Intrusion Detection System

Abstract

Distributed Denial of Service (DDoS) attack is a menace to network security that aims at exhausting the target networks with malicious traffic. The goal of this project was to use classification models to predict the benign or DNS in network in order to help improvetp prevent cyber attacks and maintenance stability. I worked with data provided by UNB university, leveraging categorical feature engineering along with a random forest model to achieve promising results for this multiclass problem. After refining a model, I built an interactive visualiztion and communicate my results using seaborn library.

Design

CICDDoS2019 contains benign and the most up-to-date common DDoS attacks, which resembles the true real-world data (PCAPs). It also includes the results of the network traffic analysis using CICFlowMeter-V3 with labeled flows based on the time stamp, source, and destination IPs, source and destination ports, protocols and attack What makes this dataset special it generate background attakcs and threats.

Data

The dataset contains 5074413 combined of malcious and bengin with over 88 features for each, 20 of which are categorical. 12 DDoS attacks includes NTP, DNS, LDAP, MSSQL, NetBIOS, SNMP, SSDP, UDP, UDP-Lag, WebDDoS, SYN and TFTP on the training day and 7 attacks including PortScan, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag and SYN in the testing day. The traffic volume for WebDDoS was so low and PortScan just has been executed in the testing day and will be unknown for evaluating the proposed modendertaken to inform baseline models and feature engineering

Segoe UI

Feature Importance

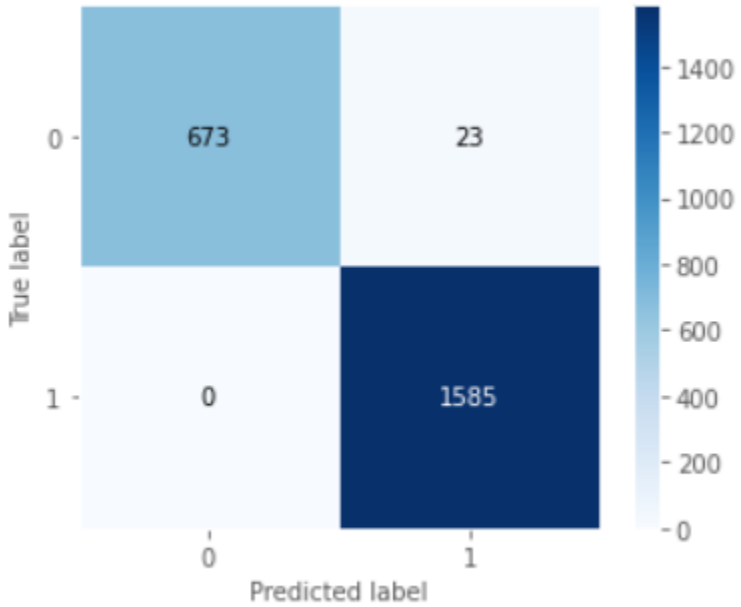
1. Encoding categorical features to Int variables
2. Used feature importance to highlight correlate feature to the target
3. Selecting subsets of the total unique values and the basis for dimensionality reduction and feature selection that can improve the efficiency and effectiveness of a predictive model on the problem

Models

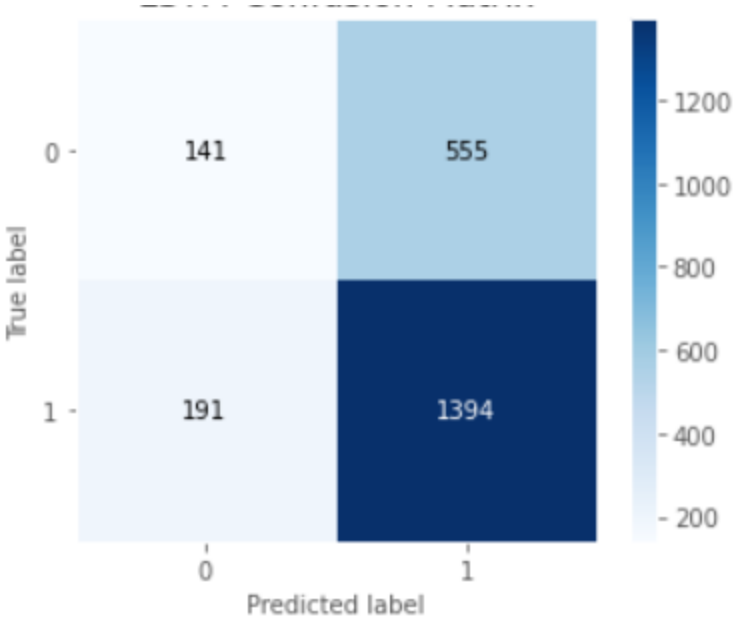
Used RNN and LSTM for the reason that in intrusion detection system we need to remember the traffic the goes through the system and analyze it and an LSTM provide that as it provides three gates. forget gate, input gate and output gate thus this will help in blocking and remembering the traffic. Then after that I will use transformer model in which it implements what is called attention that is advanced and faster than LSTM to improve my model

Model Evaluation and Selection

- RNN : `Accuracy: 67.30%`
- LSTM: `Accuracy: 98.99%`
- LSTM



- RNN



Transformer model to be added

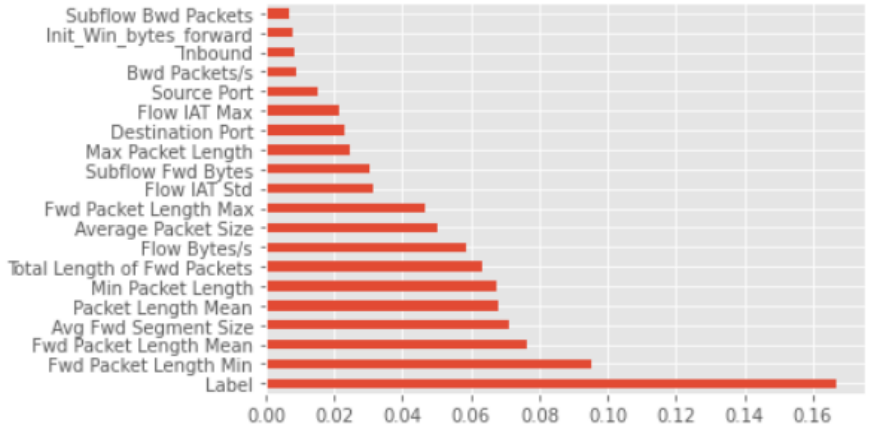
Tools

- Numpy and Pandas for data manipulation
- Scikit-learn for modeling
- Matplotlib and Seaborn for plotting
- Keras and tenserflow for deep learning
- LSTM, RNN and transformer

Communication

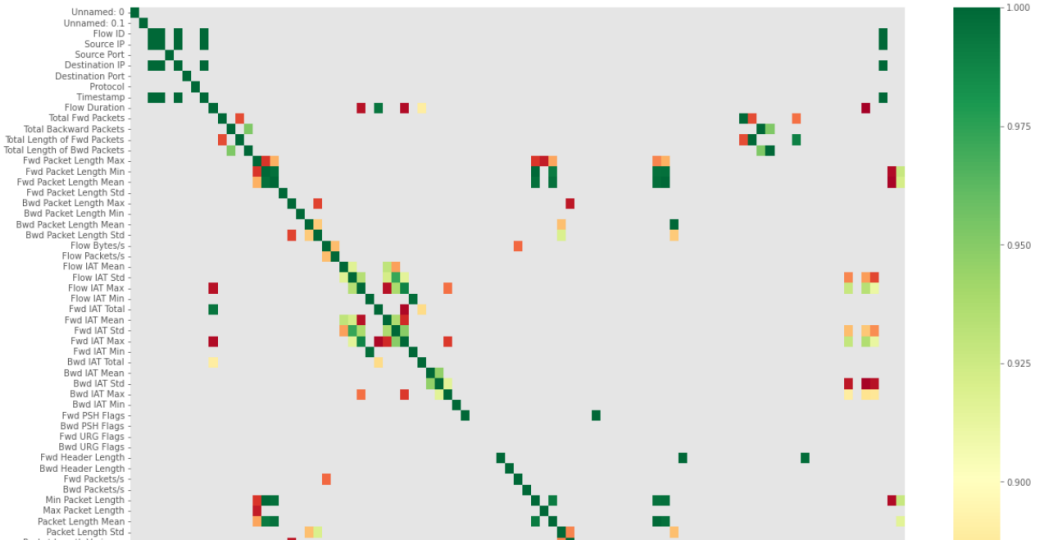
In [15]:
```
(pd.Series(model.feature_importances_, index=df.columns)
   .nlargest(20)
   .plot(kind='barh'))
```

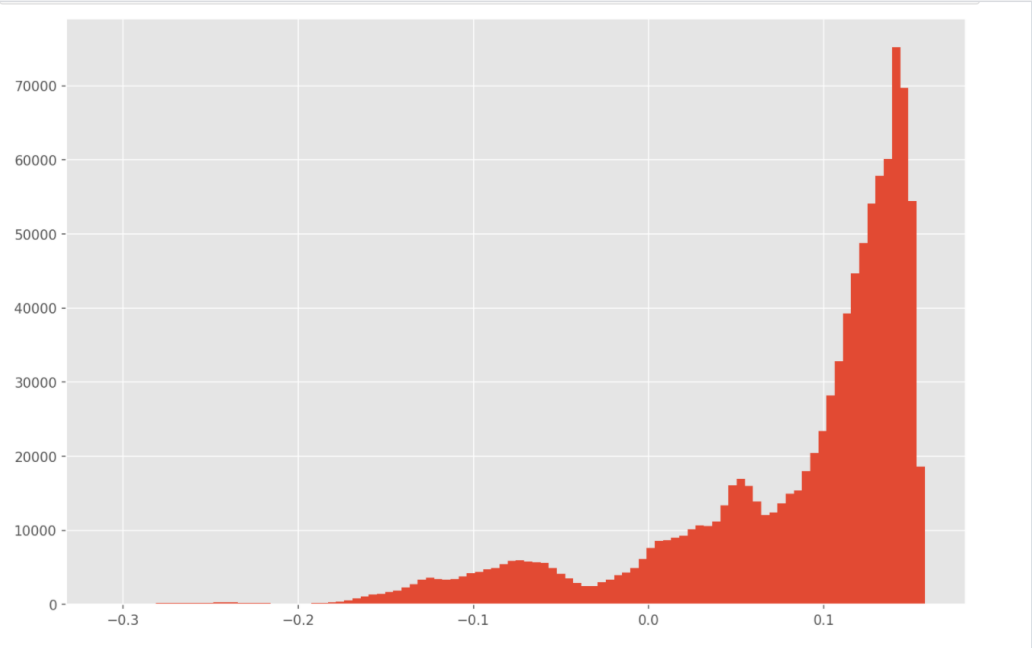
Out[15]: <AxesSubplot:>



We have below variables which are higly correlated > 0.8 which will cause real problem, to fix it we will be performing PCA

In [17]:
```
corrmat = df.corr()
top_corr_features = corrmat[corrmat>=.8]
plt.figure(figsize=(20,20))
#plot heat map
g=sns.heatmap(top_corr_features,cmap="RdYlGn")
```




70000
60000
50000
40000
30000
20000
10000
0
−0.3
−0.2
−0.1
0.0
0.1