NETWORK ANALYSIS USING MACHINE LEARNING

INTRODUCTION

Situation | Problem Definition : This project is all about predicting the types of a new protocal with the help of given input data set that contains all most 84 different features like packet length, flow duration, segment size, header length and many more.

Task: We were assigned with the task to come up with an efficient machine learning model which will result in maximum accuracy on a given data set.

Action: To do this work in an organised way, we enlisted all the relevent classification machine learning algorithm on a paper which fits the problem like SVM kernal, Decision Tree and Random forest . After that we started to analyse the perfomance of all those models meticulously.

Result: Eventually we got maximum accuracy of 98% on both training set and test set with help of random forest.

REQUIREMENT

Technology

- Python
- Machine Learning
- Data Science

Libraries

- Numpy
- Pandas
- Sklearn
- Matplotlib
- Seaborn

Software Used

- Notepad++
- Microsoft Office Word
- Google Colab
- Window
- Ubuntu

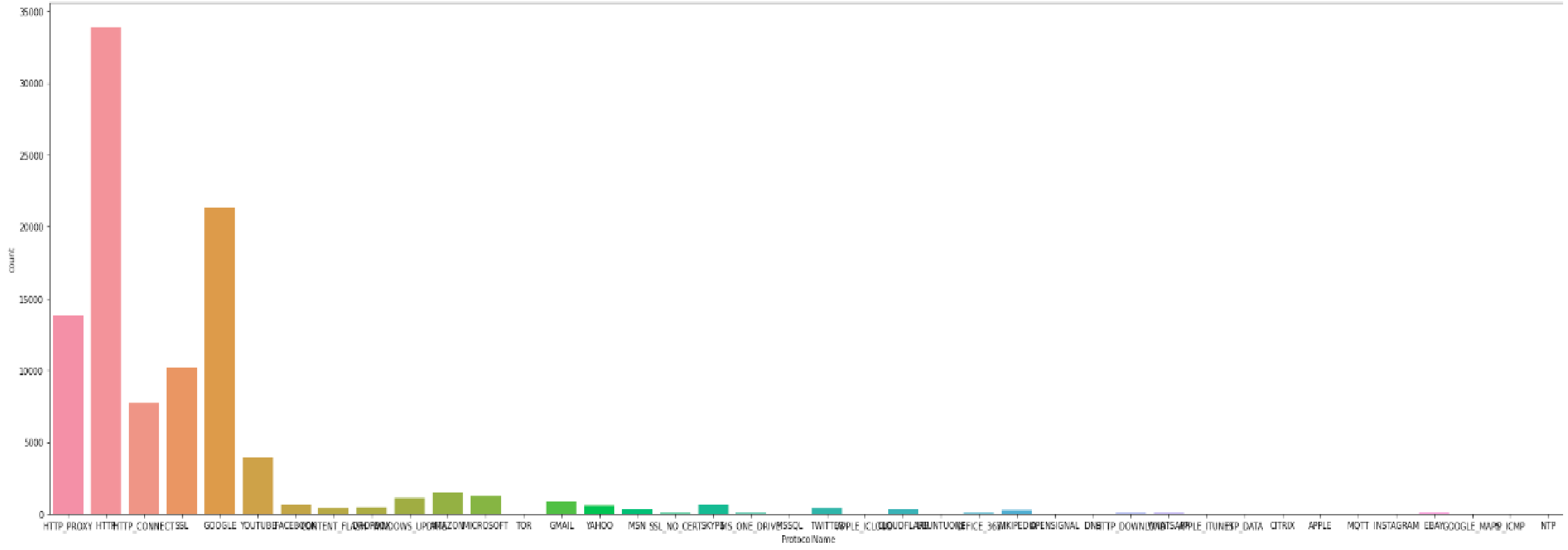
GRAPH EXPLORATORY ANALYSIS

- It’s also known as Exploratory Data analysis (EDA).
- It refers to the critical process of performing Initial investigations on data.
- It’s used to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.
- Improves understanding of variables by extracting averages, mean, min. and max Values etc.
- Discover errors, outliers, and missing values in the data.
- Identify pattern by Visualizing data in graphs such as box plots, scatter plots, and histograms.

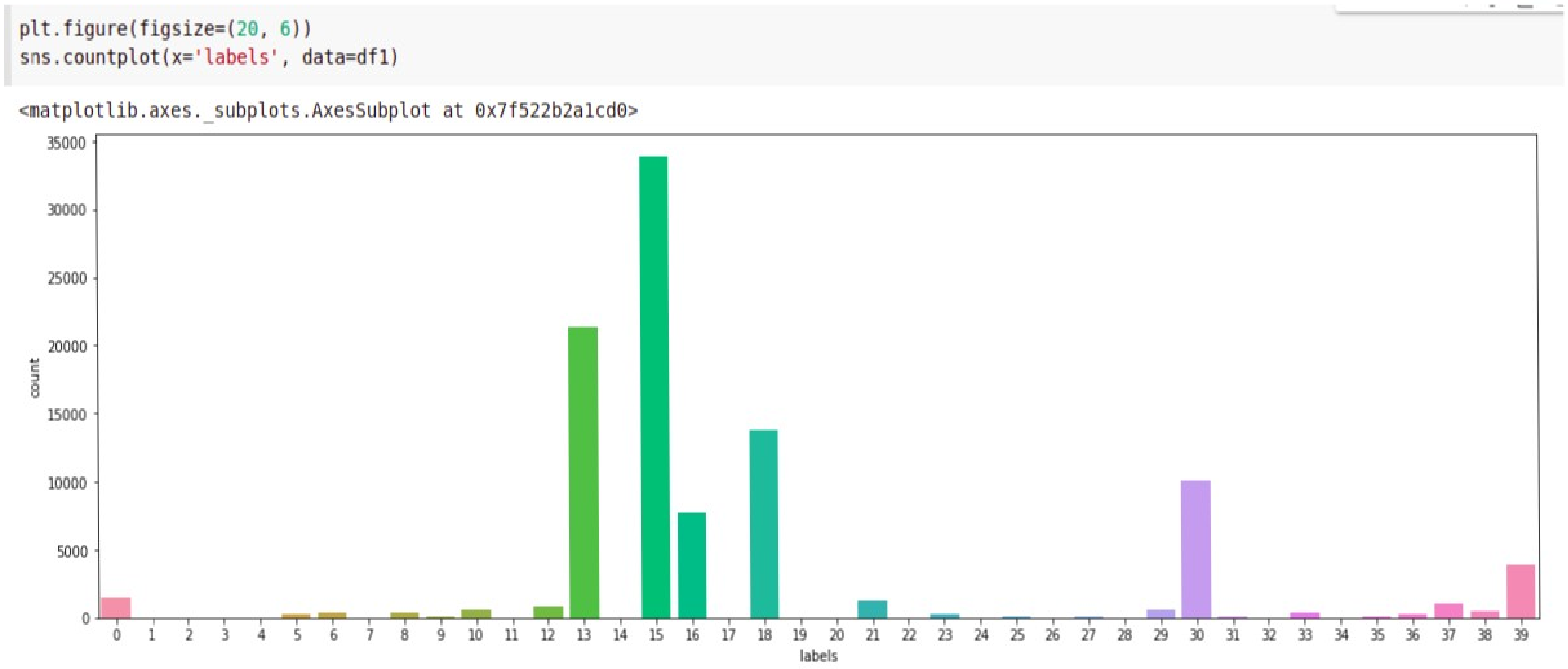
- Count plot of ProtocolName or Labels

```python
plt.figure(figsize=(35, 10))
sns.countplot(x="ProtocolName", data=df1)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f522c131ad0>
```



- Label - Protocol Name



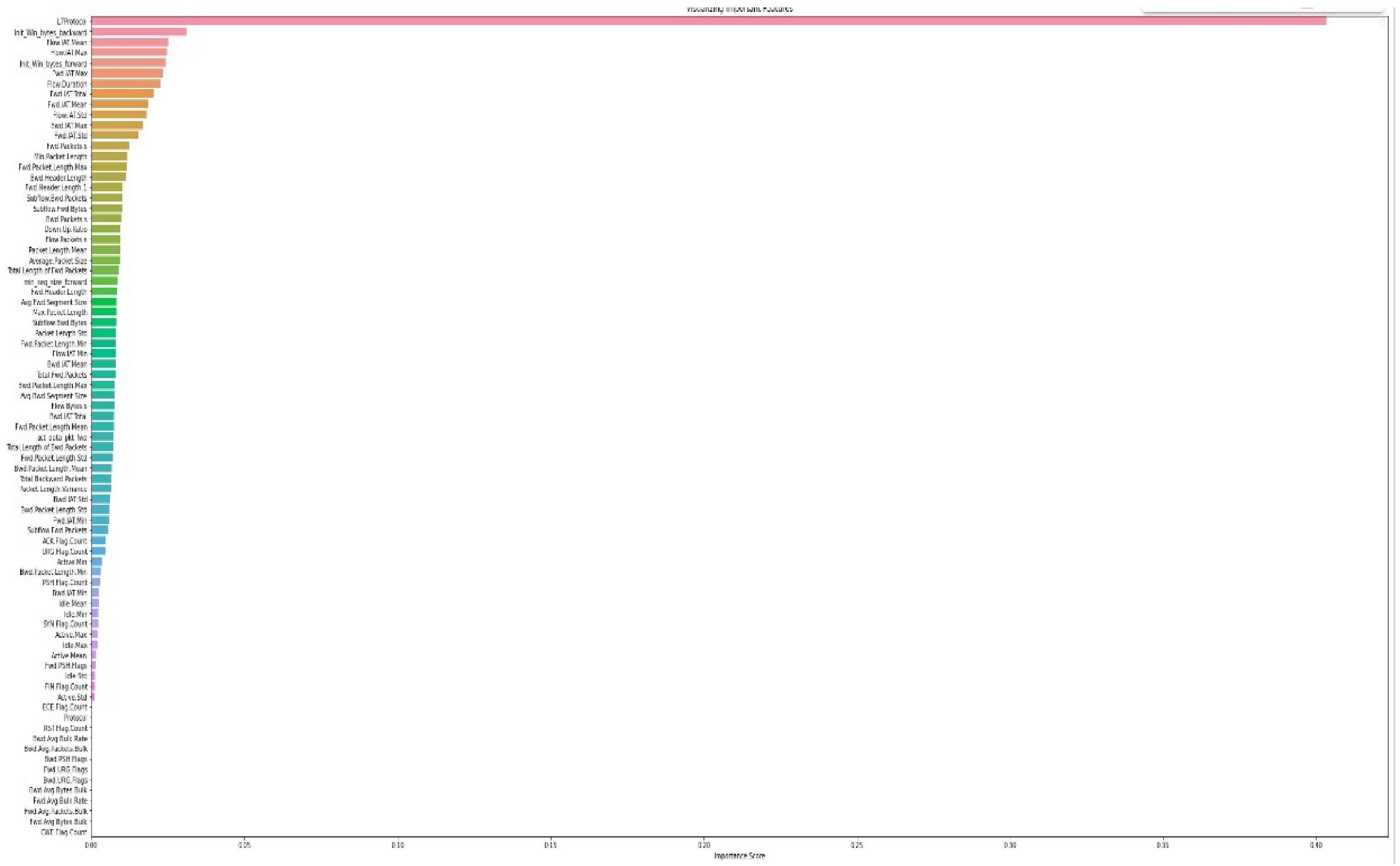
```
plt.figure(figsize=(20, 6))
sns.countplot(x='labels', data=df1)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f522b2a1cd0>
```

- **15 – HTTP**
- **13-Google**
- **18-HTTP_Proxy**
- **30-SSC**
- **39-HTTP_Connect**
- **0-Youtube**

- Visualizing Important Features (Importance Scroe Vs Attribute)

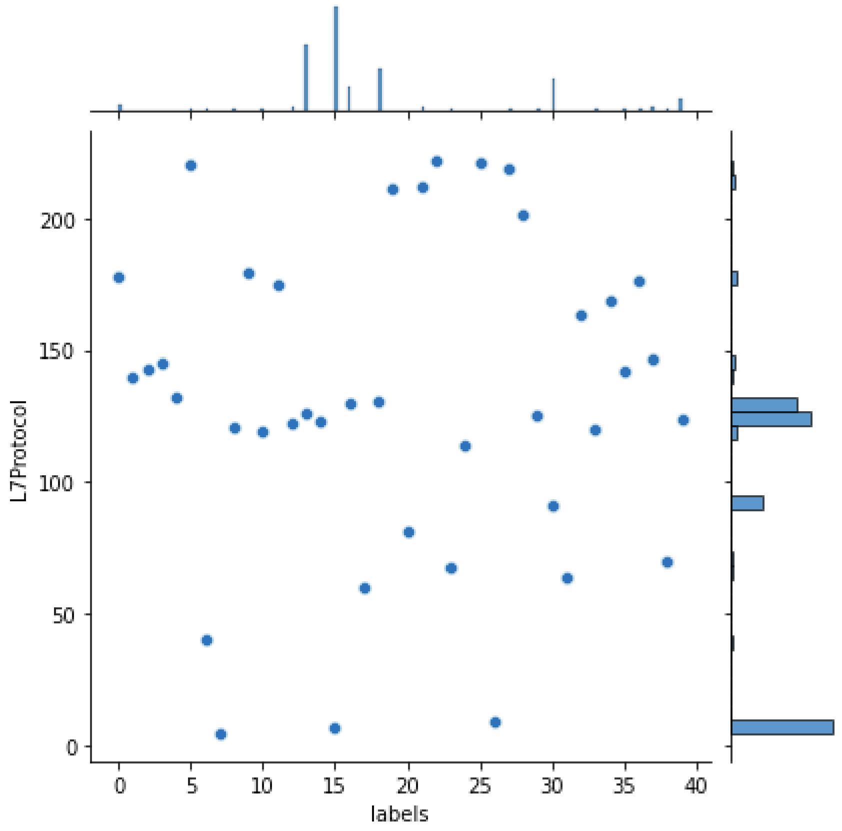
L7Protocol	0.403257
Init_Win_bytes_backward	0.031210



- Joint plot between labels and L7Protocol

```
sns.jointplot(x='labels', y="L7Protocol", data=df2)
```

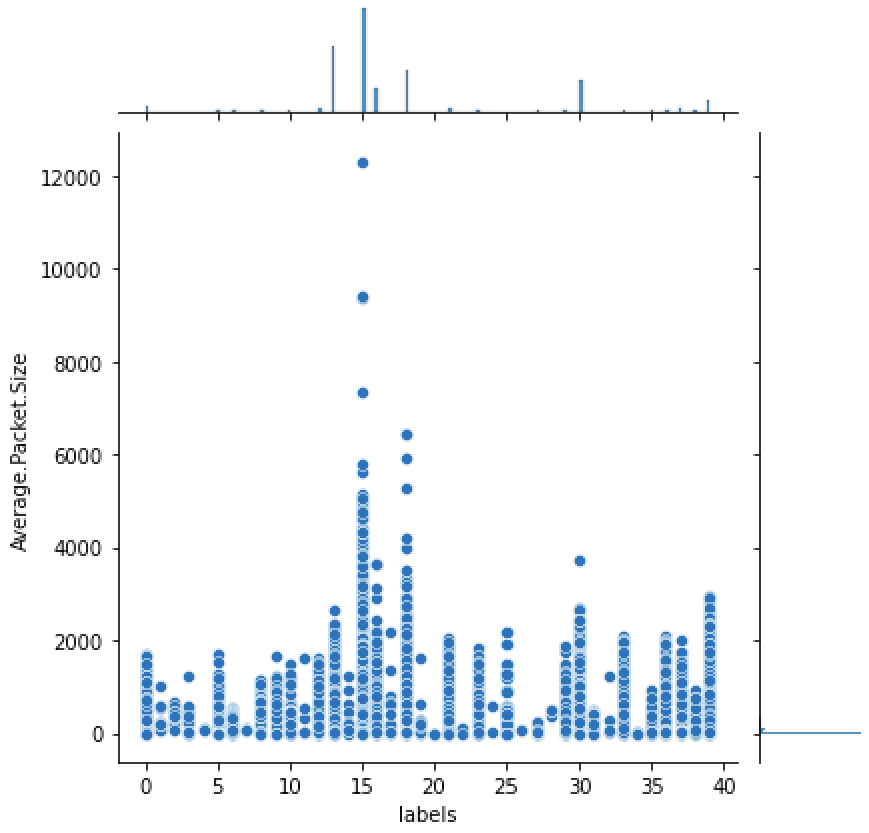
```
<seaborn.axisgrid.JointGrid at 0x7f522aa6f310>
<Figure size 720x432 with 0 Axes>
```



- Joint plot between labels vs avg.packet.size

```
sns.jointplot(x="labels", y="Average.Packet.Size", data=df2)
```

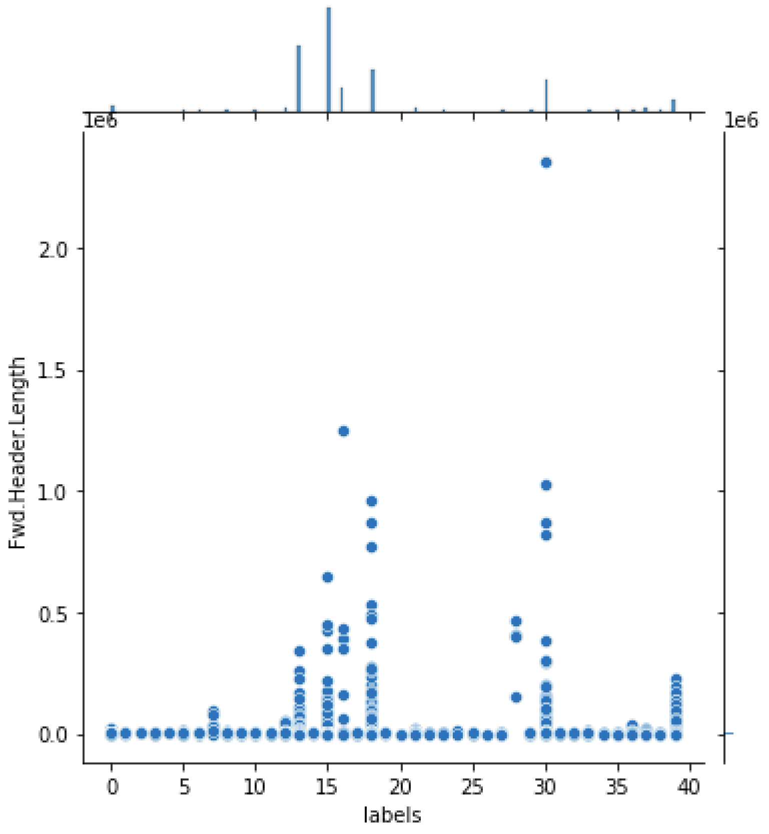
```
<seaborn.axisgrid.JointGrid at 0x7f522addf250>
```



- Joint plot between labels and forward.header.length

```
sns.jointplot(x='labels', y='Fwd.Header.Length', data=df2)
```

```
<seaborn.axisgrid.JointGrid at 0x7fc5c2e258d0>
```



MODEL DESCRIPTION (RANDOM FOREST)

- **What is random forest?**

  Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result.

- **How Random Forest Works ?**

  Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction

MODEL INTIUTION (RANDOM FOREST)

STEP 1: Pick at random K data points from the Training set.

⇩

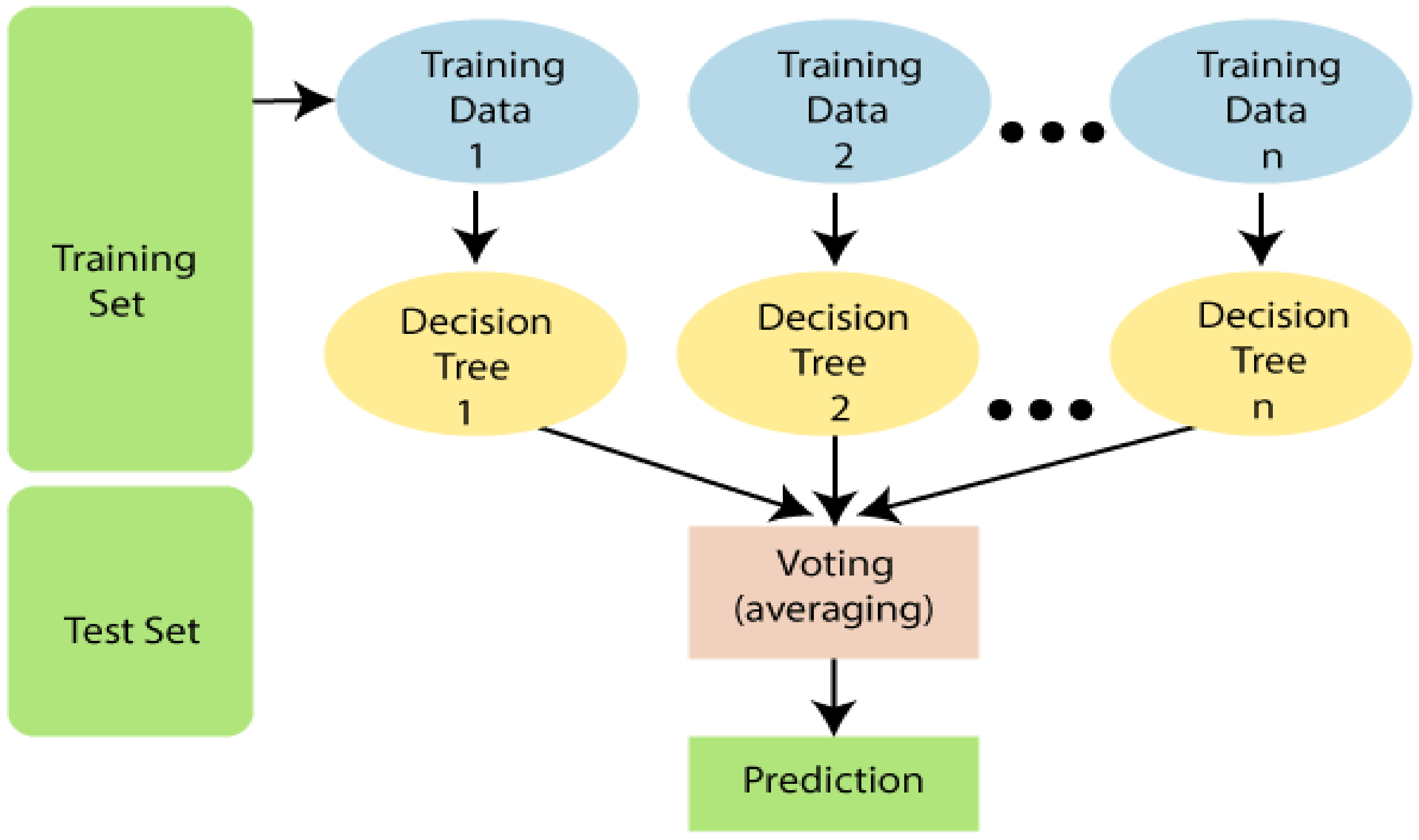
STEP 2: Build the Decision Tree associated to these K data points.

⇩

STEP 3: Choose the number Ntree of trees you want to build and repeat STEPS 1 & 2

⇩

STEP 4: For a new data point, make each one of your Ntree trees predict the category to which the data points belongs, and assign the new data point to the category that wins the majority vote.

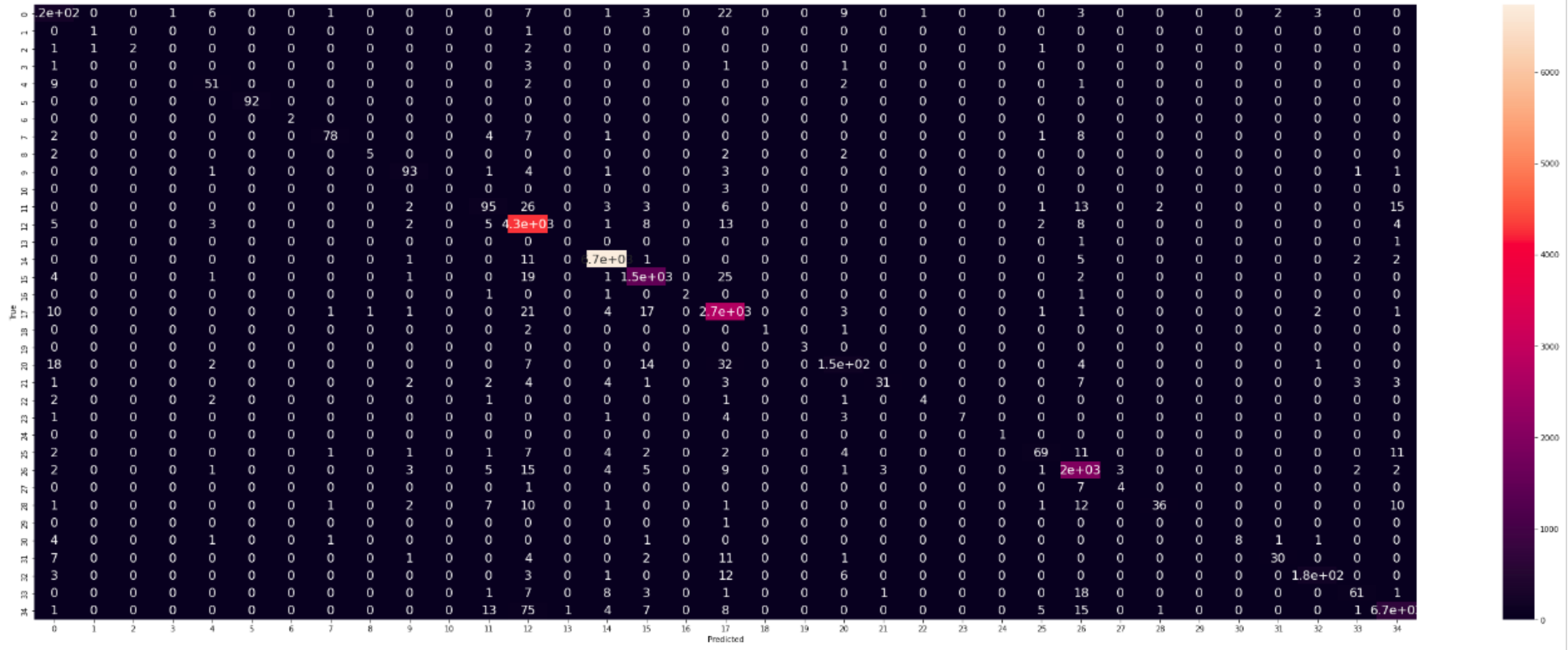

Training Set
Test Set
Training Data 1
Training Data 2
Training Data n
Decision Tree 1
Decision Tree 2
Decision Tree n
Voting (averaging)
Prediction

IMPLEMENTATION STEPS

- Importing The Libraries
- Importing The Dataset
- Splitting The Dataset Into Training Set and Test Set
- Removing non-integer column and Encoding The String Data Column Into Integer Column
- Feature Scaling
- Training The Random Forest Classification Model On Training Dataset
- Predicting A New Result
- Predicting The Test Set Result
- Building And Ploting The Confusion Matrix
- Data Visualisation

```python
plt.figure(figsize=(40, 15))
sn.heatmap(cm, annot=True, annot_kws={"size": 16}) # font size
plt.ylabel('True')
plt.xlabel('Predicted')
```

```
Text(0.5, 114.0, 'Predicted')
```



CONCLUSION

- We have sucessfully implemented the random forest machine learning model on a given network data set. Which predict the types of protocal with an accuracy of 98 percentage.

- **PREVIEW** CLICK ON PREVIEW TO SEE IMPLEMENTATION

GROUP MEMBERS

Anil Kumar

Kunal

Ravi Rajesh Keer

Sandeep Kumar