**ARCHITECTURAL DESIGN**

**Network Security With Machine Learning**

**ABHIJIT SEN**

**INeuron**

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
|  | **V1.0** | **Architectural Design** | **ABHIJIT SEN** |

Table of Content:

|  |  |
| --- | --- |
| Content | Page No. |
| Abstract | 4 |
| 1. Introduction | 4 |
| 1.1 What is Architectural Design? | 4 |
| 1.2 Scope | 4 |
| 1.3 Constraint | 4 |
| 2. Technical Specifications | 5 |
| 2.1 Dataset | 5 |
| 2.2 Deployment | 7 |
| 3. Technology Stack | 7 |
| 4. Proposed Solution | 7 |
| 5. Architecture | 8 |
| 5.1 Data Acquisition | 8 |
| 5.2 Exploratory Data Analysis | 8 |
| 5.3 Feature Engineering | 9 |
| 5.4 Model saving | 9 |
| 5.5 Github | 9 |
| 5.6 Deployment | 9 |
| 6. User Input-Output workflow | 9 |

Abstract:

As the cost of data is going cheaper with time, the number of online transactions is proliferating at a rapid pace. Now-a-day almost every transaction is done online which means most of the people are directly connected over the internet. In such a situation it gives a very good breeding ground for the bad black hat guys to launch an attack into our network and steal valuable resources or even destabilize our networks. Traditional approach to thwart network attacks has been the use of Firewalls but often times they are slow in detecting anomalous traffic as compared to the real time streaming data rate. With the advent of machine learning, it’s possible to detect unknown patterns out of the various characteristics of data packets and classify them as either good or bad traffic at a very fast rate. So, in this project I am going to try various machine learning models which can do the packet classification with high degree of accuracy.

1. Introduction
   1. What is Architectural Design?

The main purpose of the Architectural Design (AD) is to give the internal details of the actual program in order to detect if a network packet is normal or anomalous. It gives the purpose and features of various systems, interfaces of the systems and other various attributes and constraints of the systems. The AD also describes various modules so that a programmer can code by taking direct reference from this document.

* 1. Scope:

This will be an application which will be deployed in the Web. The system will be designed to detect if a network packet is normal or anomalous and take appropriate actions accordingly. Accurate detection of anomalous traffic is extremely important as these kinds of traffic can render our network useless and also can valuable digital resources. Hence, this system is designed to detect a network packet as either normal or anomalous with as much accuracy as possible.

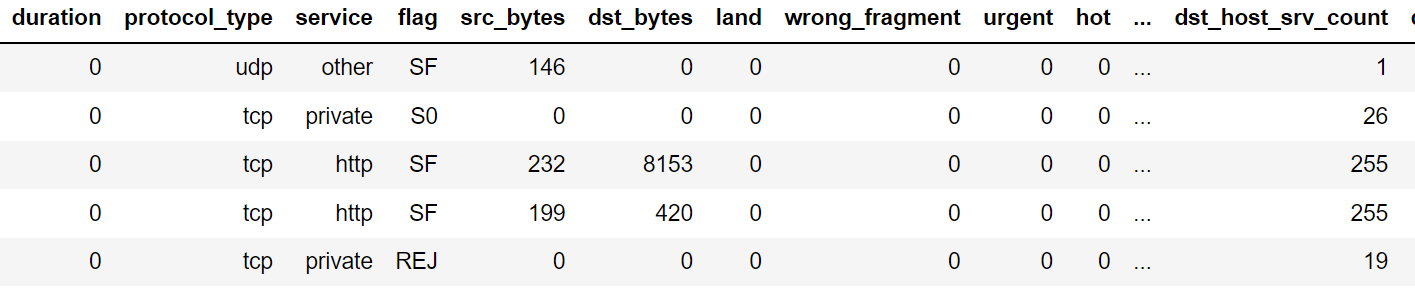
* 1. Constraint:

We predict if a network packet is normal or bad given some of the network packet attributes.

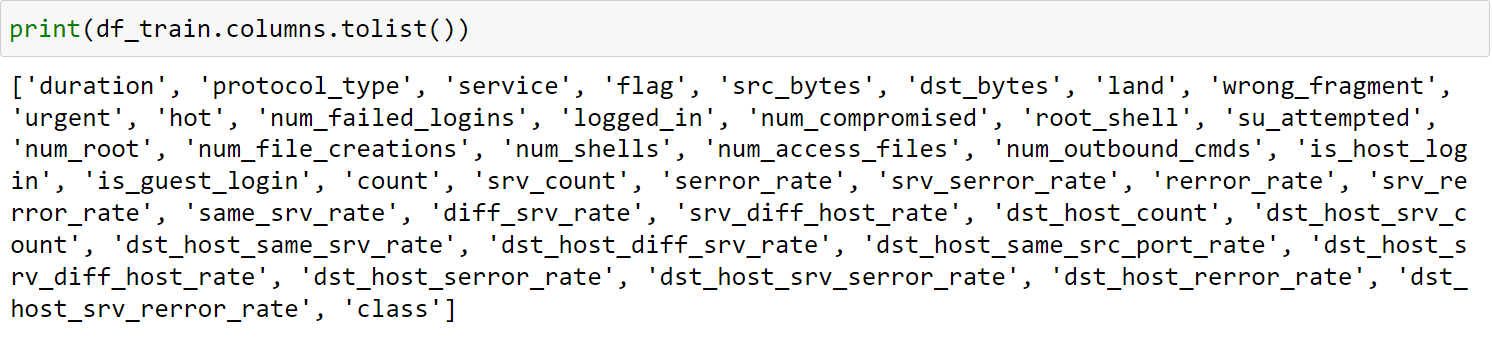
1. Technical Specifications:
   1. Dataset:

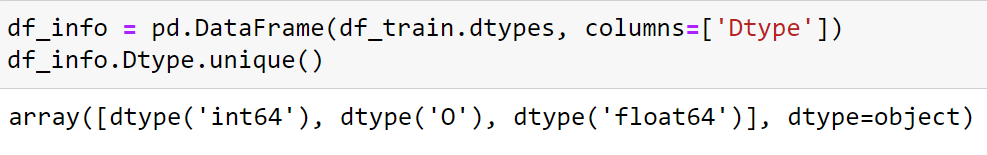
The dataset consists of train and test data in separate files. The train dataset has around 125972 datapoints and test dataset has 22543 datapoints in total. The train and test datapoints are classified as either normal or anomaly in the class column. The main philosophy of keeping a separate train and test dataset is that, the test dataset has a different distribution as compared to train dataset and we through our model wants to check how efficient is our model in detecting zero day attack traffic. The dataset has around 41 attributes associated with each datapoints in both the train and test dataset where some of them are categorical and some are numerical. Our objective is to effectively detect a traffic class given all the various attributes of a packet.

A sample dataset with only a few columns is as shown below:



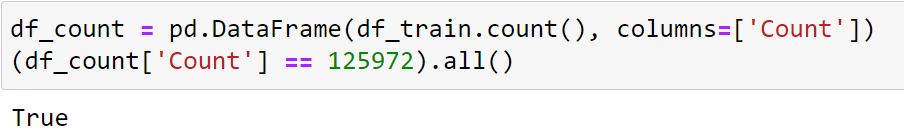
The various columns of the dataset is:

The various types of columns are as shown below:

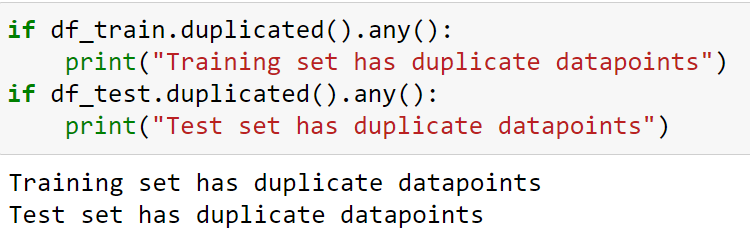


As we can see that, there are primarily two types of columns, categorical and numerical.

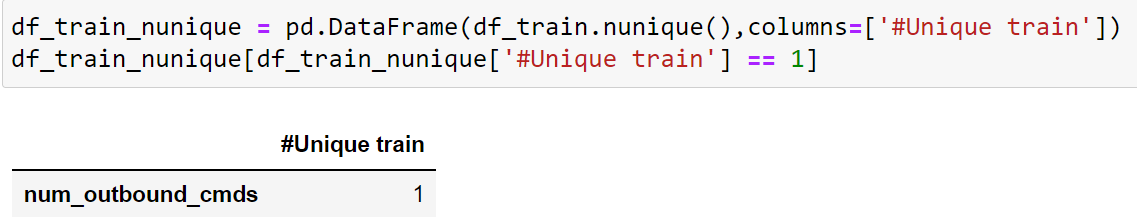
There are no null values in the dataset as shown below:



There are duplicate values in the datasets which needs to be removed:



There is a column which holds only a single value for all the values:



As there is no variance in the data for this feature so it can be removed from the dataset.

In a nutshell, preprocessing the dataset includes checking the number of values for all the features and check if there are any null values for imputation. Then we check then number of unique values for each column and if any column has only a single value then we need to remove such redundant feature. Then we check the various types of features that we have like categorical or numerical etc. and process the data accordingly. The categorical columns are one hot encoded and the numerical columns are done min-max scaling. The class labels are done label encoding- normal as 0 and anomaly as 1. Then various univariate and multivariate analysis is done to select some of the features for doing the further ML modelings.

* 1. Deployment:



1. Technology Stack:

|  |  |
| --- | --- |
| Front End | HTML |
| Backend | Python |
| Deployment | Streamlit |

1. Proposed Solution:

We will be initially mining the various aspects of the dataset. We will see if the dataset has any missing value for imputation. If there are any duplicate values for removal. If there is any non-variant features in the dataset. Doing one hot encoding and min-max scaling for the categorical and numerical datasets respectively. Then we will perform Exploratory Data Analysis for doing the univariate and bivariate feature analysis. Here we check the variability of the various categories for each of the output class. Also for numerical data we see the variance of the mean and distribution using box plots and pdf/cdf plots.

We employ various feature selection mechanisms to select some of the best features for doing the further predictive analysis with various ML/DL techniques. For testing the model, the client will have to manually fill the required features as input and these data will get routed to a web server. The system will pass the data to the backend program where the features will be validated and pre-processed. Finally the preprocessed data will be passed to a trained ML model for doing the final prediction about the class of the packet.

1. Architecture:

Model Testing

Model Building

Start

Deployment

Data Acquisition

Feature Selection

EDA

Data Preprocessing

* 1. Data Acquisition:

The data for our model is taken from the following location: [NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB](https://www.unb.ca/cic/datasets/nsl.html). Both the train and test dataset is given as .arff formal which is transformed to .csv format and stored in my local hard-drive.

After the data is loaded into the local system some steps are performed to check if a particular feature is contributing to the class label like if all the values are same, or if most/all the values of a feature are NaN etc. These feature are identified and removed.

* 1. Exploratory Data Analysis:

In EDA various aspects of the data is seen both visual and otherwise. We check if there are intermittent missing values in the dataset which are imputed. We also check for any duplicate values in the dataset which does not contribute much to the model’s performance and remove them. Then we perform various univariate analyses for the categorical data like check the distribution of the various categories against each class. We prefer those categories which has a different distribution for the various classes. For the numerical data we check the distribution of the mean for each class in box plot and we prefer those feature which has a significant different in the distribution of the mean for each class. We also perform correlation check, if check for bivariate correlation between the various features and the class label and prefer those feature which has a higher correlation with the class labels.

* 1. Feature Engineering:

For the numerical feature, min-max scaler is performed to scale down all the numerical features. For the categorical feature, one-hot encoding is performed. Now, various feature selection methods are performed to select a set of features in order to perform the modeling.

* 1. Model saving:

The various model parameters are saved in a pickle file.

* 1. Github:

The whole project directory will be pushed into the Github repo.

* 1. Deployment:

The project will be deployed from Github into the Heroku platform.

1. User Input-Output workflow:

Error Page

Post

Input

Start App

Pre-process

Prediction