**ATTACK DETECTION USING DEEP LEARNING IN VANET**

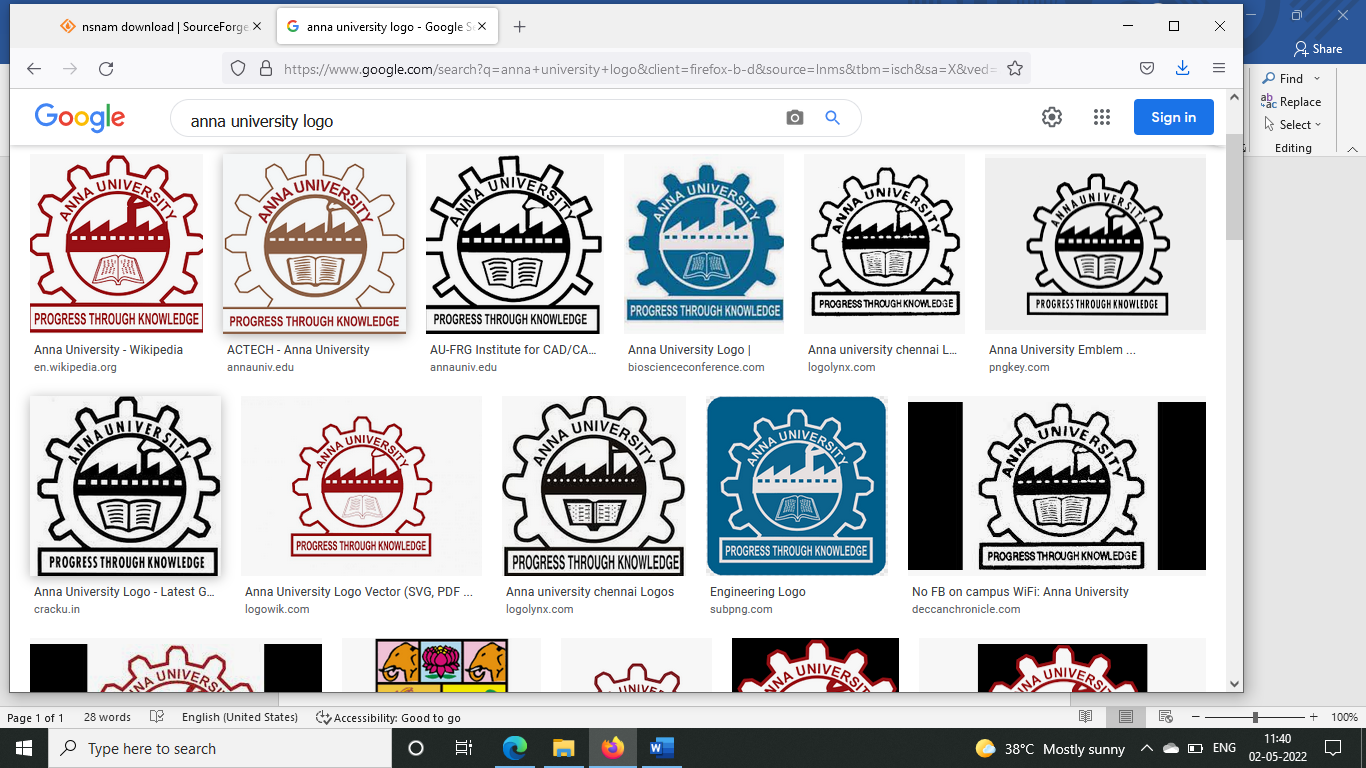
**PHASE II REPORT**

***Submitted by***

**P. SRILAKSHMI**

***In partial fulfilment for the award of the degree of***

**MASTER OF COMPUTER APPLICATION**



**DEPARTMENT OF INFORMATION SCENCE AND TECHNOLOGY**

**ANNA UNIVERSITY, CHENNAI**

**MAY, 2022**

**ANNA UNIVERSITY, CHENNAI**

**BONAFIDE CERTIFICATE**

Certified that this Report titled **ATTACK DETECTION USING DEEP LEARNING IN VANET** is the bonafide work of **P. SRILAKSHMI (2019272037)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. S. Sridhar Ms. G. Mahalakshmi

Professor and head Teaching Fellow

Department of information Department of information

Science and technology science and technology

Anna university Anna university

Chennai - 600 025 Chennai – 600 025

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE PAGE NO.**

**TABLE OF FIGURES 4**

**ABSTRACT 5**

1. **INTRODUCTION 6**
   1. MOTIVATION
   2. PROPOSED SYSTEM
   3. OBJECTIVE
2. **REVIEW OF LITERATURE 7**
   1. EXISTING / RELATED WORKS
3. **SYSTEM REQUOREMENT**

**SPECIFICATION 10**

3.1 REQUIREMENT ANALSIS

3.2 SYSTEM SPECIFICATION

1. **DESIGN CONCEPTS 11**
   1. DETAILED ARCHITECTURE

DIAGRAM

1. **IMPLEMENTATION 12**
   1. LIST OF MODULES
   2. ALGORITHM AND FLOW CHART
   3. MODULE BASED OUTPUT
2. **RESULT 33**
3. **CONCLUSION 34**
4. **REFERENCES 35**

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Fig no** | **Title** | **Page no** |
| 4.1 | Architecture diagram | 11 |
| 5.1.1 | Pre processing module | 12 |
| 5.1.2 | Feature selection module | 13 |
| 5.1.3 | Classification strategy | 13 |
| 5.1.4 | Attack generation | 14 |
| 5.2.1 | Flow chart | 15 |

**ABSTRACT:**

Vehicular Ad hoc Networks (VANETs) are the core of Intelligent Transportation Systems (ITS), allowing vehicles to communicate between themselves and with other entities. However, these are very complex networks with volatile architectures, ever-changing members, and multiple types of entities, making them an appealing target for attackers since they can find vulnerabilities and perform attacks with massive impact. The presence of a large number of communicating vehicles greatly increases the number and types of possible anomalies in the network. However, since there can be a multitude of anomalies possible in the network, there is a need for better anomaly detection frameworks that can address this unprecedented scenario. Due to these anomalies intrusion detection system comes into play. Intrusion detection systems are used to find the occurrence of intrusions and notifying it to the users. Therefore, it is very important to safe guard our data with the help of intrusion detection system. In this project, we propose an anomaly detection framework for VANETs based on deep neural networks (DNNs) using PCA and DNN algorithms. In order to find the attacks, we do feature selection on the data generated through VANETS. This feature selection process helps us to do effective detection which increases our packet delivery ratio. The DNN model learns the abstract and high-dimensional feature representation of the Intrusion Detection System data by passing them into many hidden layers. This also tells us that the false positive and false negative should be minimal.

**CHAPTER 1: INTRODUCTION**

* 1. **MOTIVATION**
* Intrusions occurs frequently in the networks. So, in order to identify and detect these intrusions this project is used.
* In this way, we can detect the intrusion early and save our data. The main objective is to identify the attacks and then classify them into different categories.
* It uses DNN to predict an incoming value as attack or not and classify them accordingly.
  1. **PROPOSED SYSTEM**

The project aims to detect and identify the different types of attacks that occur in the VANETs. We use Deep learning-based intrusion detection system. In this project we present an end-to-end DL-Based intrusion detection method for the VANET with only the given datasets and data gathered from the NS2 stimulations. We also evaluate our method on public datasets, experimental results show the capability of applying this in the VANET and the effectiveness of detecting malware traffic. This project uses DNN algorithm. They are structured in the form of interconnected nodes with an input layer, an output layer and one or more hidden layers. The model as various layers where input to each node is produced by applying some non-linear function on the data.

* 1. **MOTIVATION**

The aim of our study is to incorporate feature engineering to enhance the performance of Deep Learning techniques for attack classification. Due to the wide variety of attacks, and the unavoidable loopholes in traditional security methods, it is not possible to entirely prevent attacks on VANETs. This is done using the available feature selection and classification algorithm through which we get best results for the dataset. Numerical results will show that we can correctly detect data corresponding to several anomaly types.

**CHAPTER 2: LITERATURE SURVEY**

**2.1 PROPOSED / EXISTING WORKS:**

1. **Deep Learning Approach for Intelligent Intrusion Detection System:**

Authors: R. Vinayakumar, Mamoun Alazab, K. P. Soman, Prabaharan

Poornachandran , Ameer Al-Nemrat , And Sitalakshmi Venkatraman.

Paper published in: IEEE Access (Volume: 7) Digital Object Identifier 10.1109/ACCESS.2019.2895334.

**Concept in the paper**:

In this paper, a hybrid intrusion detection alert system is proposed which has the capability to analyze the network and host-level activities. It uses DNNs for handling and analyzing very large-scale data in real-time. The proposed architecture is able to perform better than previously implemented classical machine learning classifiers in both HIDS and NIDS. The execution time of the proposed system can be enhanced by adding more nodes to the existing cluster. In addition, the proposed system does not give detailed information on the structure and characteristics of the malware.

1. **DeepADV: A Deep Neural Network Framework for Anomaly Detection in VANETs:**

Authors: Tejasvi Alladi,Bhavya Gera, Ayush Agrawal, Vinay Chamola and Fei Richard Yu

Paper published in: IEEE TRANSACTIONS ON VEHICULAR TECHNOLO-GY, VOL. 70, NO. 11, NOVEMBER 2021

**Concept in the paper:**

A DNN-based anomaly detection framework for the identification of anomalous data pertaining to various possible anomalies in VANETs. Anomaly scenarios are considered for analyzing the performance of the proposed framework. Performance analysis of the framework using key evaluation metrics is presented by employing several DNN architectures.

1. **Malicious Node Detection in Vehicular Ad-Hoc Network Using Machine Learning and Deep Learning:**

Authors: Elvin Eziama , Kemal Tepe , Ali Balador , Kenneth Sorle Nwizege and Luz M. S. Jaimes

Paper published in: 2018 IEEE Globecom Workshops (GC Wkshps)

**Concept in the paper**:

The paper proposes the Bayesian Neural Network (BNN) model framework for high performance prediction, classification accuracy and low detection latency, in trust computation in VANET. The idea of BNN framework is needed to overcome the challenges in NN. The future work aims at implementing the proposed framework in Veins simulator and providing simulation experiments.

1. **Deep Neural Networks for Securing IoT Enabled Vehicular Ad-Hoc Networks**

Authors: Tejasvi Alladi, Ayush Agrawal , Bhavya Gera, Vinay Chamola, Biplab Sikdar, Mohsen Guizani

Paper published in: CC 2021 – IEEE International Conference on Communications.

**Concept in the paper**:

In this paper, a deep neural network architecture for securing IoT-enabled VANETs is proposed. Deep learning models were trained on time sequences generated from normal vehicle data in the network. The paper talks about classify sequences into normal data and anomalies and perfectly classify most of the anomalous types. The proposed CNN & LSTM model is shown to give a higher performance compared to a stacked LSTM model.

1. **DeepVCM: a Deep Learning Based Intrusion Detection Method in VANET:**

Authors: Yi Zeng, Meikang Qiu, Dan Zhu, Zhihao Xue, Jian Xiong, and Meiqin Liu

Paper published in: IEEE Intl Conference on Intelligent Data and Security (IDS)

**Concept in the paper**:

The DeepVCM procedure is consist of two common DL models, namely CNN and LSTM. This method does not require the heavy work of selecting features and private featured details. Moreover,it can attain a much more robust and accurate performance on intrusion detection than state-of-art methods with a less storage resource requirement.

**CHAPTRER 3: SYSTEM REQUIREMENT SPECIFICATION**

**3.1 REQUIREMENT ANALYSIS**

Language used:

* Python
* tcl

Platform used:

* Jupyter notebook
* NS-2.35 stimulator

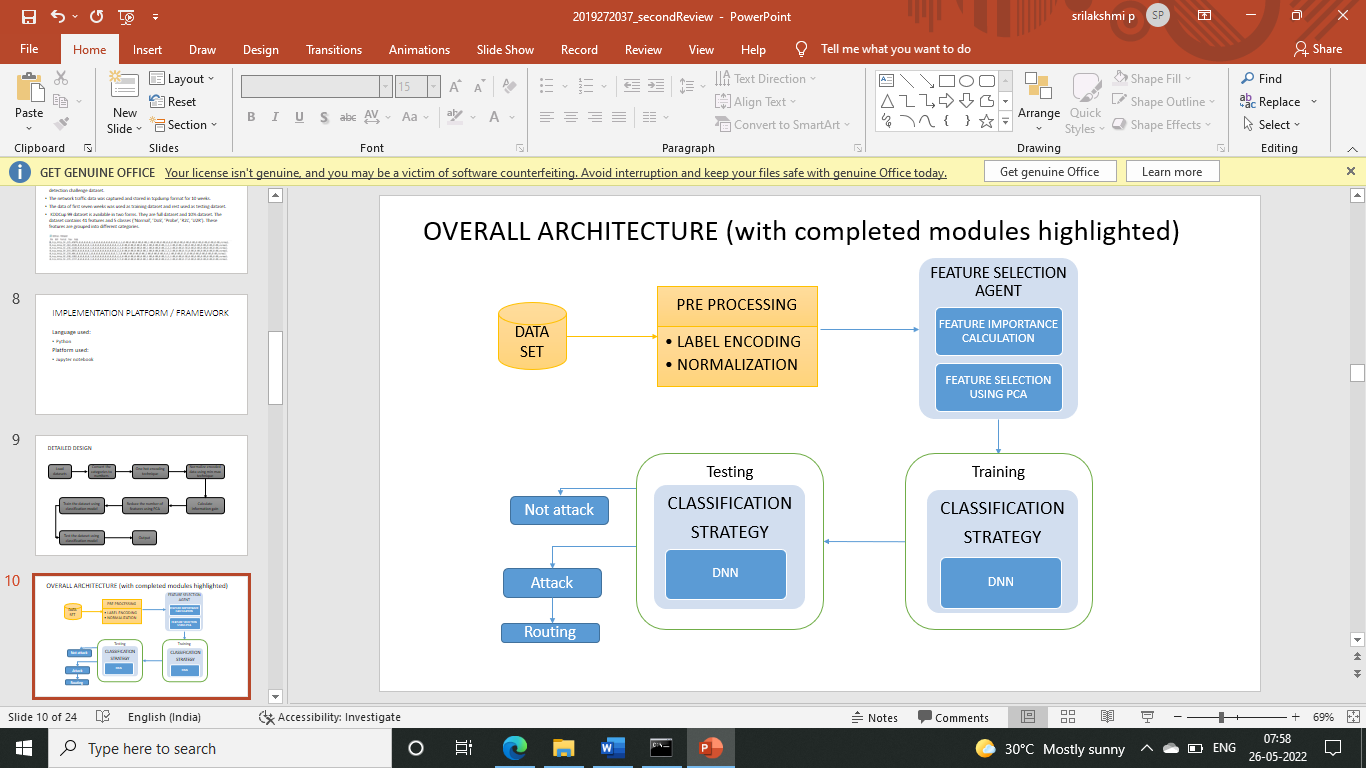
**3.2 SYSTEM SPECIFICATION**

Hardware specification:

* Processor: standard processor with a speed of 1.6 GHz or more
* RAM: 4GB
* Windows 10 operating system (64-bit)

**CHAPTER 4: DESIGN CONCEPTS**

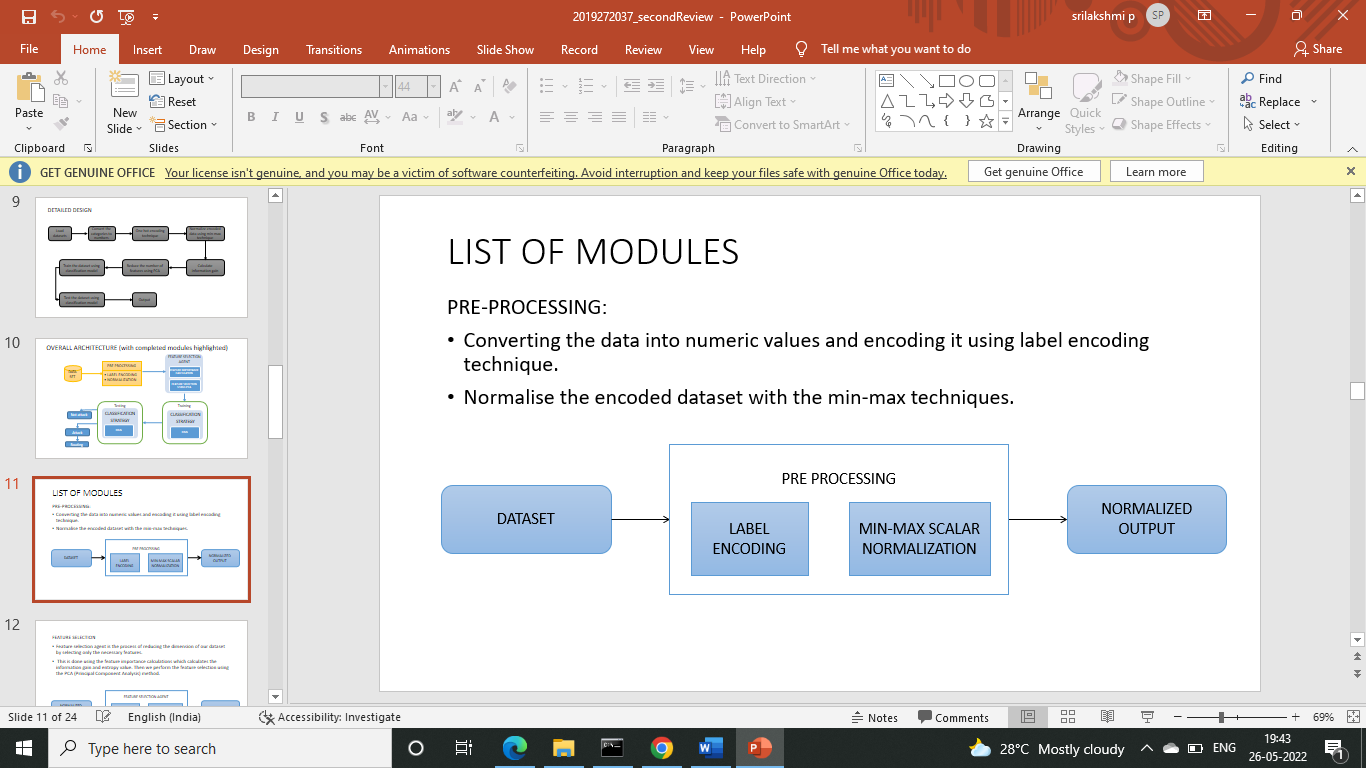
**4.1 DETAILED ARCHITECTURE DIAGRAM:**



**CHAPTER 5: IMPLEMENTATION**

* 1. **LIST OF MODULES**
     1. **Pre-processing**

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn’t specialized for any specific problems. Thus, the process of finding the attacks is done with the help of python. In this pre-processing module the data is converted into numeric values and encoding it using label encoding technique. Normalise the encoded dataset with the min-max techniques.

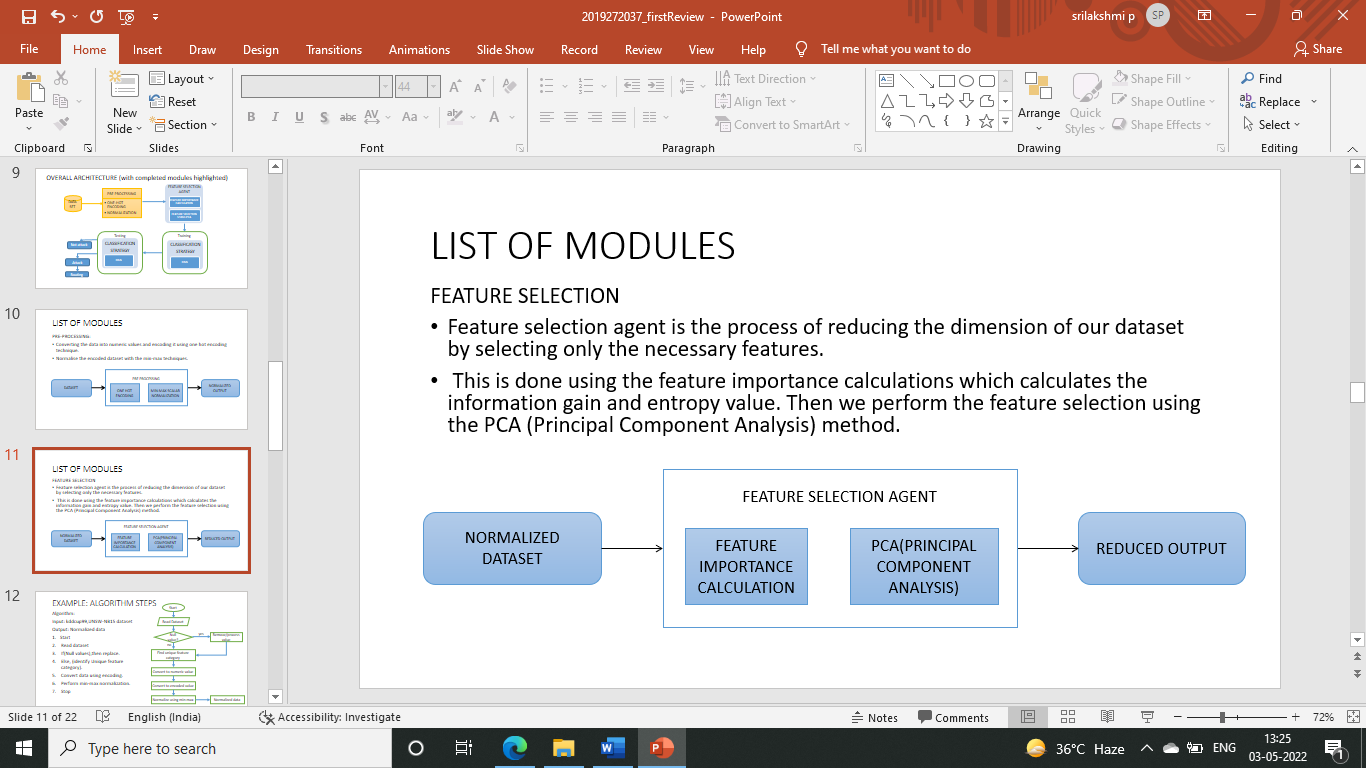


* + 1. **Feature selection:**

Feature selection agent is the process of reducing the dimension of our dataset by selecting only the necessary features. This is done using the feature importance calculations which calculates the information gain and entropy value. The entropy is calculated using the formula

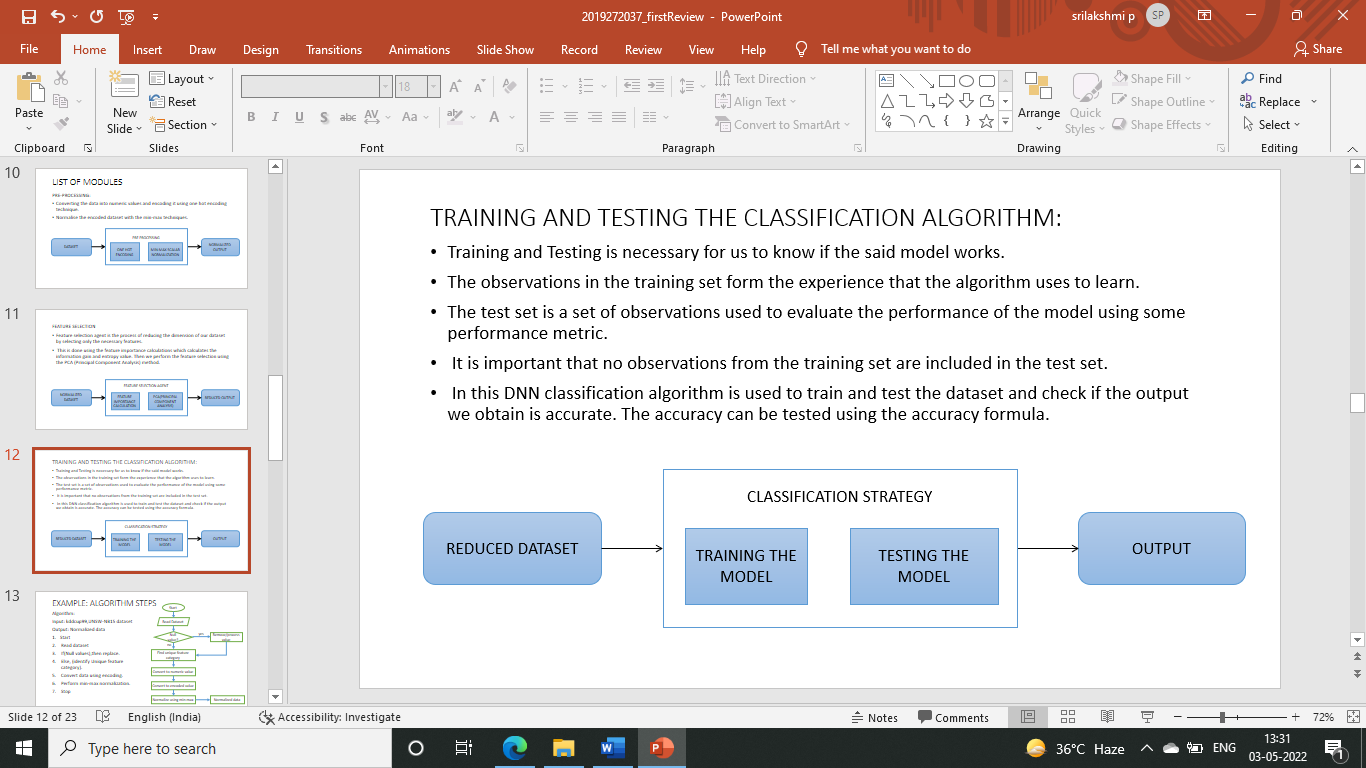
Entropy = -(p(0) \* log(P(0)) + p(1) \* log(P(1)))

Then we calculate the mutual information gain using the mutual\_info\_classif() method. The we perform the feature selection using the PCA (Principal Component analysis) method.



* + 1. **Training and testing the classification Algorithm:**

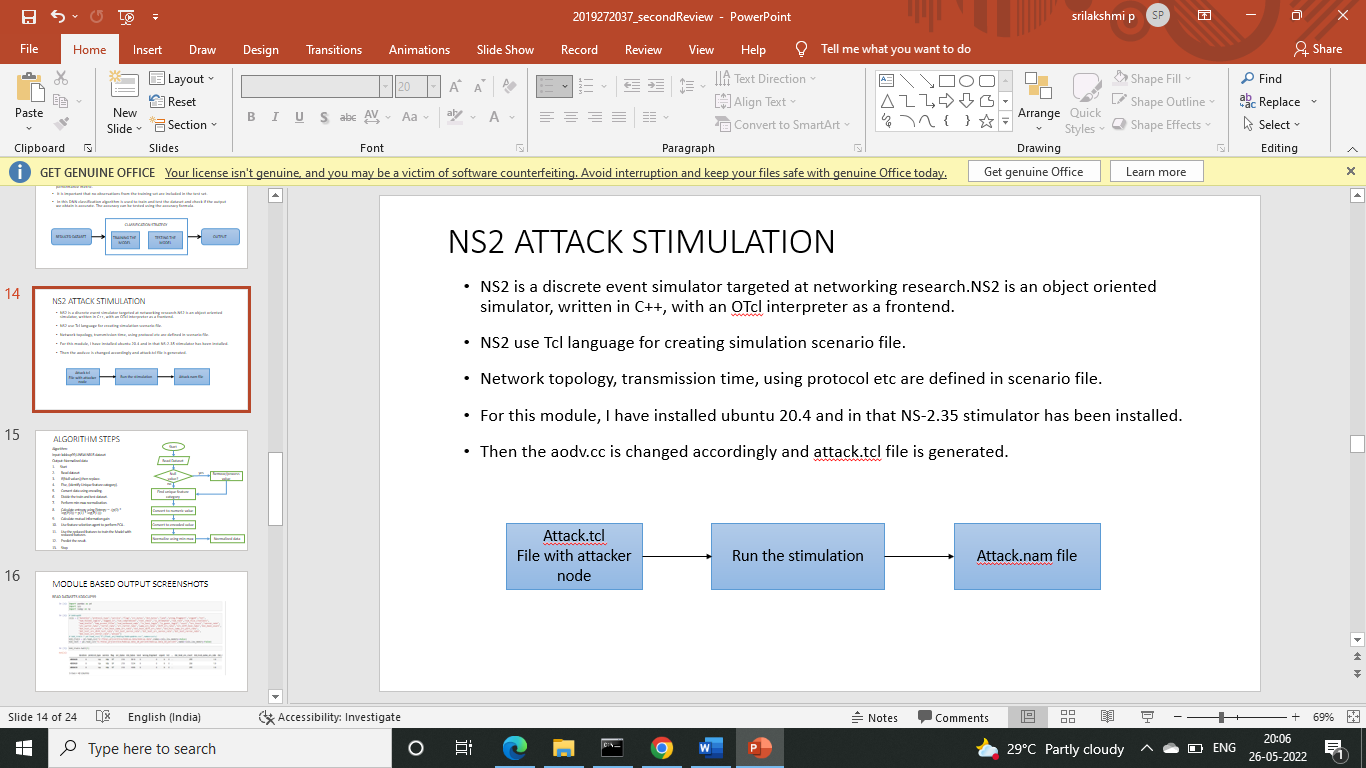
Training and Testing is necessary for us to know if the said model works. The observations in the training set form the experience that the algorithm uses to learn. The test set is a set of observations used to evaluate the performance of the model using some performance metric. It is important that no observations from the training set are included in the test set. In this DNN classification algorithm is used to train and test the dataset and check if the output we obtain is accurate. In this module, RELU and softmax activations are used. The accuracy can be tested using the accuracy formula.



* + 1. **NS2 blackhole attack generation:**

NS2 is a discrete event simulator targeted at networking research.NS2 is an object oriented simulator, written in C++, with an OTcl interpreter as a frontend. NS2 use Tcl language for creating simulation scenario file.

Network topology, transmission time, using protocol etc are defined in scenario file. For this module, I have installed ubuntu 20.4 and in that NS-2.35 stimulator has been installed. Then the aodv.cc is changed accordingly and attack.tcl file is generated.



* 1. **ALGORITHM AND FLOW DIAGRAM:**

Algorithm:

Input: UNSW-NB15 dataset

Output: Normalized data

1. Start
2. Read dataset
3. If (Null values), then replace.
4. Else, (identify Unique feature category).
5. Convert data using label encoding.
6. Divide the train and test dataset.
7. Perform min-max normalization.
8. Calculate entropy using Entropy = -(p (0) \* log2(P (0)) + p(1) \* log2(P(1)))
9. Calculate mutual information gain. (information gain)
10. Use feature selection agent to perform PCA.
11. Use the reduced features to train the Model with reduced features.
12. Predict the result.
13. Stop.

Flow diagram:

Read Dataset

Null values?

Remove/Process the value

yes

no

Find unique feature category

Convert to numeric value

Divide train and test set

Find entropy value

Calculate information gain

Feature selection using PCA

Train using DNN

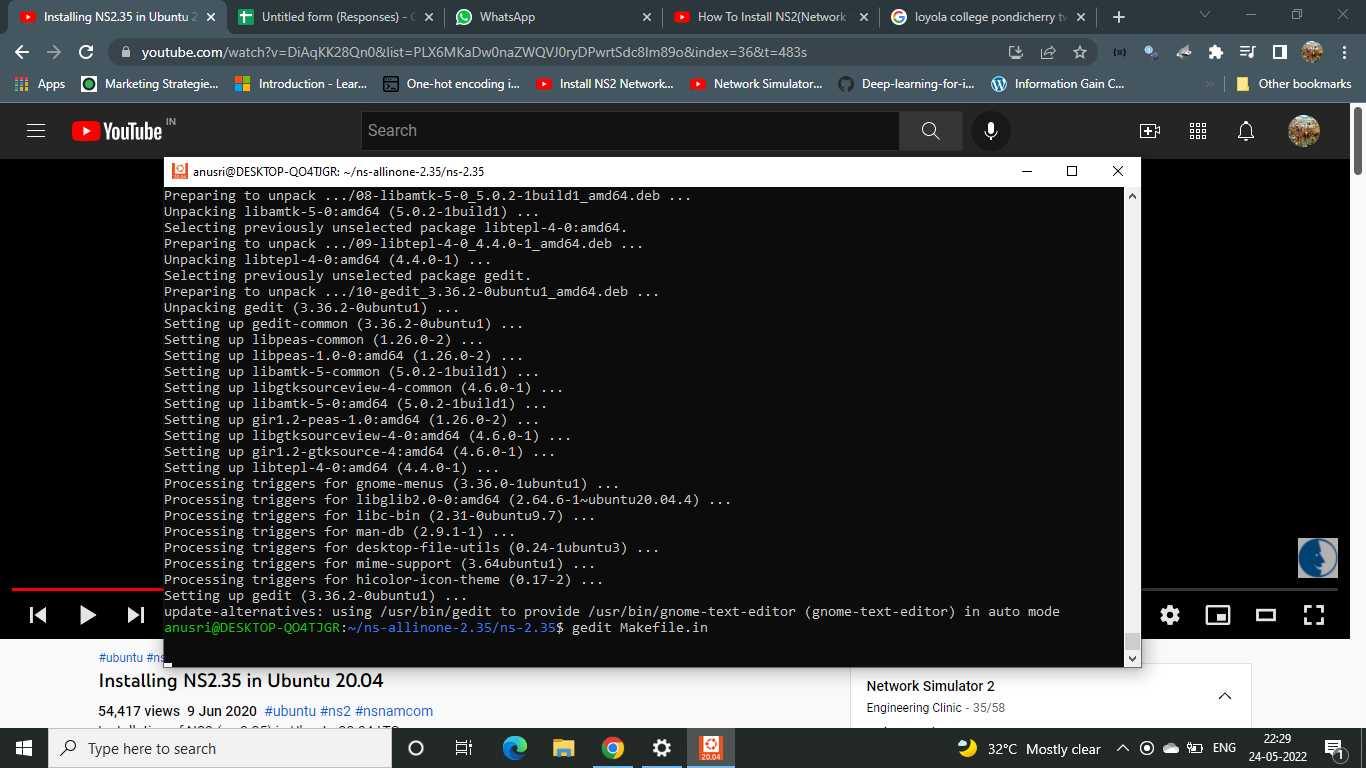
Predict result using testing data

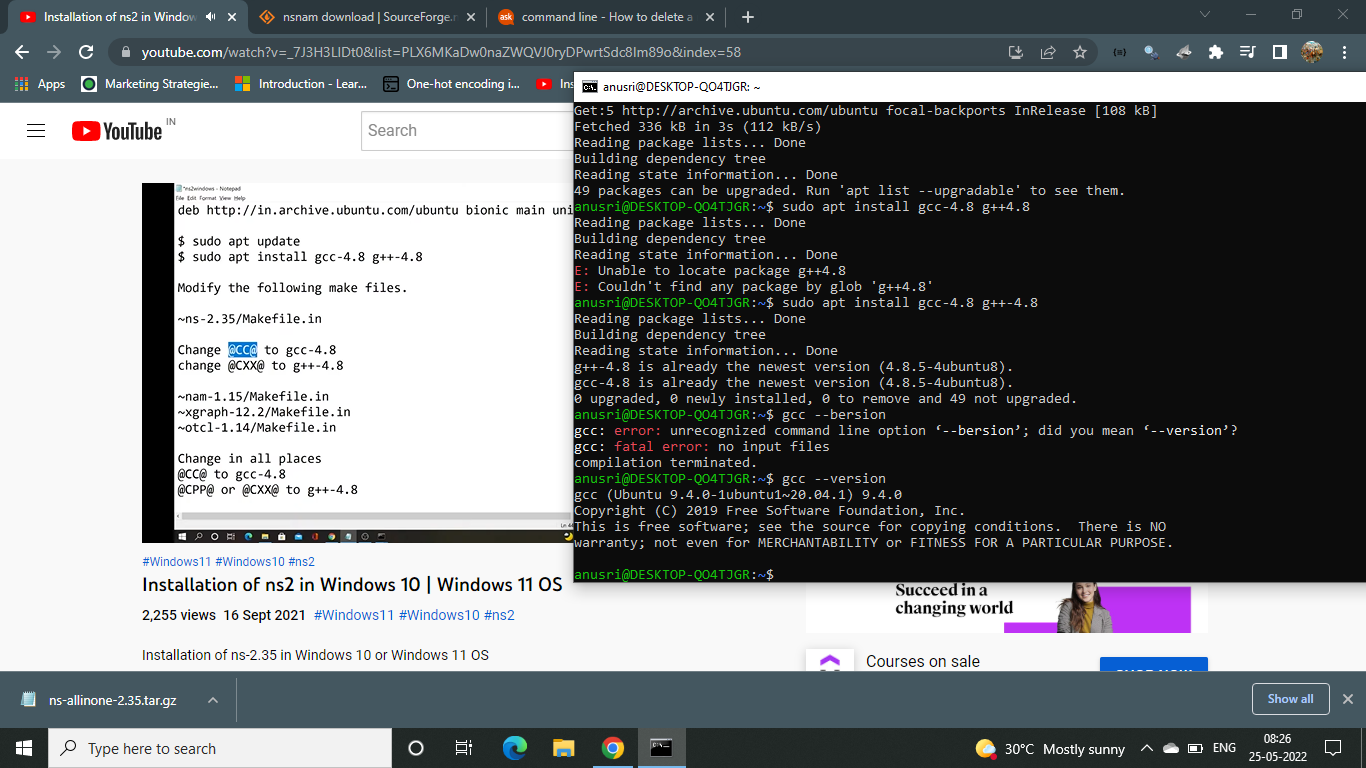
Find Accuracy and score

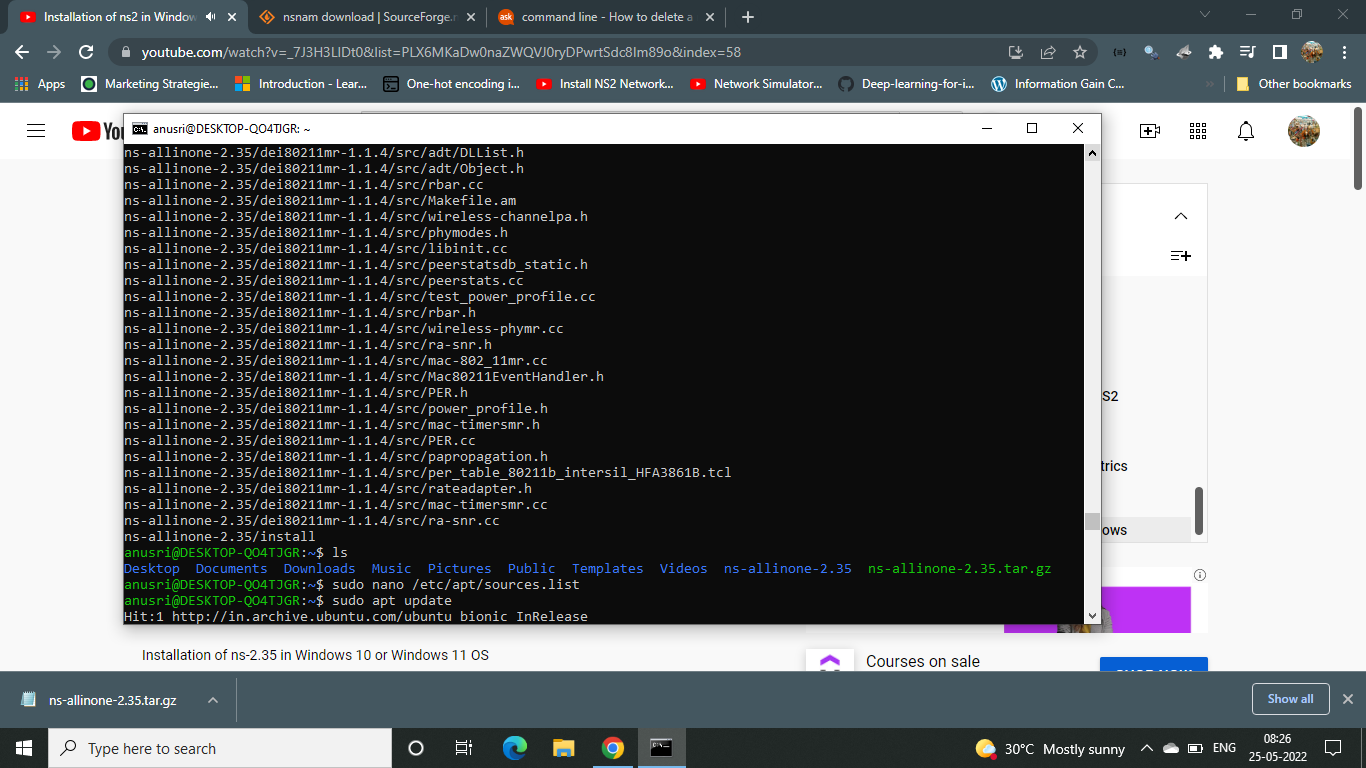
* 1. **MODULE BASED OUTPUT:**

**GENERATING ATTACKS:**

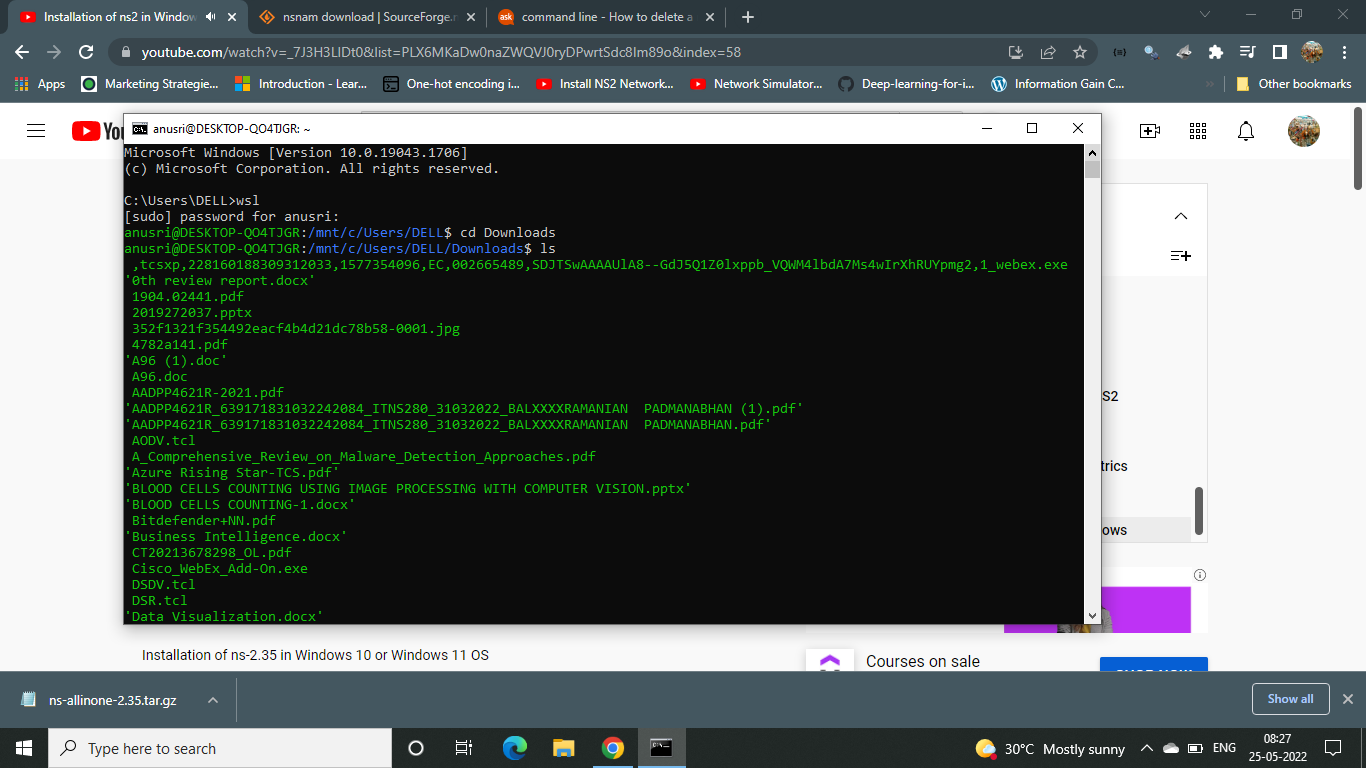
Installing ubuntu: First we will download the ubuntu from Microsoft store and also install wsl.Then make changes in makefile.in and install.

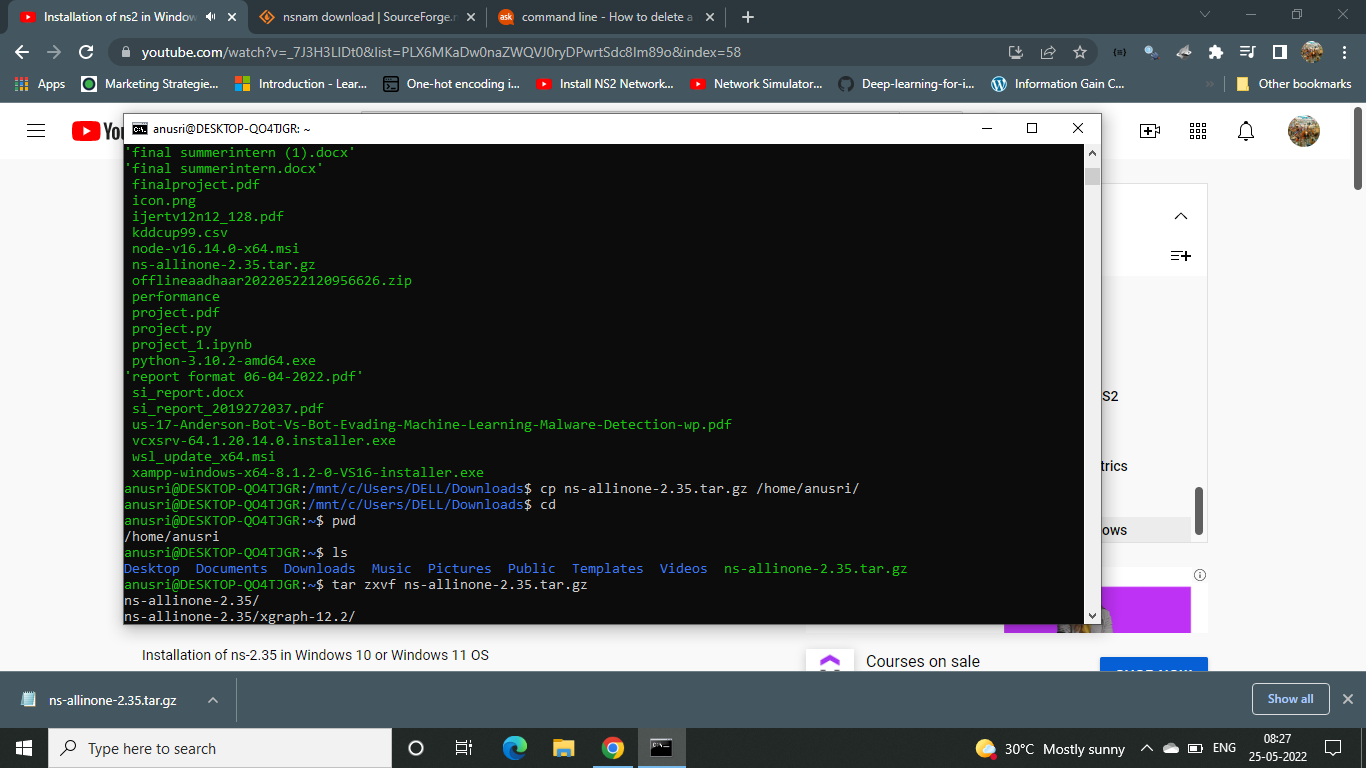


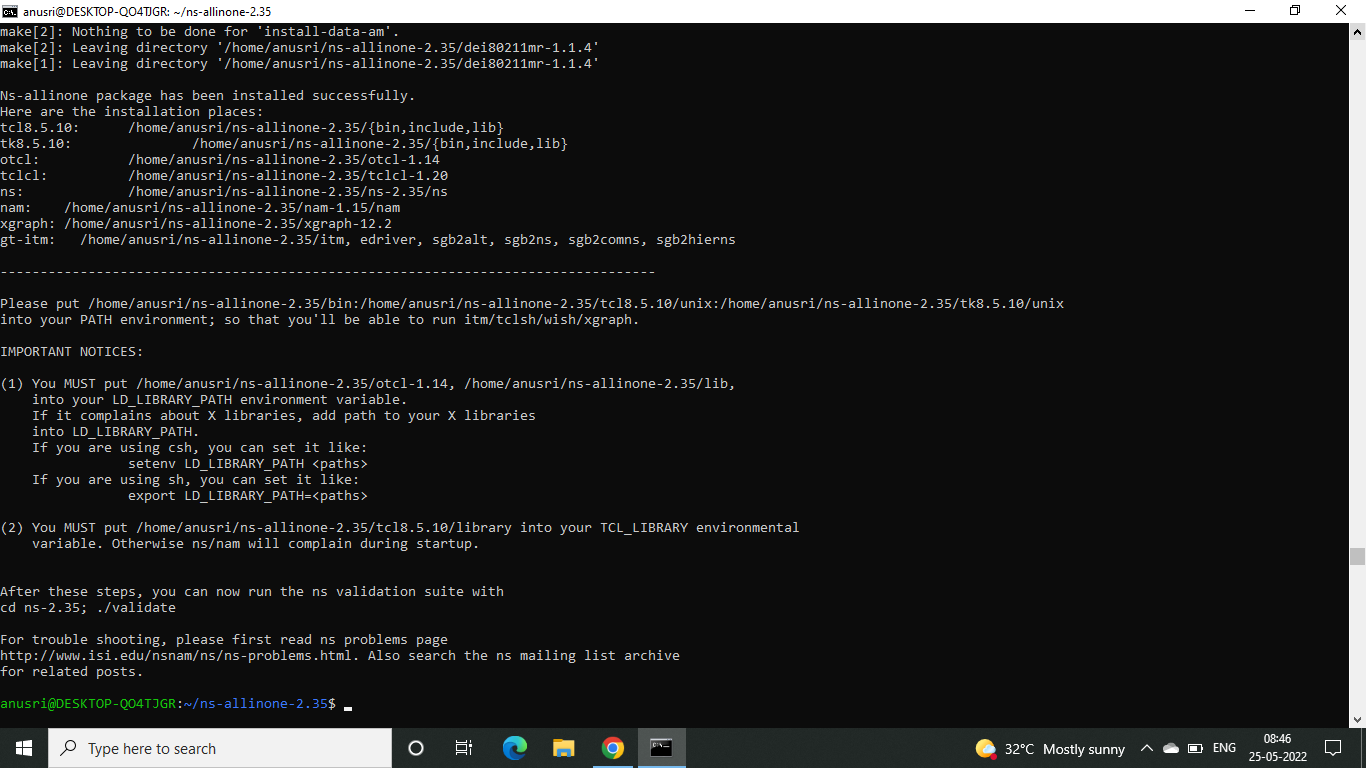
****

****

Unzip the downloaded NS2 stimulator folder and install them

****

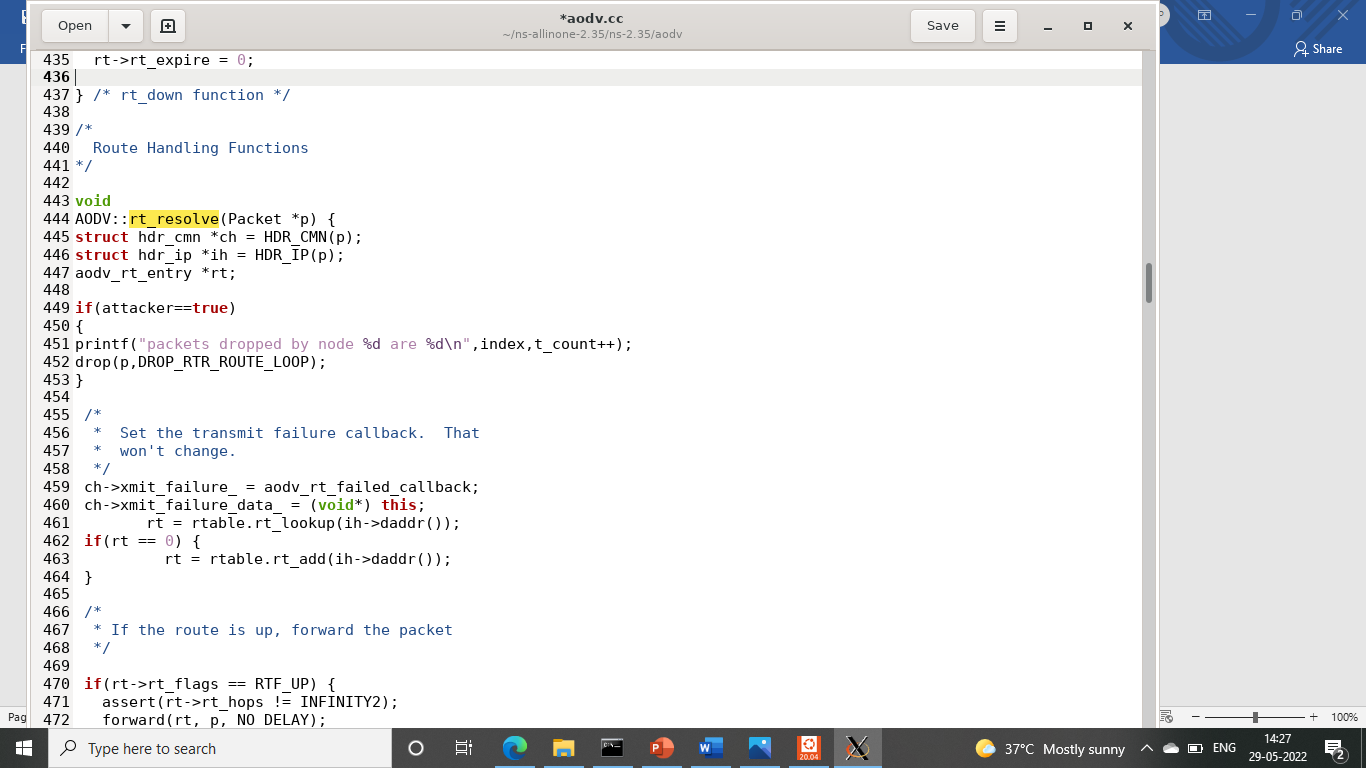
****

****

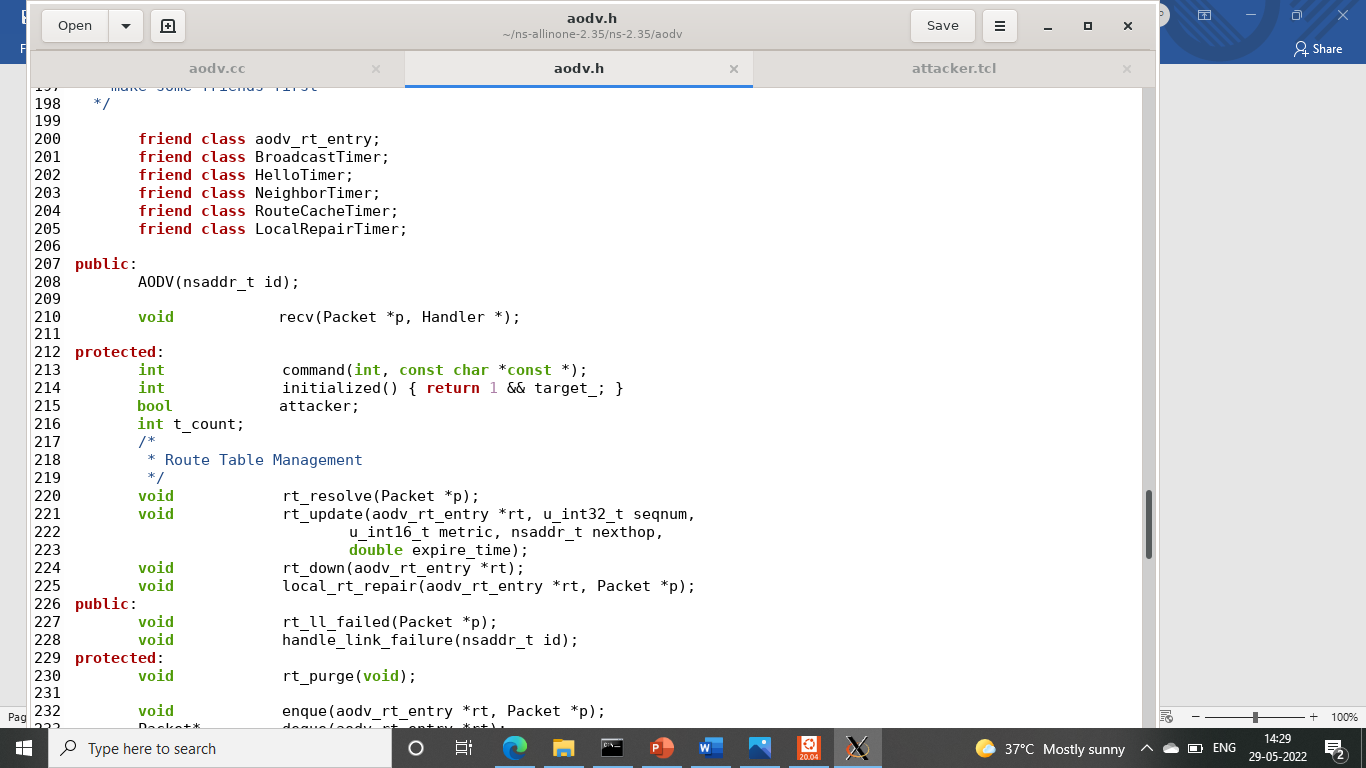
After installing open the AODV.cc file and AODV.h file to make changes

Adov.cc

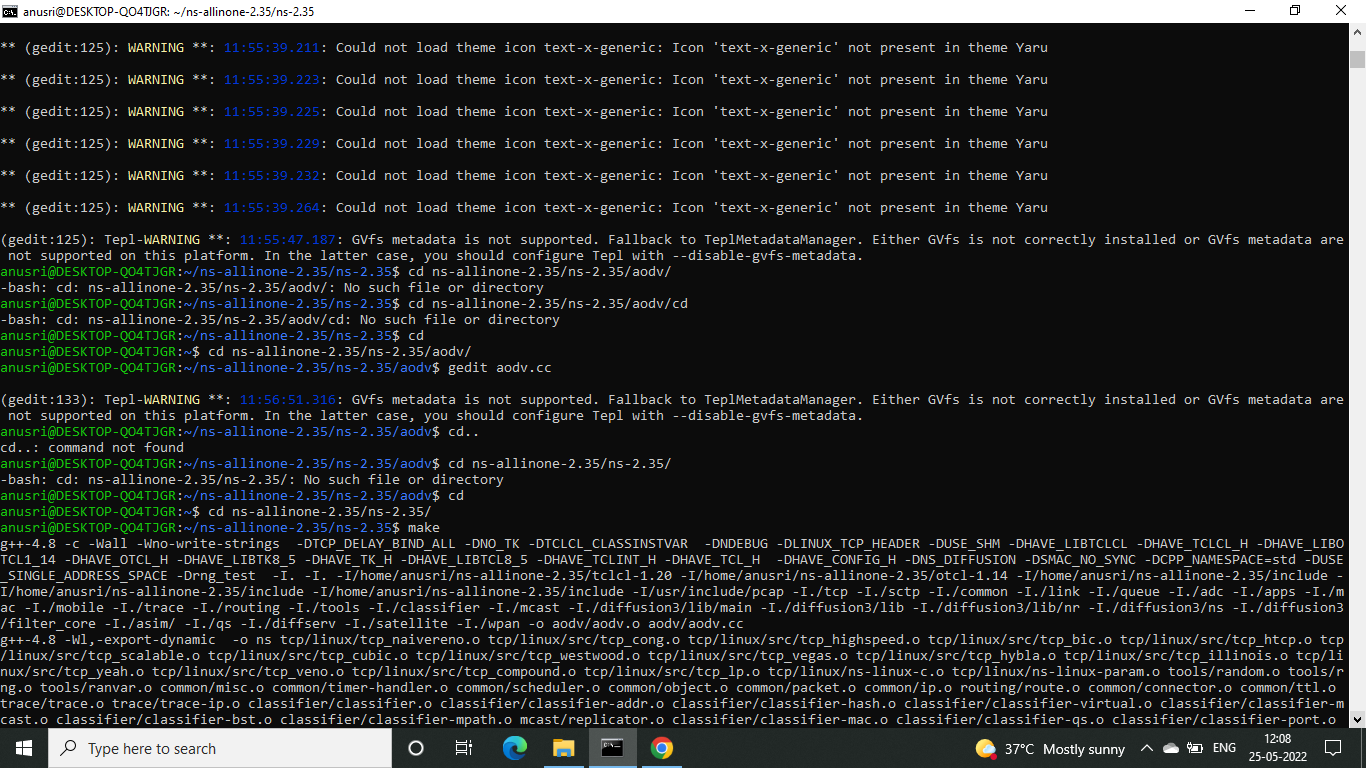




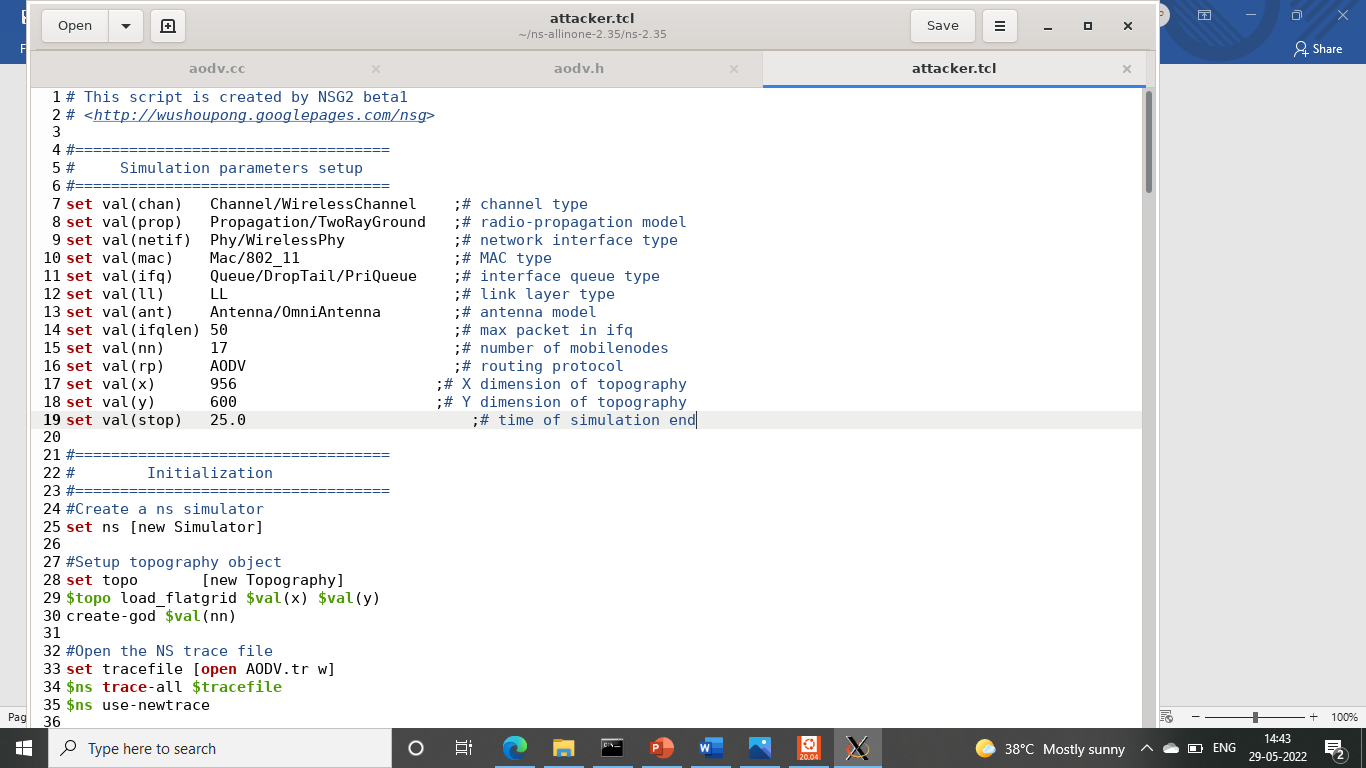
Adov.h



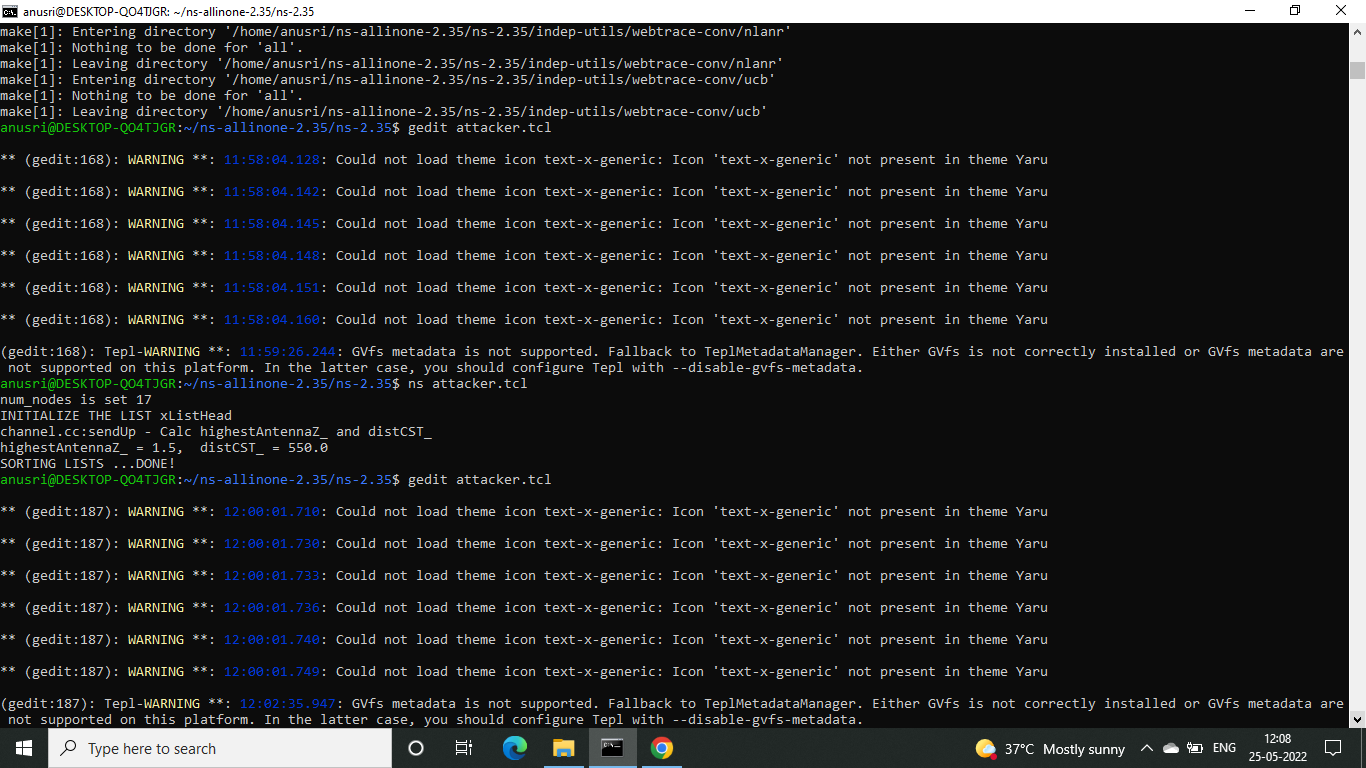
Now we build the codes.

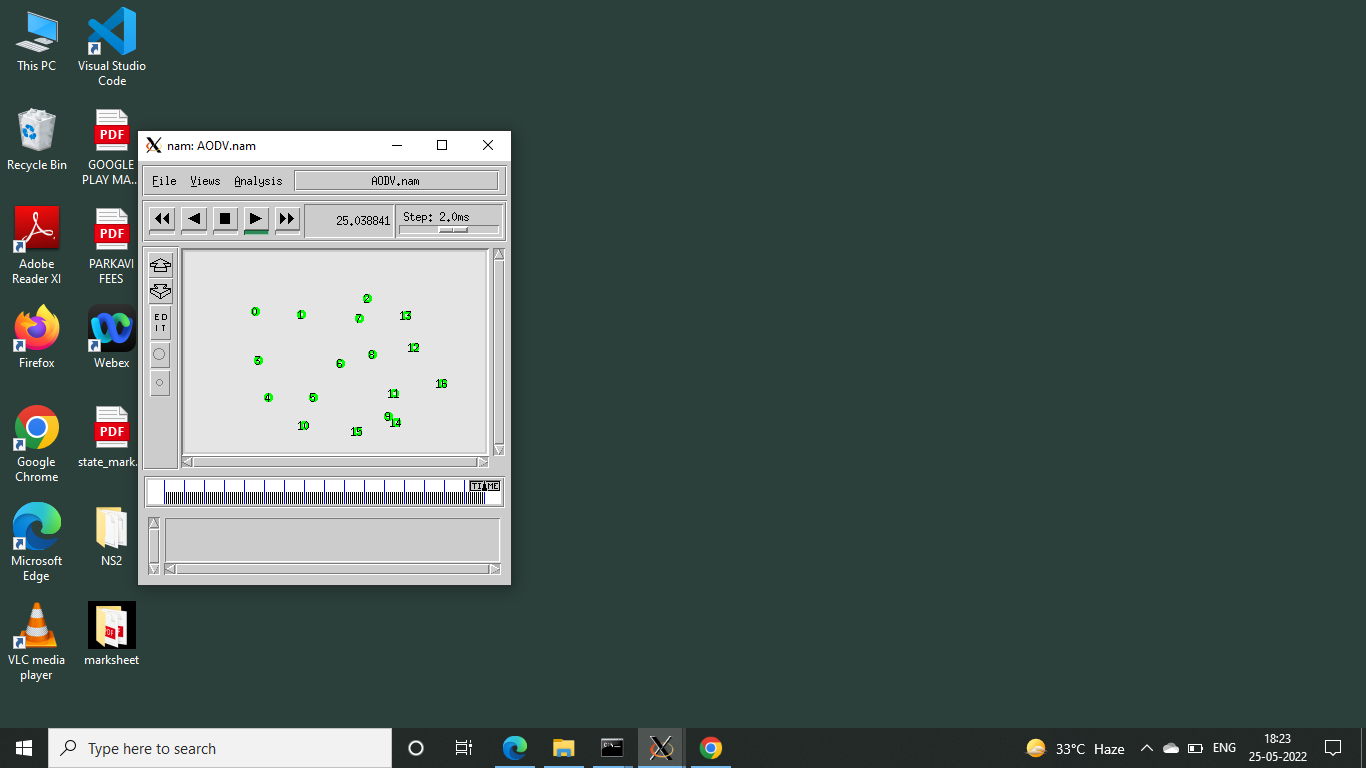
****

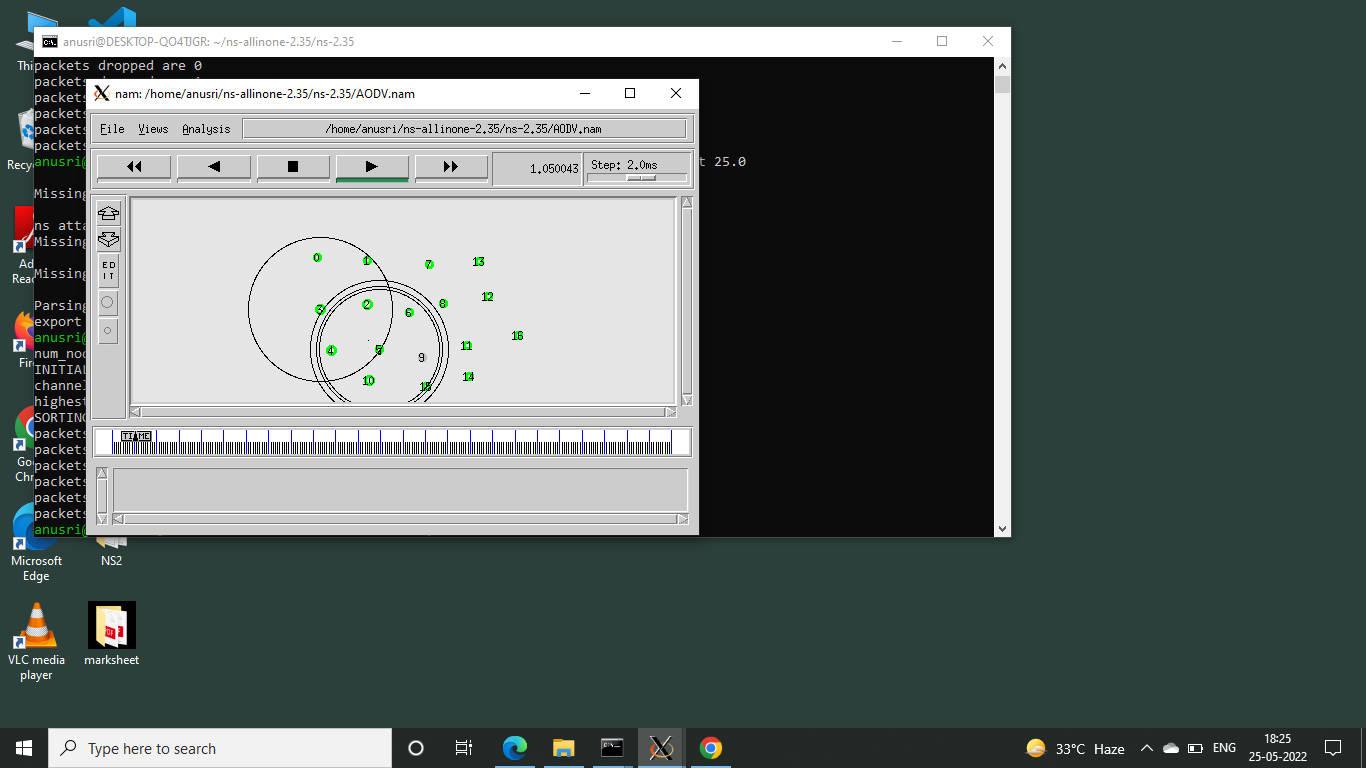
Now create a tcl code with no attacks:



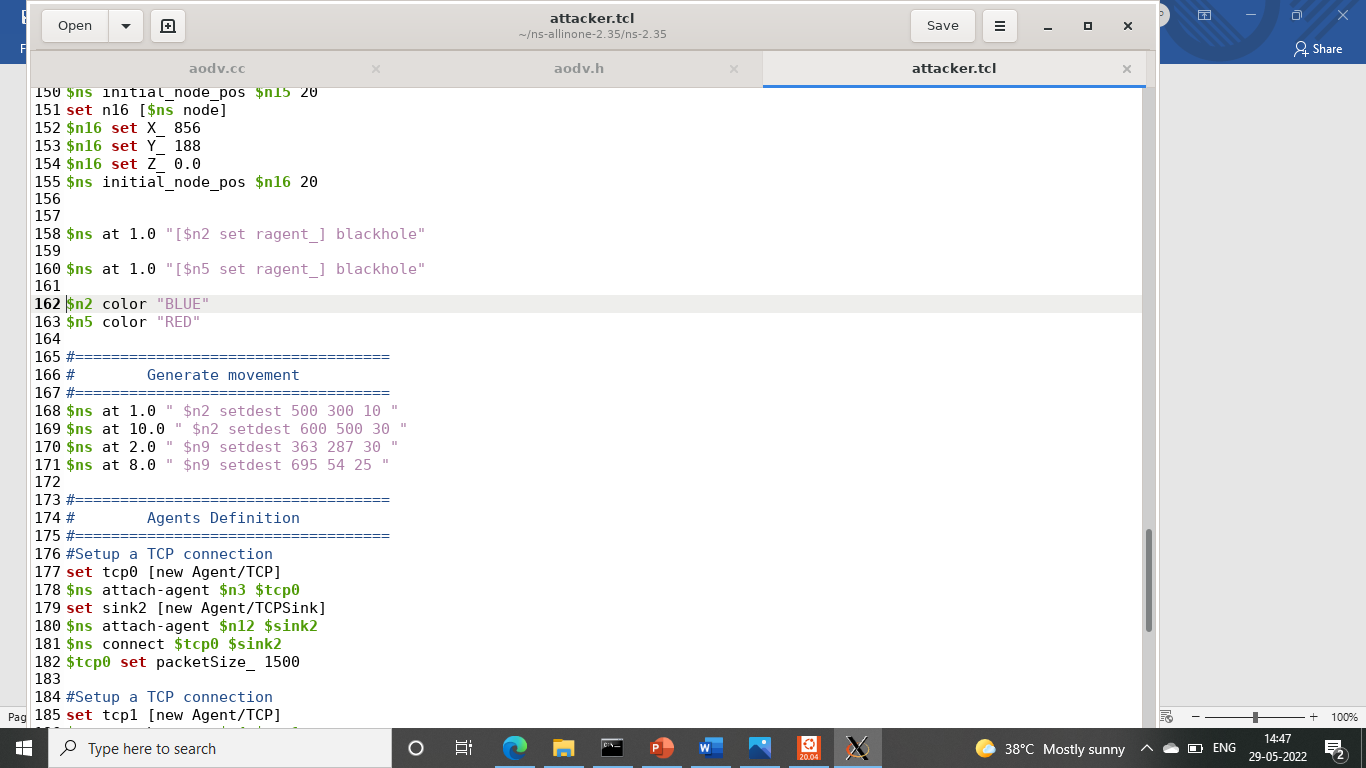
Run the code. In this since there is no attacker node no packets are dropped

****

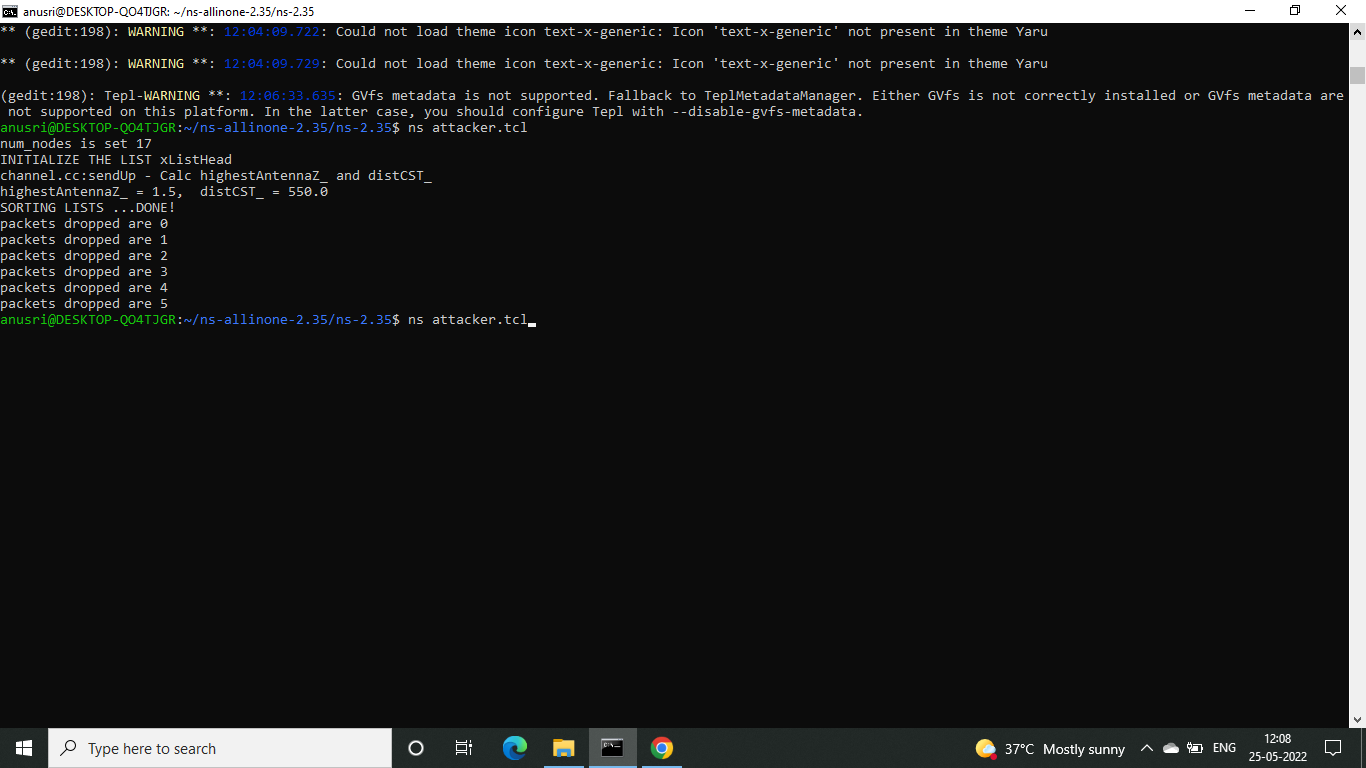
****

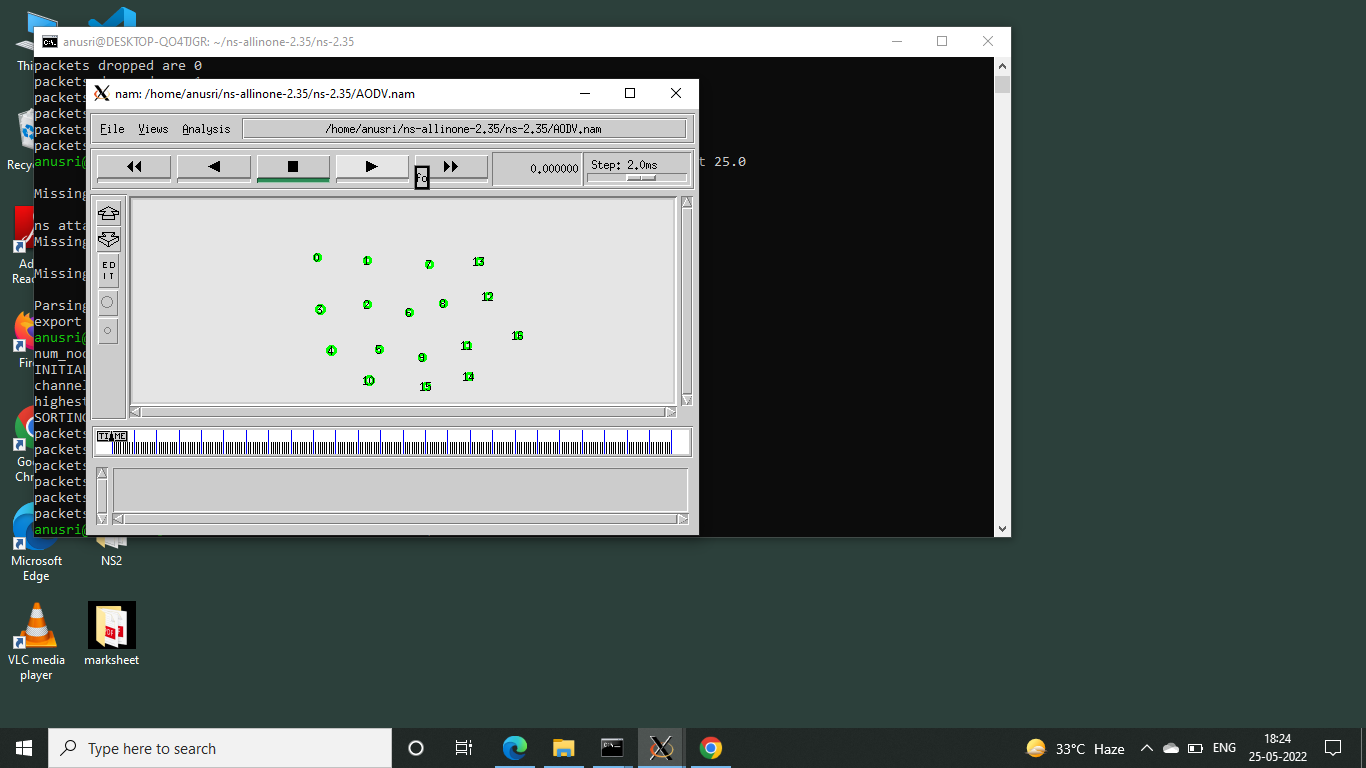
****

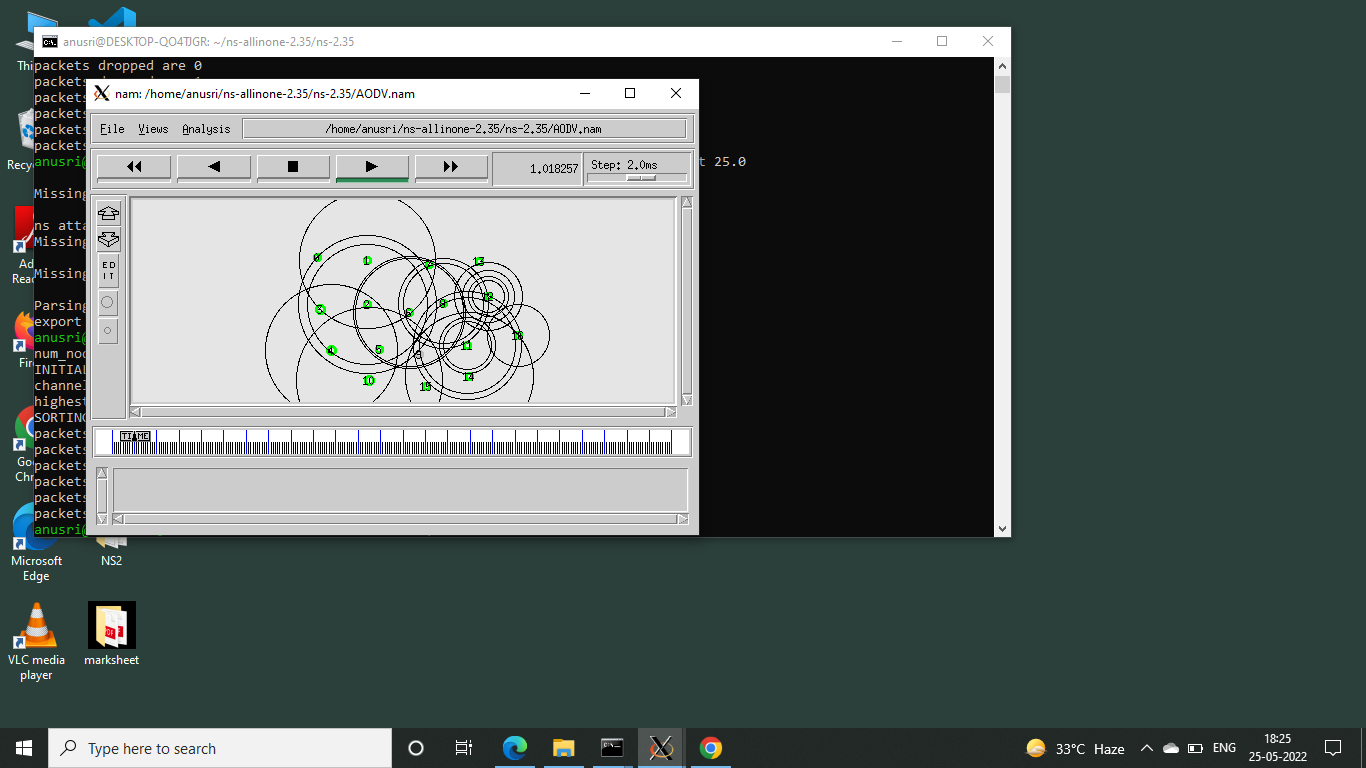
Build a code with attacker node:



Attack is generated and packets are dropped

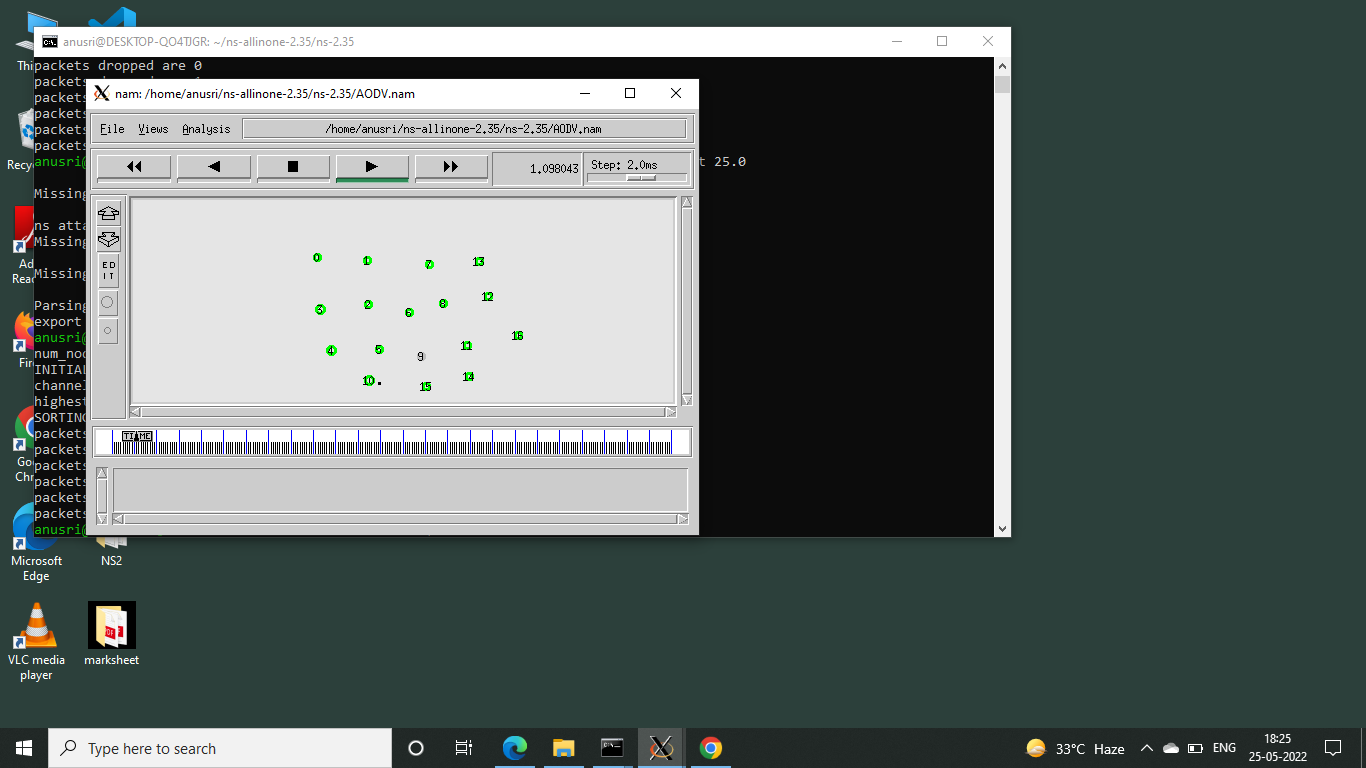
****

****

****

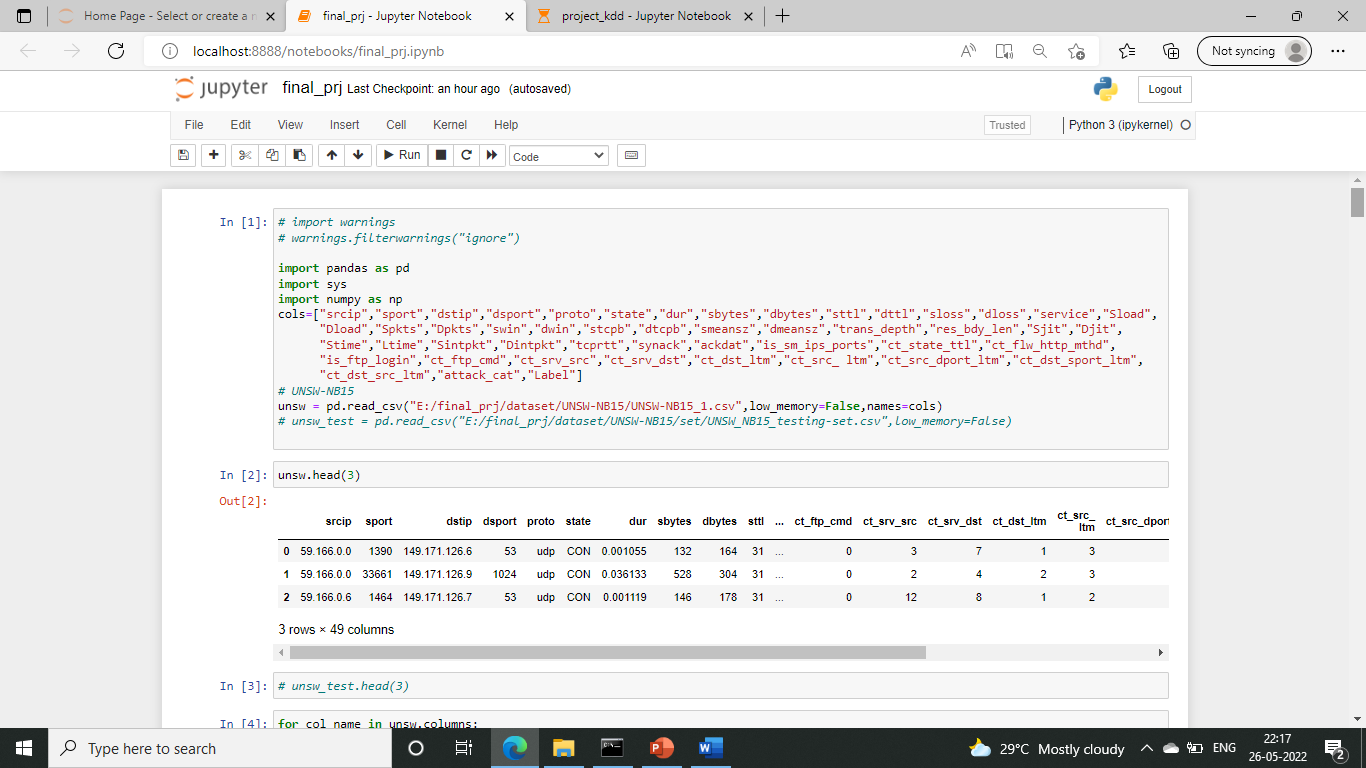
Here we can see node 5 dropping a packet

****

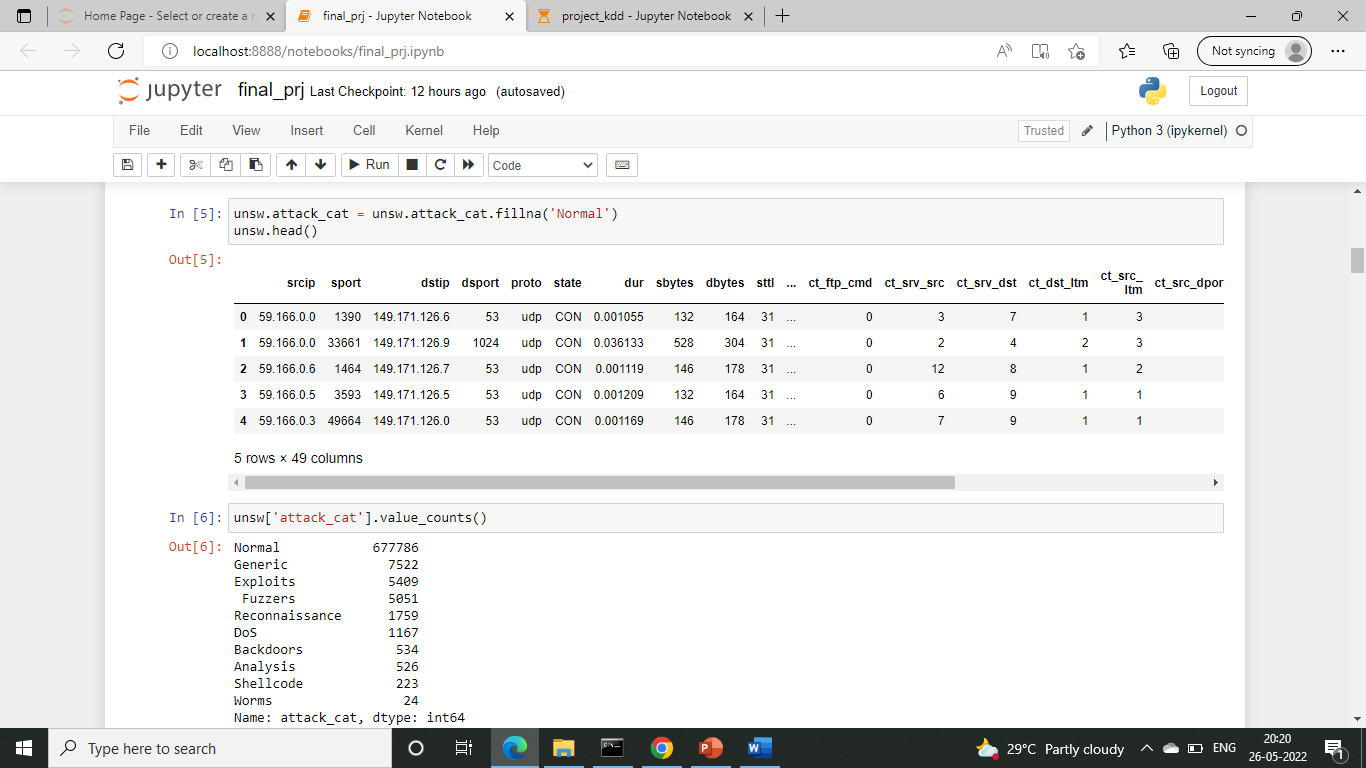
****

**PRE-PROCESSING:**

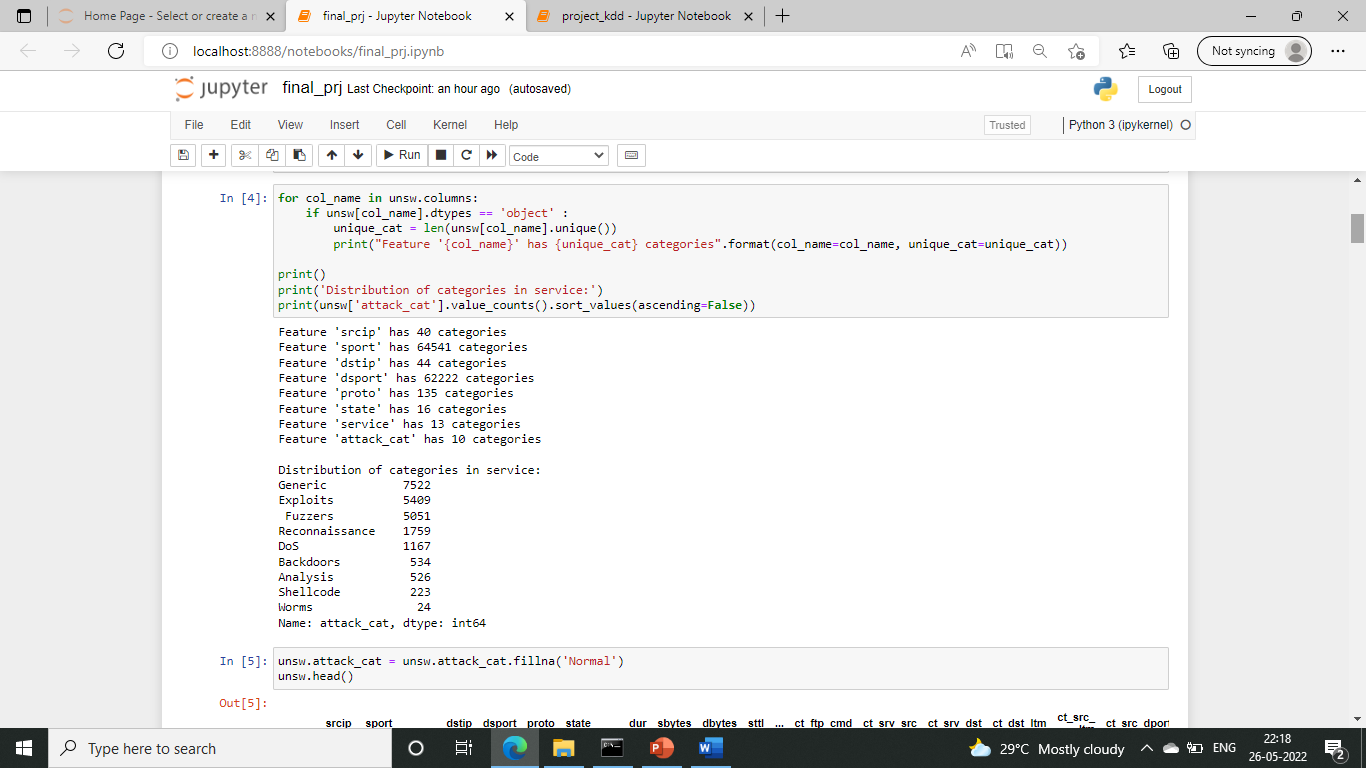
Insert dataset



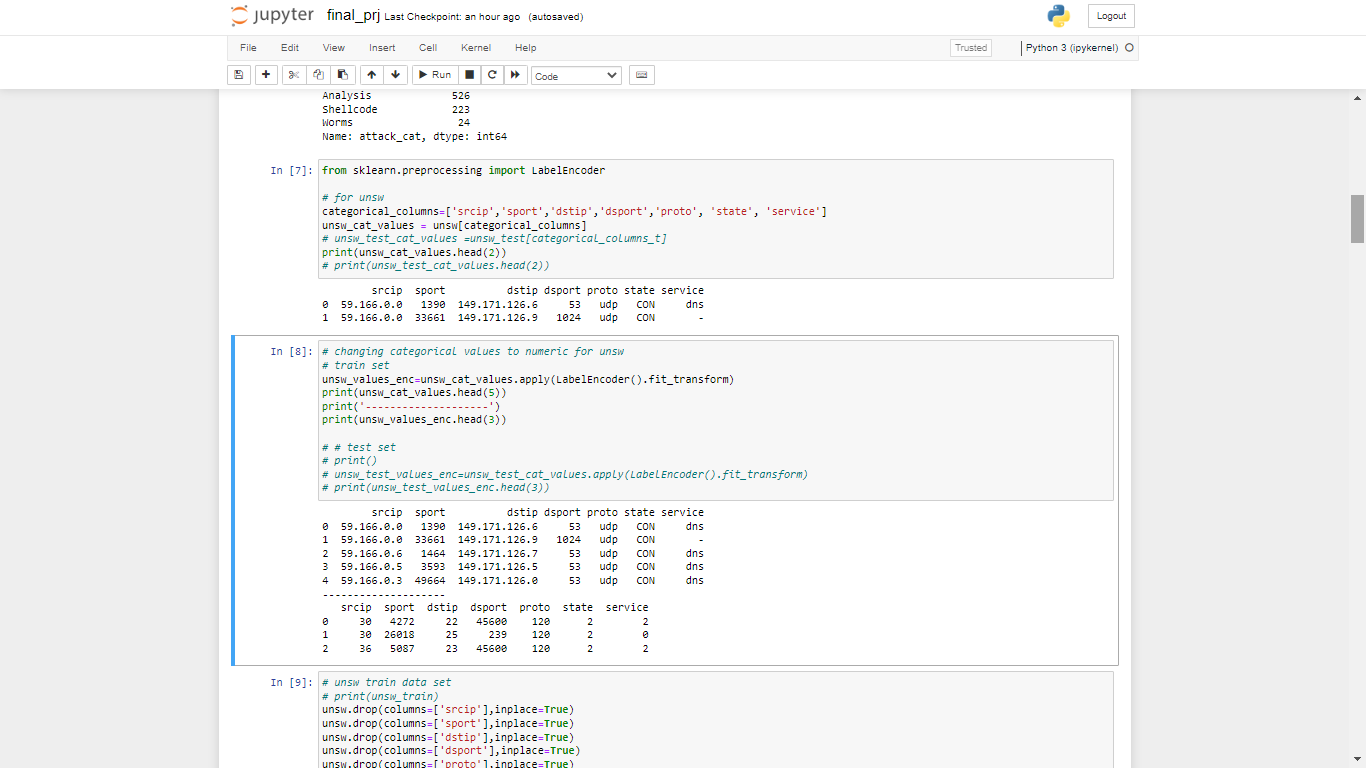
Checking for null values and replacing:

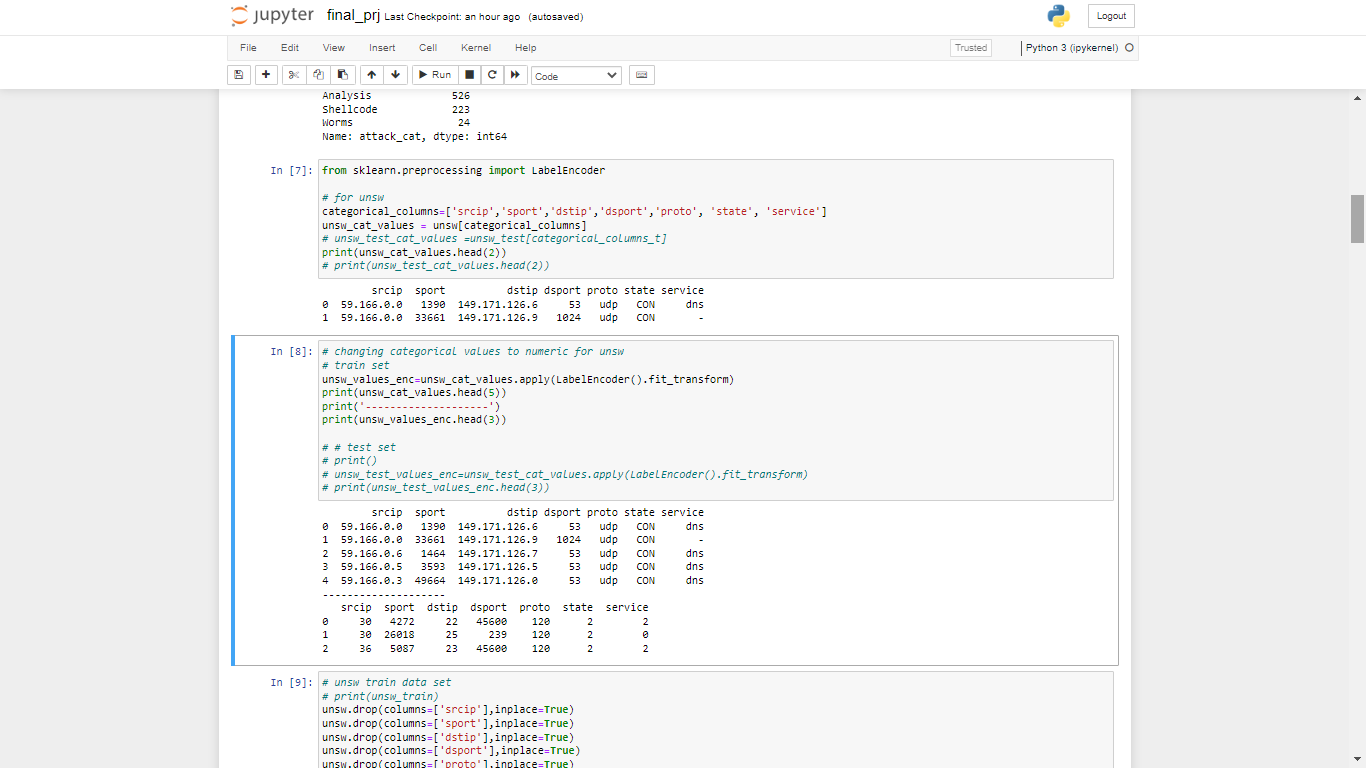


Identify Unique feature category for UNSW-NB15

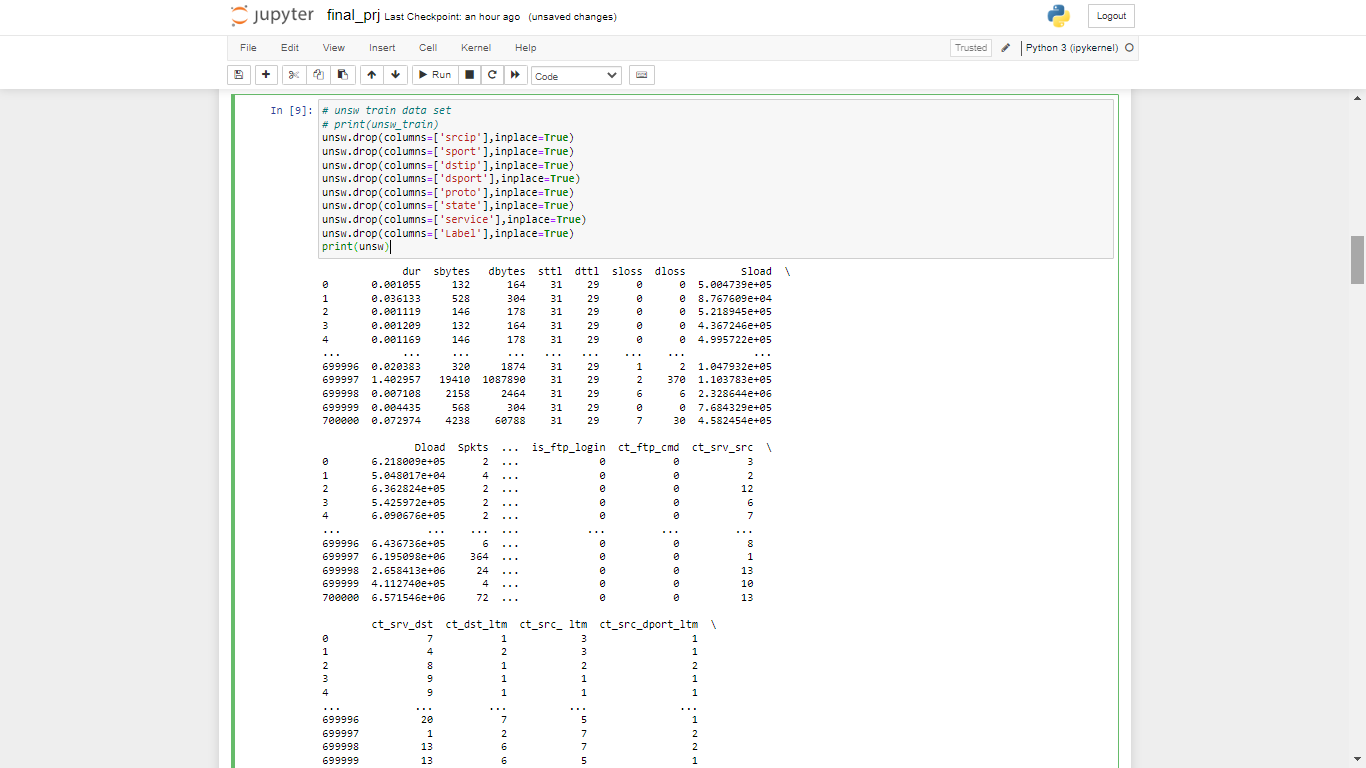


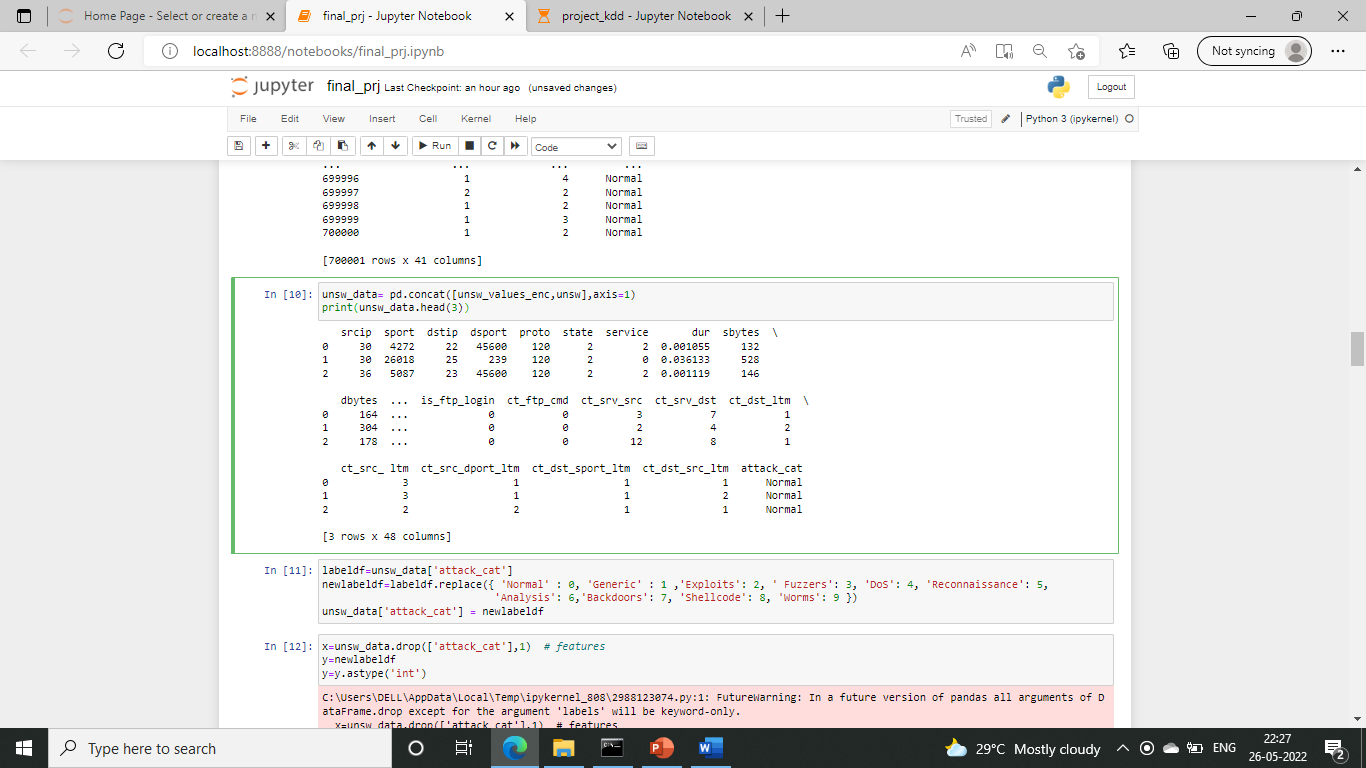
Convert to numeric values:





Dropping the columns and concatenating with categorical values



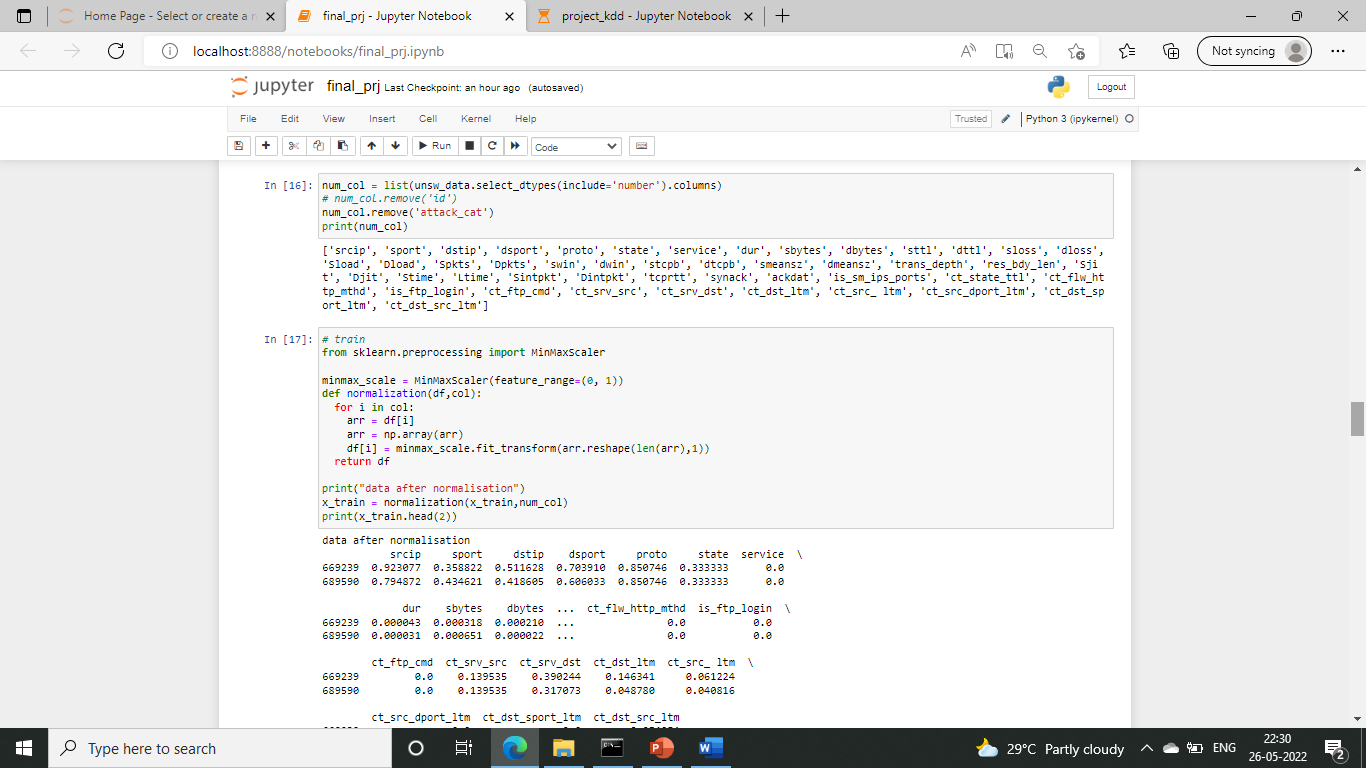


**Dividing training and testing set:**

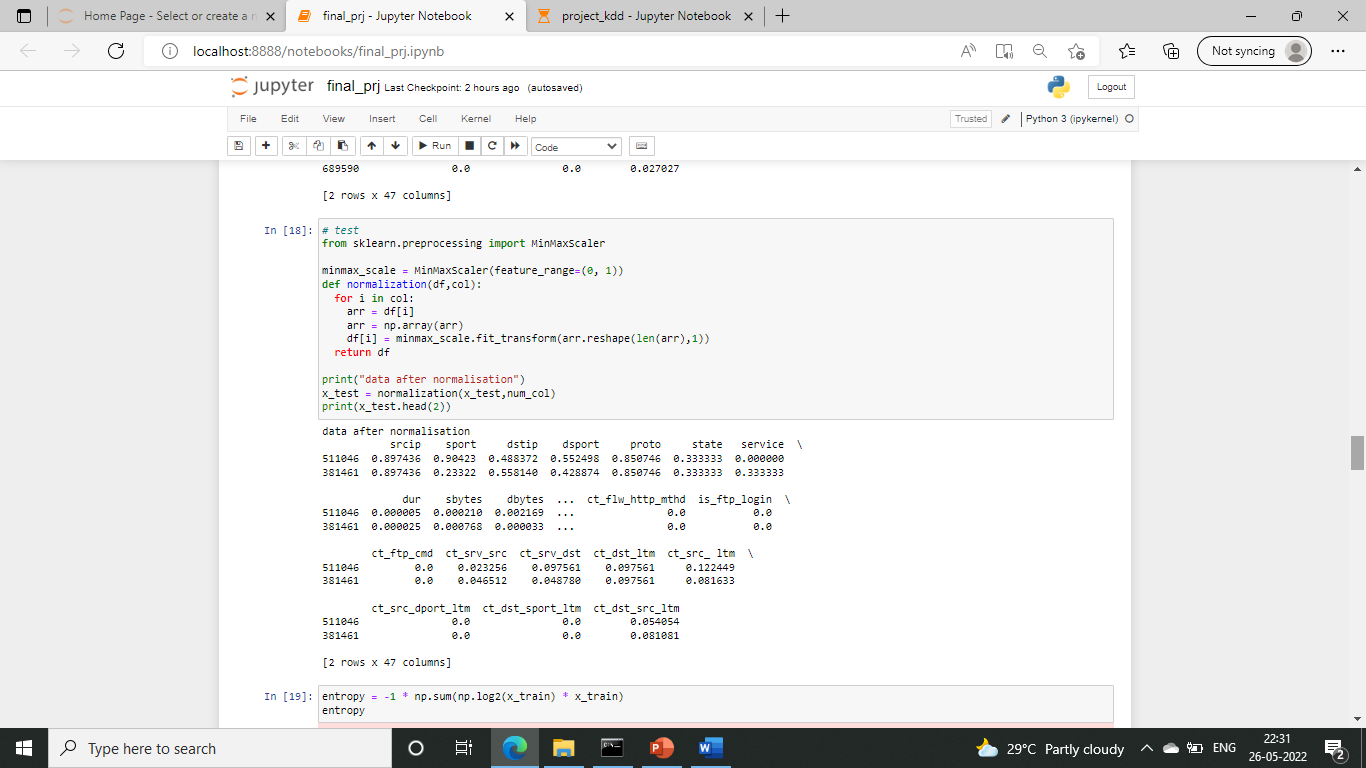


Normalization:

For training set

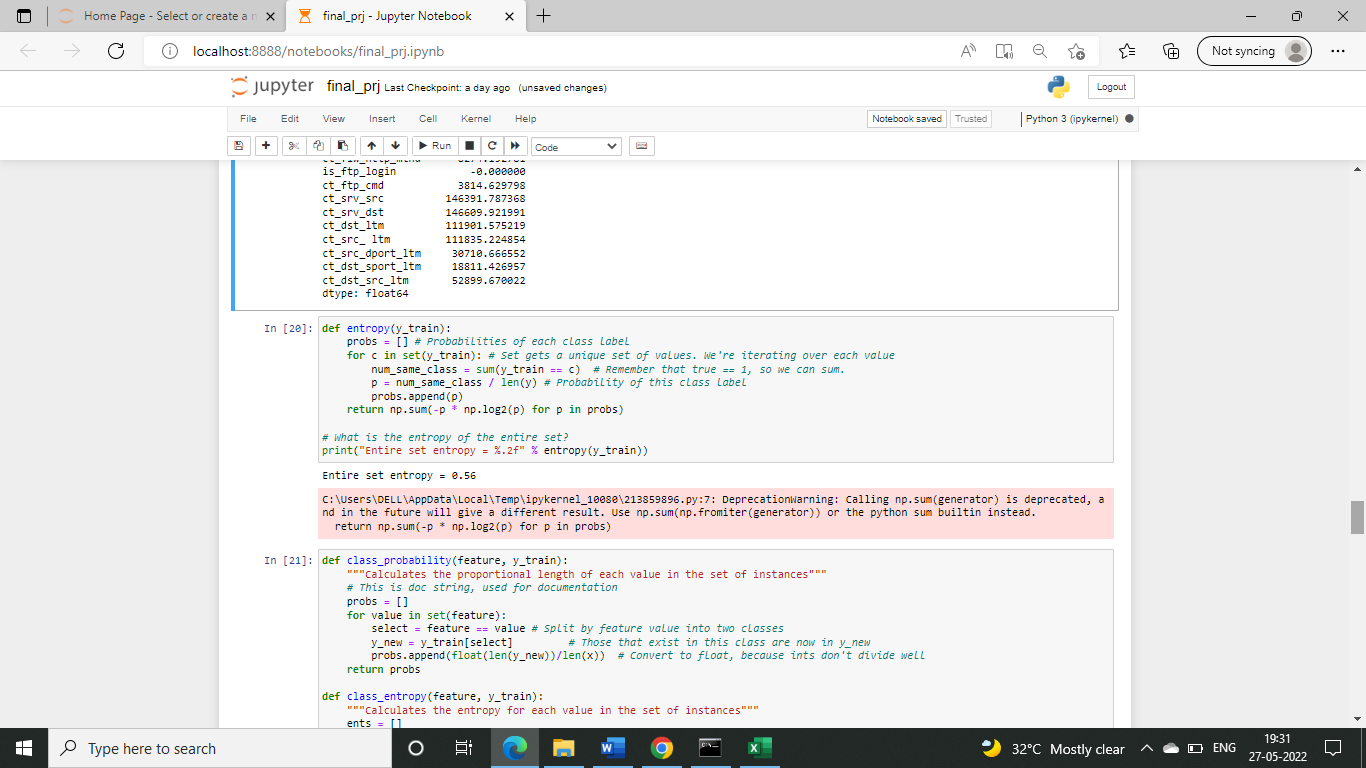


For testing set

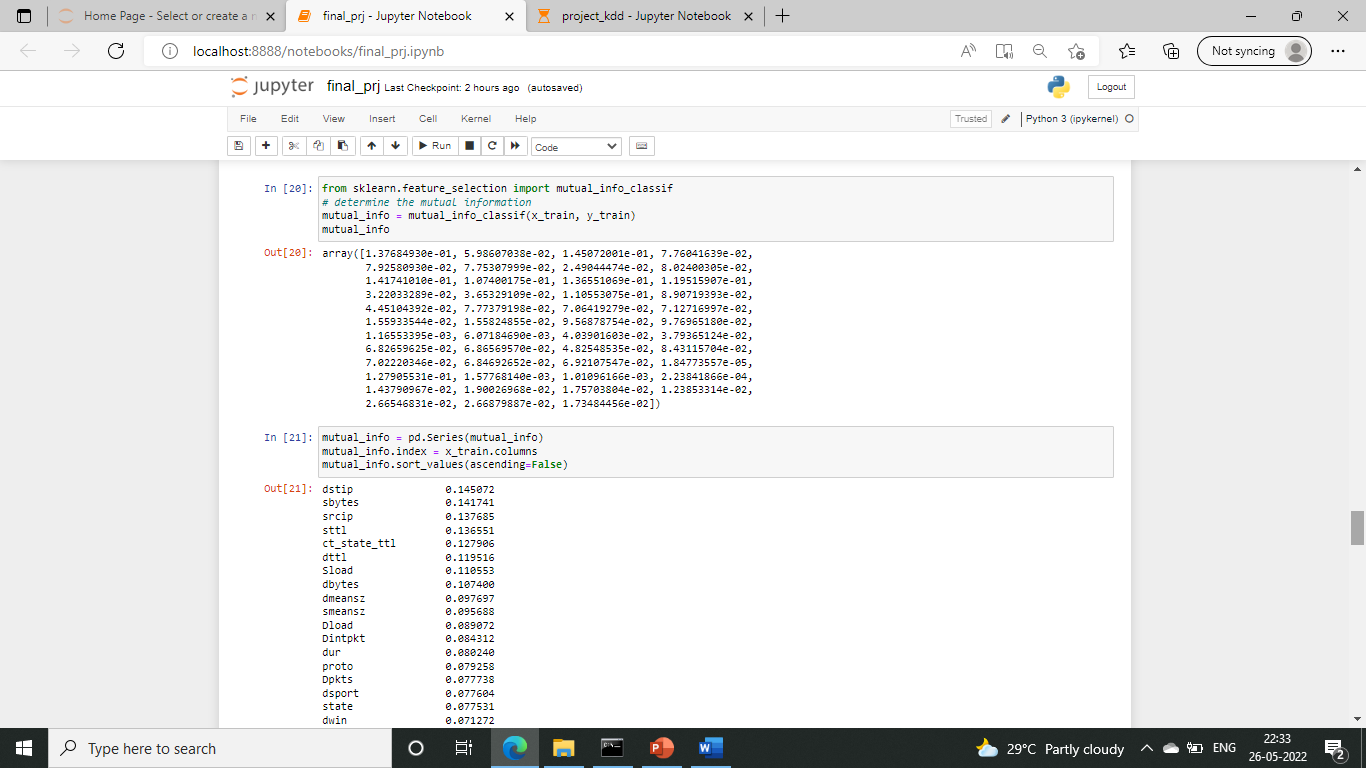


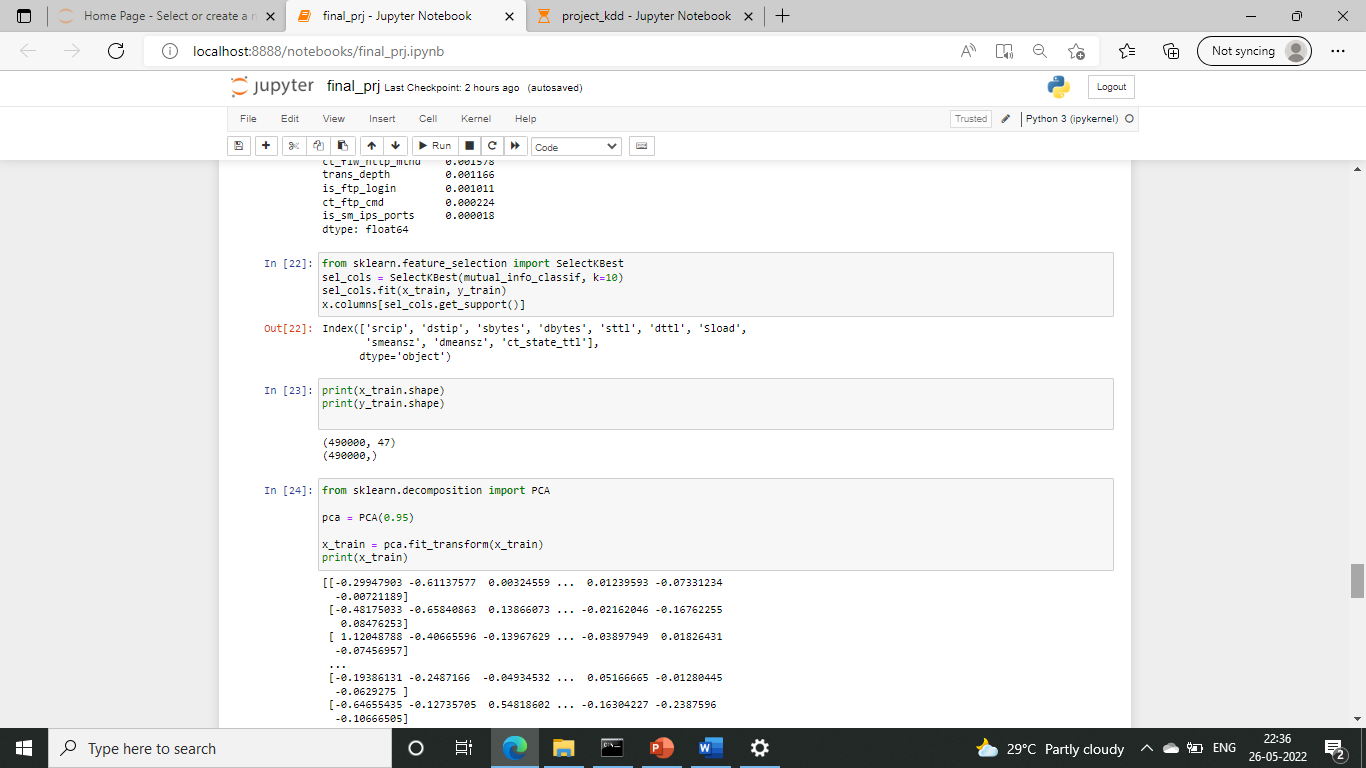
**FEATURE SELECTION AGENT:**

Entropy:

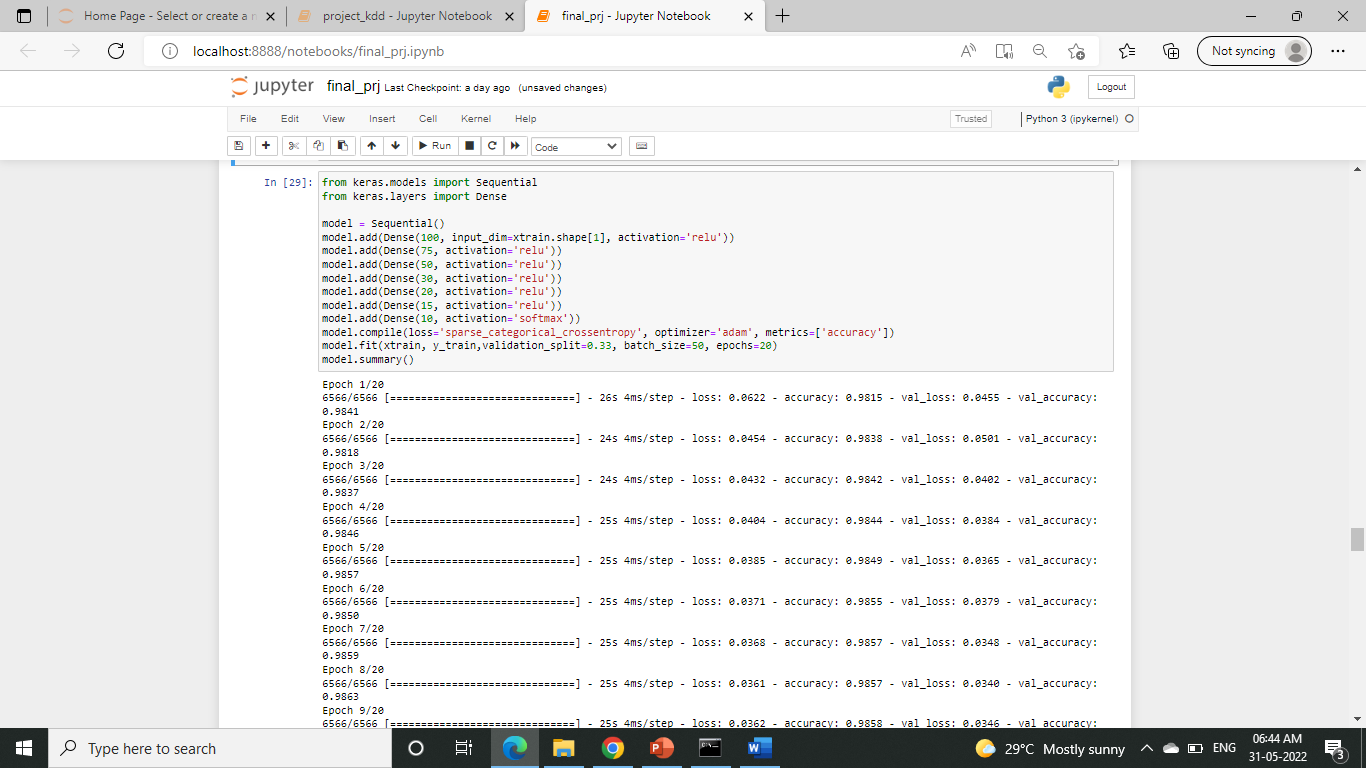


Mutual information gain:





DNN with features selected from mutual information gain

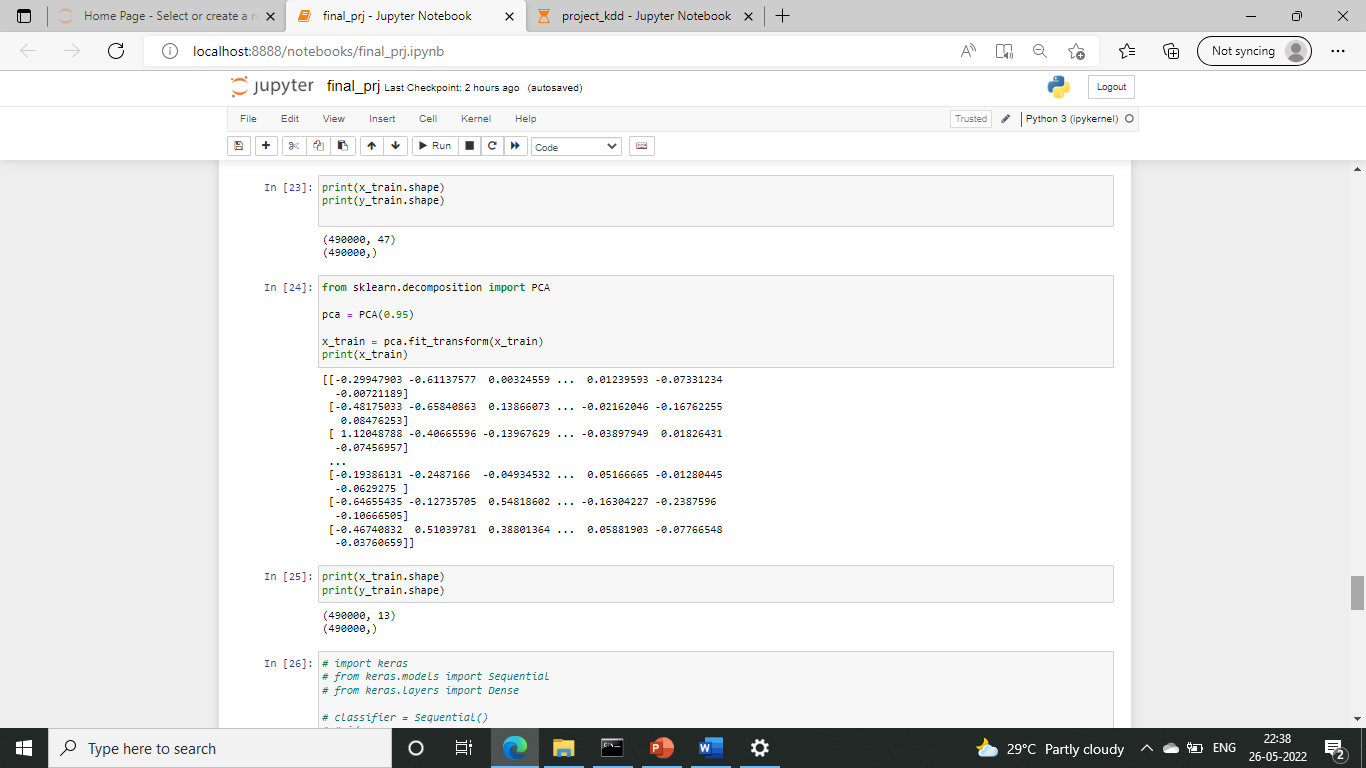


Prediction and accuracy

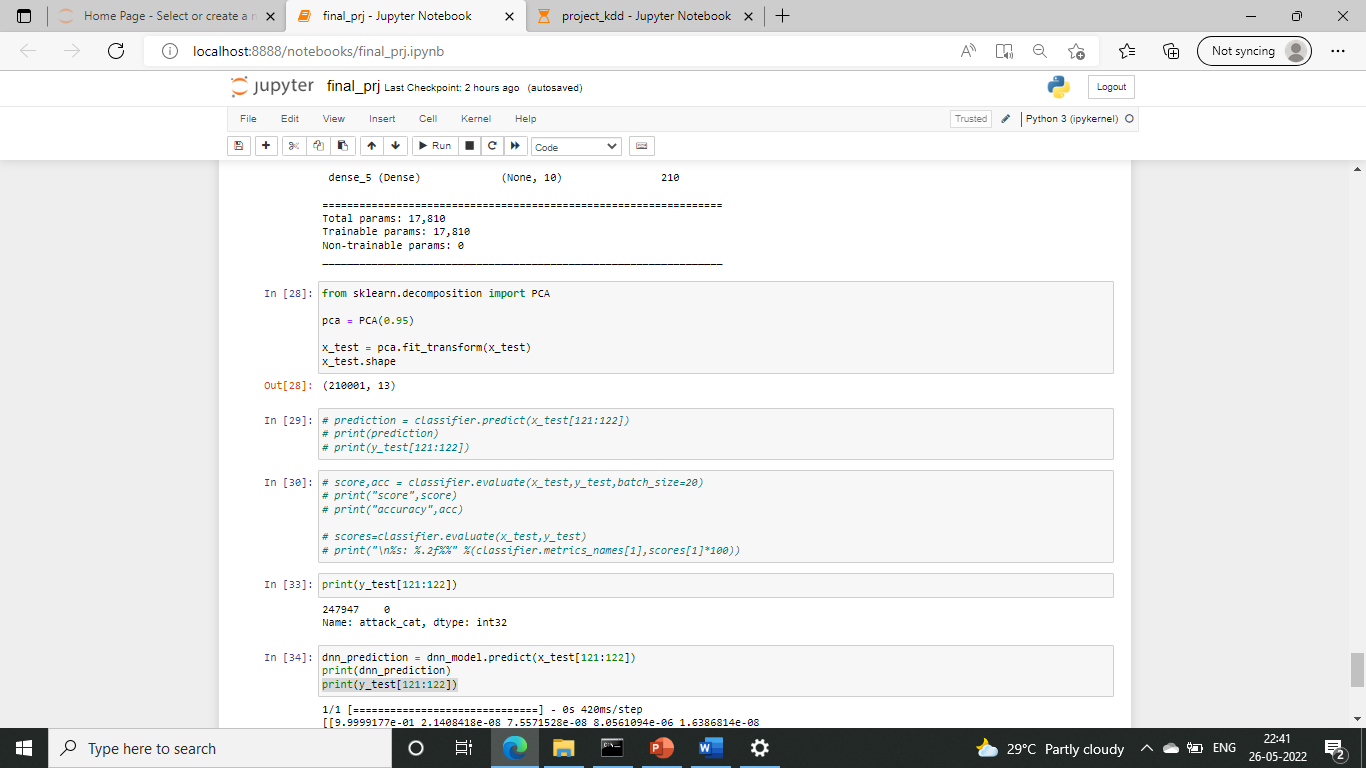


Principal Component Analysis:

For training set:

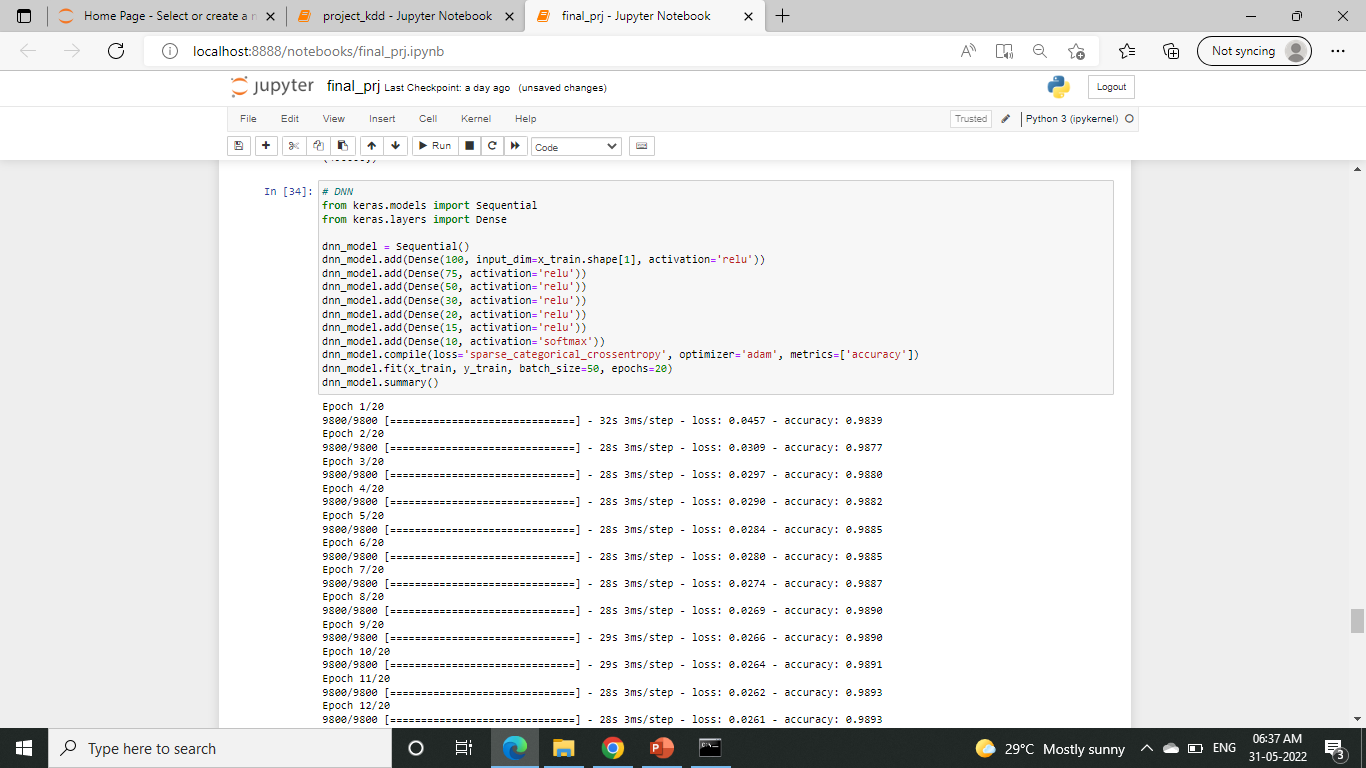


For testing set:

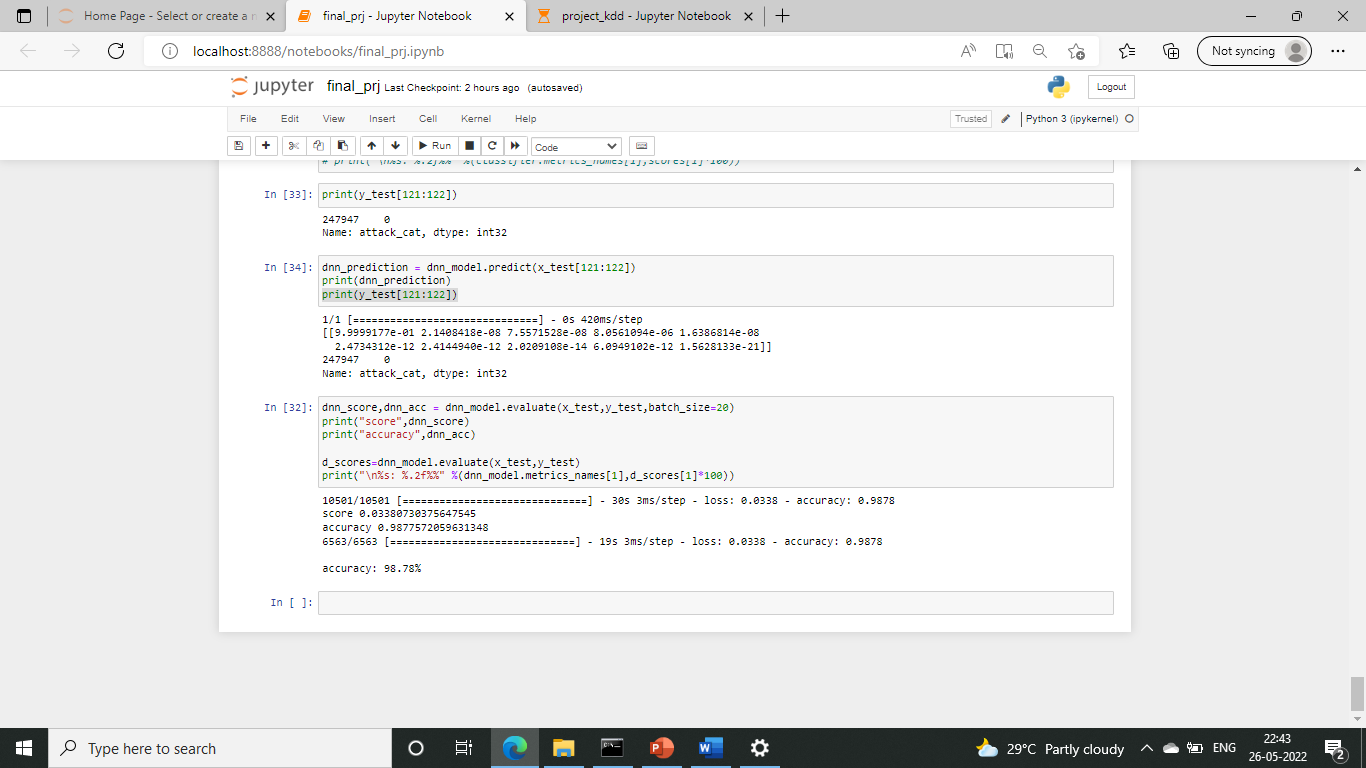


**CLASSIFICATION STRATEGY:**

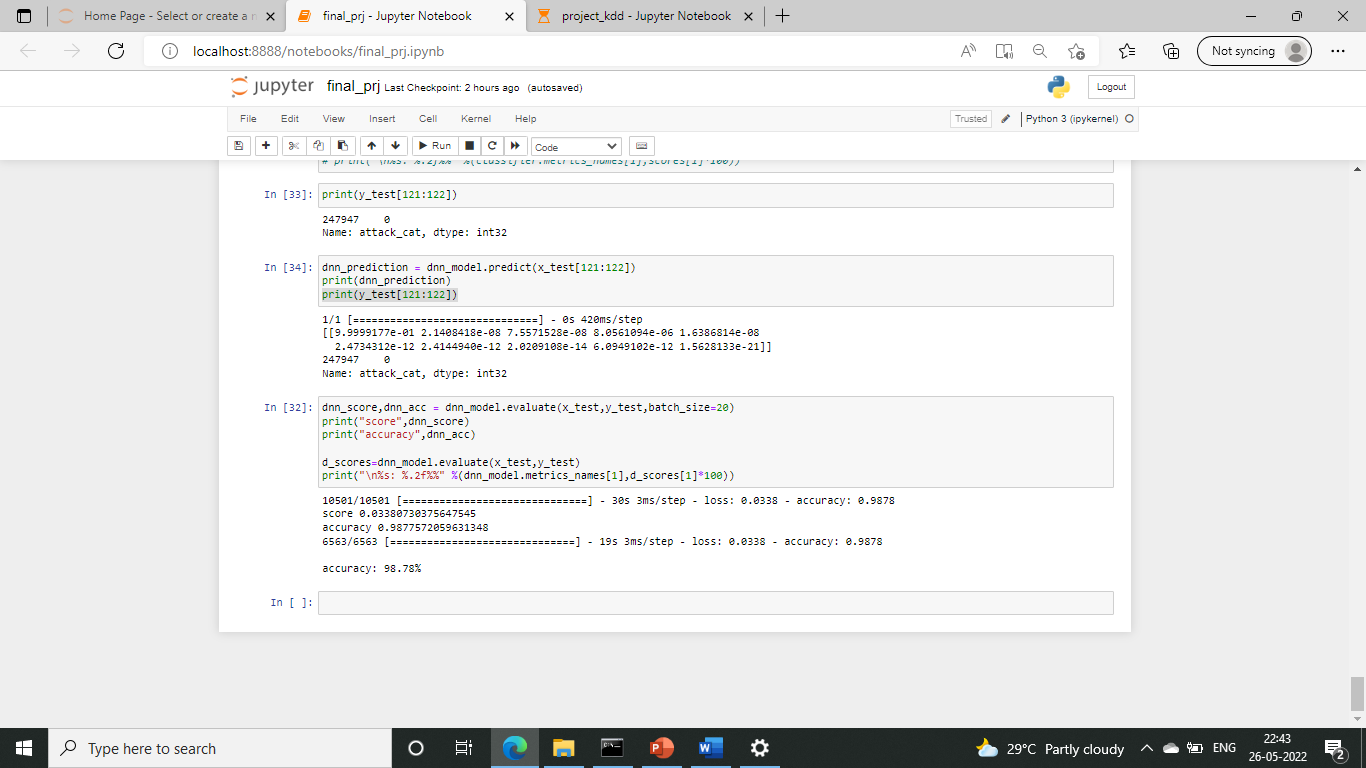
Training the model using DNN:



Prediction



Score and accuracy:



**CHAPTER 6: RESULTS**

In this project, Attack generation is done with the help of NS2 stimulator in ubuntu. The attack is generated in tcl language with the help of aodv protocols and the attacker packets are dropped. This stimulation generates an .nam file which will be used in the next phase to detect the attacks.

I have pre-processed the data set using label encoding by converting the features into categorical values. Then the dataset is divided into training and testing data after which the min-max techniques is used to normalise the dataset to make sure the values are in a uniform scale. Then information gain was calculated for the training dataset with the help of entropy calculation. This was done to select the best features from the given dataset. In this project, I have also implemented the Principle Component Analysis (PCA) model to extract the necessary features to train the dataset. Then the dataset was trained with the DNN classification algorithm which uses Relu activation and softmax activation methods. After training the dataset it was evaluated using the test dataset and predict the results. The result acquired was the expected output and the accuracy of my model is 98.78%.

**CHAPTER 7: CONCLUSION**

Attack detection is an essential commodity in the technical world. So, we need intrusion detection methods to detect these attacks properly. This project helps us to do effective detection which increases our packet delivery ratio. and this paper helps us to build one such model which will tell us if it is an attack and the type of this attack. The model comes with an accuracy of 98.78%. The attack is generated using NS2 stimulator and this output file will be used in the next phase of the project to train and test this model and show its efficiency.

**CHAPTER 8: REFERENCE**

1. DeepADV: A Deep Neural Network Framework for Anomaly Detection in VANETs, Tejasvi Alladi,Bhavya Gera, Ayush Agrawal, Vinay Chamola and Fei Richard Yu , IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 70, NO. 11, NOVEMBER 2021
2. DeepVCM: a Deep Learning Based Intrusion Detection Method in VANET, Yi Zeng, Meikang Qiu, Dan Zhu, Zhihao Xue, Jian Xiong, and Meiqin Liu, IEEE Intl Conference on Intelligent Data and Security (IDS).
3. Deep Learning Approach for Intelligent Intrusion Detection System, R. Vinayakumar, Mamoun Alazab, K. P. Soman , Prabaharan Poornachandran , Ameer Al-nemrat , And Sitalakshmi Venkatraman, IEEE Access ( Volume: 7)
4. Deep Neural Networks for Securing IoT Enabled Vehicular Ad-Hoc Networks, Tejasvi Alladi, Ayush Agrawal , Bhavya Gera, Vinay Chamola, Biplab Sikdar, Mohsen Guizani, CC 2021 - IEEE International Conference on Communications.
5. https://www.educative.io/edpresso/one-hot-encoding-in-python
6. https://github.com/alik604/cyber-security/blob/master/Intrusion-Detection/UNSW\_NB15.ipynb
7. https://www.youtube.com/watch?v=Fw5iiyIHzew
8. https://github.com/Ahamasaleh/Deep-learning-for-intrusion-detection-using-RecurrentneuralnetworkRNN/blob/main/KDDTest%2BBinary\_Classification.ipynb
9. https://github.com/abhinav-bhardwaj/IoT-Network-Intrusion-Detection-System-UNSW-NB15/blob/master/UNSW\_Project.ipynb