National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Nilam Choudhari
Student ID: x20154003

School of Computing
National College of Ireland

Supervisor: Mr.Aqueel Qazmi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Nilam Choudhari
Student ID:	x20154003
Programme:	Cloud Computing
Year:	2021
Module:	MSc Research Project
Supervisor:	Mr.Aqueel Qazmi
Submission Due Date:	16/12/2021
Project Title:	Configuration Manual
Word Count:	XXX
Page Count:	6

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	15th December 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Nilam Choudhari
x20154003

1 Introduction

This article expands on the specifications of proposed system , as well as the software and hardware utilized in the implementation of the project. Also it outlines the procedures taken in the development of the research project, "Detection and analysis of Network Layer Security challenges in cloud using machine-learning Algorithm."

2 System Configuration

2.1 Software Specification

Jupyter Notebook-The downloaded data was split into training, testing, and validation portions using Jupyter Notebook, an open source program.

Python 3.7(64 bit)

2.2 Hardware Specification

Lenovo IdeaPad C340, 256 GB SSD, 8 GB RAM. Processor: 1.8 GHz, Intel Core, i5

Below figure shows the anaconda prompt used for accessing Jupyter notebook as shown in figure 1

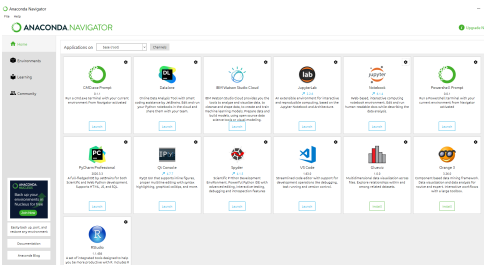

Figure 1: Anaconda prompt

I have downloaded all the libraries and listed them in the conda prompt as shown in figure 2. 1) matplotlib - this was used for data visualization and plotting of graphs 2) Pandas for data analysis and also manipulation of data 3)Scikit learn is the machine learning library

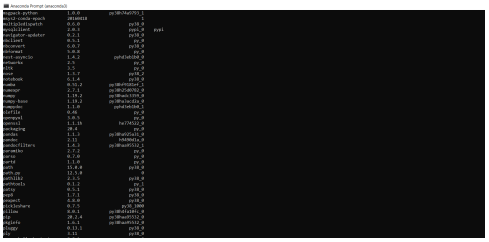

Figure 2: Conda list for libraries

3 Project Implementation

3.1 Data Collection

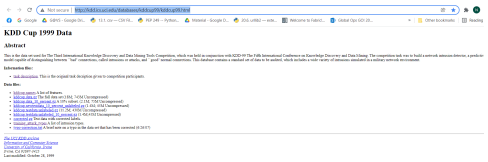

Figure 3: KDD Dataset link

Dataset was picked from the KDD Cup 1999 Data.This data set was used for The Third International Knowledge Discovery and Data Mining Tools Competition, that was conducted in connection with KDD-99, The Fifth International Conference on Knowledge Discovery and Data Mining. The aim for the challenge was to build a comprehensive network intrusion detector, a prediction model capable of differentiating between "bad" connections, known as intrusions or assaults, and "good" regular connections. This database comprises a consistent standard of auditable data, including a wide range of intrusions replicated in a military network context.

3.2 Data preparation

Extracted feature names from the kddcup names file. Since it does contain any name for the label, we will add a column name 'connection-type' for the labels (target variable)

Figure 4: Feature name extraction

After loading the training dataset we saw that the column 'connection-type' is misplaced in the dataset we need to make it appropriate as it is the 'target-variable'. Renamed the column no. 41 as 'connection-type' and dropped column number 42.

Extracted feature names from the kddcup names file. Since it does contain any name for the label, we will add a column name 'connection-type' for the labels (target variable)

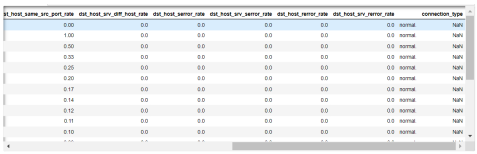

Figure 5: Coloum mislabelled

Figure 6: Renaming and dropping

Then we checked the datatypes for training data

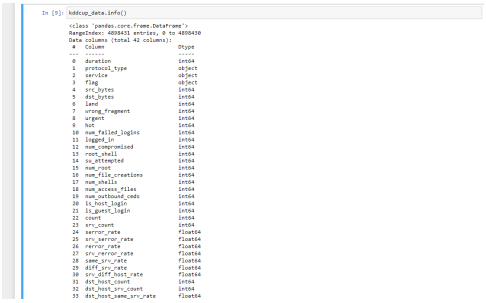

Figure 7: Data types

Then checked the size of the data as shown in figure 8

Later checked missing values in dataset as shown in figure 9

Repeated the same steps for the Testing dataset as well. Checked the sizes of both test and training dataset as shown in figure 10

Later combined both train and test data as shown in figure 11

Next we want to perform numeric encoding on the categorical features. Pandas has a function get-dummies() for this. This function adds additional columns based on the categories present in the feature.As shown in figure 12.

observed that there is an extra '.' in the 'connection-type' column after the connection type. For better visualization and interpretation, removed the '.'. This is shwon in figure 13.

Later we checked for duplicates and removed them from the training dataset.As shiwn in figure 14

Originally, we had 4898431 rows in the data set. After removing the duplicates, we have 1074992 rows. This shows that there were too many redundant rows in the data set. Since this data is a TCP dump, having same values is very much expected.

Next, we scale all the numeric features between 0 and 1. We are given a detailed description of the data set features denoting which features are continuous and which are discrete along with the data types.

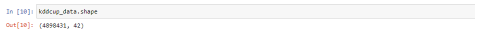

Figure 8: Data size

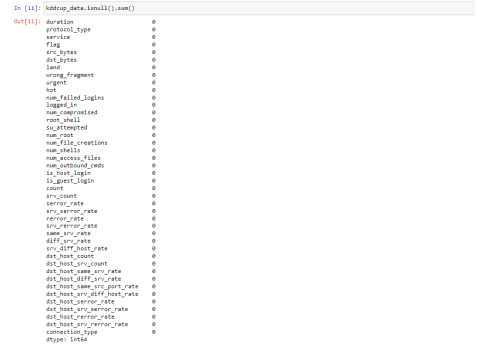

Figure 9: Check of missing values in dataset

Figure 10: Size checking of training and testing data

Figure 11: Combining test and train dataset

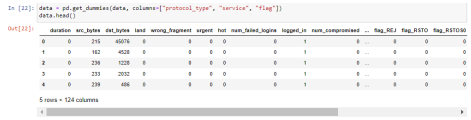

Figure 12: numeric encoding

Figure 13: Correction

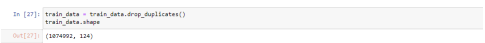

Figure 14: Duplicate removal

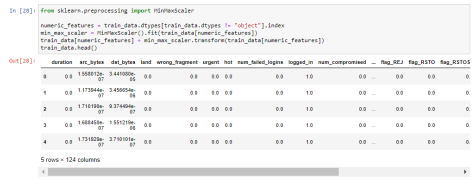

Figure 15: scaling to 0 and 1

Here, we are dealing with binary classification i.e. whether a connection is an attack or not. In our data set, columns are labelled as either 'normal' or as the attack type. So we need to need to denote normal connections as one class type and all the attack types as another class. The new class label would be 0 if the connection is normal and 1 if it is an attack.

Figure 16: Binary classification of data

3.3 Models

The snippets in this section will be all about the different models that were used for implementing the project

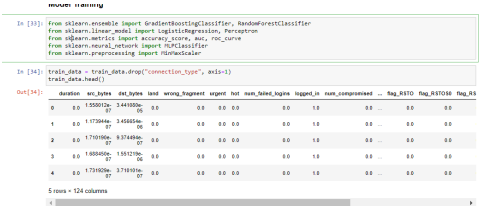

Figure 17: Library import for model training

Accuracy of Multi layer perceptron is highest than other models

We have also used ROC curve and AUC here for evaluating the results. They areused to determine best cut-off which has the lowest false positive rate and the greatestgenuine positive rate which is true positive.

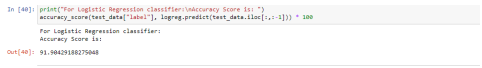

Figure 18: Logistic Regression Accuracy

Figure 19: Random forst accuracy

Figure 20: Gradient Boosting Accuracy

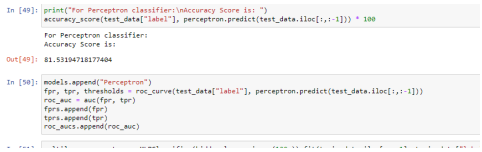

Figure 21: Perceptron Accuracy

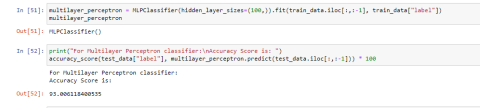

Figure 22: Multi-layer Perceptron Accuracy

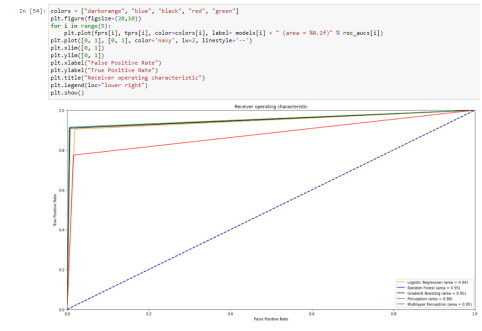

Figure 23: ROC Curve