guven, E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutor. 2015, 18, 1153–1176.

[3] Phurivit Sangkatsanee, Naruemon Wattanapongsakorn and Chalermpol Charnsripinyo, “Real-time

Intrusion Detection and Classification”, IEEE network, 2009.

[4] Liberios Vokorokos, Alzebeta Kleniova, “Network Security on the Intrusion Detection System

Level”, IEEE network, 2004.

[5] Thomas Heyman, Bart De Win, Christophe Huygens, and Wouter Joosen, “Improving Intrusion

Detection through Alert Verification”, IEEE Transaction on Dependable and Secure Computing, 2004

[3] Phurivit Sangkatsanee, Naruemon Wattanapongsakorn and Chalermpol Charnsripinyo, “Real-time

Intrusion Detection and Classification”, IEEE network, 2009.

[4] Liberios Vokorokos, Alzebeta Kleniova, “Network Security on the Intrusion Detection System

Level”, IEEE network, 2004.

[5] Thomas Heyman, Bart De Win, Christophe Huygens, and Wouter Joosen, “Improving Intrusion

Detection through Alert Verification”, IEEE Transaction on Dependable and Secure Computing, 2004