Machine Learning Project Report
The Boys

Project Name - Yahoo Troll Question

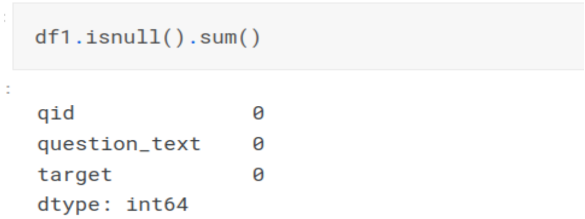
Team Members :-
IMT2020090 - Balaji Sankapal
IMT2020100 - Teja Janaki Ram

Problem Statement

From given data we have to predict whether certain questions will be classified into SPAM/Troll questions or not.

Preprocessing of Data

- No null values



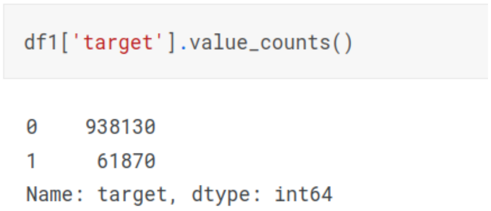
```
df1.isnull().sum()
```

```
qid              0
question_text    0
target           0
dtype: int64
```

- No duplicate data entries

Exploratory Data Analysis

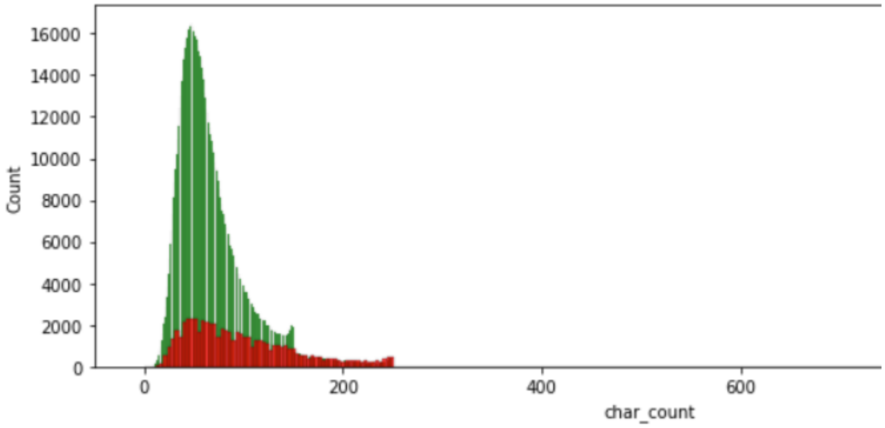
- Checking if training data is imbalanced



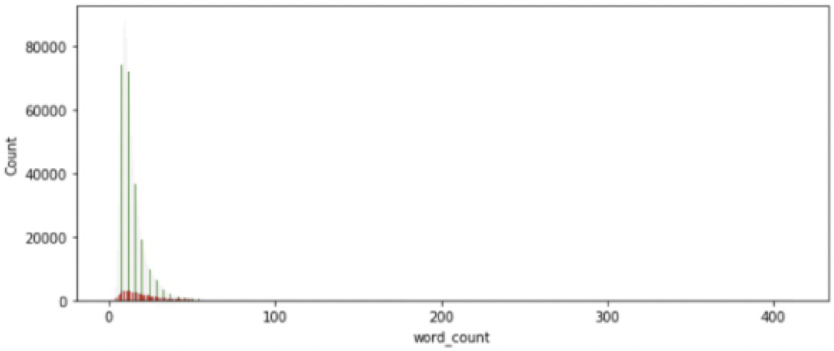
```
df1['target'].value_counts()
```

```
0    938130
1     61870
Name: target, dtype: int64
```

- Exploring character/word length relation to classification - Distribution of Number of words, characters and sentences in Spam(Red) and Ham(Green).



```
<AxesSubplot:xlabel='word_count', ylabel='Count'>
```



Preprocessing (NLP)

Tried the following methods of the NLP Toolkit:

- Lower case
- Tokenization
- Removing stop words
- Removing everything except alphanumeric characters
- Lemmatization

But had to remove these preprocessing steps because of the following reasons:

- Normally, people tend to make “typos” while web surfing.
- People tend to type in capitals when they are angry or in a rush.
- Good results after removing them

TfIdf vectorizer with the following hyperparameters is used:

- Strip_accents = ‘unicode’
- Analyzer = ‘word’/ ‘char’(Both vectorizers horizontally stacked)
- Ngram_range = (1,3)
- Max_df = 0.5
- Max_features = 10000

TfIdf gives better results than CountVectorizer.

Models applied and accuracy

- Gaussian Naive Bayes (Not used):
  - Needs ‘dense’ data as input but TfIdf gives a sparse matrix as input.
- Multinomial Naive Bayes (Not used):
  - Doesn’t give the best results
- Random Forest Classifier (Not used):
  - Doesn’t give good results
- Logistic Regression:
  - Dual - False
  - Class weights - {0:0.23,1:0.77}

Ensemble Methods

These models didn’t give good results:

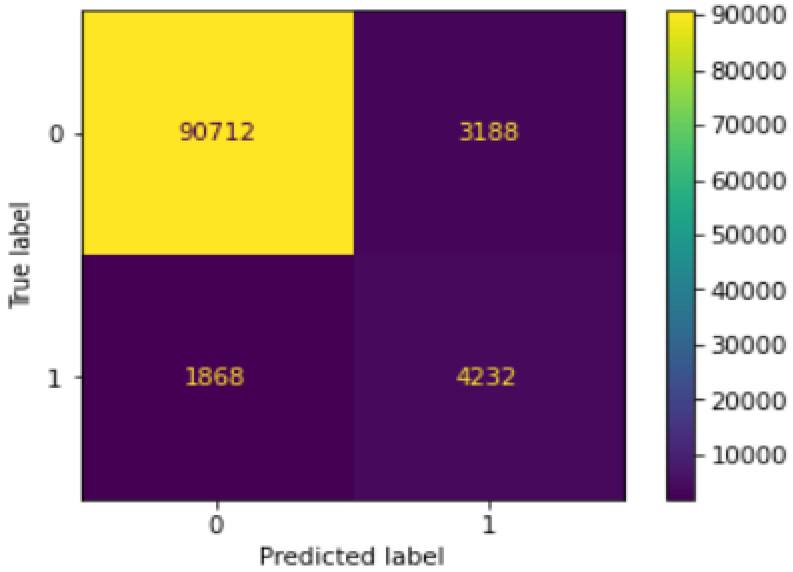
- Bagging - We applied Bootstrap Aggregation with Decision Tree Classifier with a random state.

- Stacking - Base models we used are Gaussian, Multinomial Naive Bayes and logistic regression.

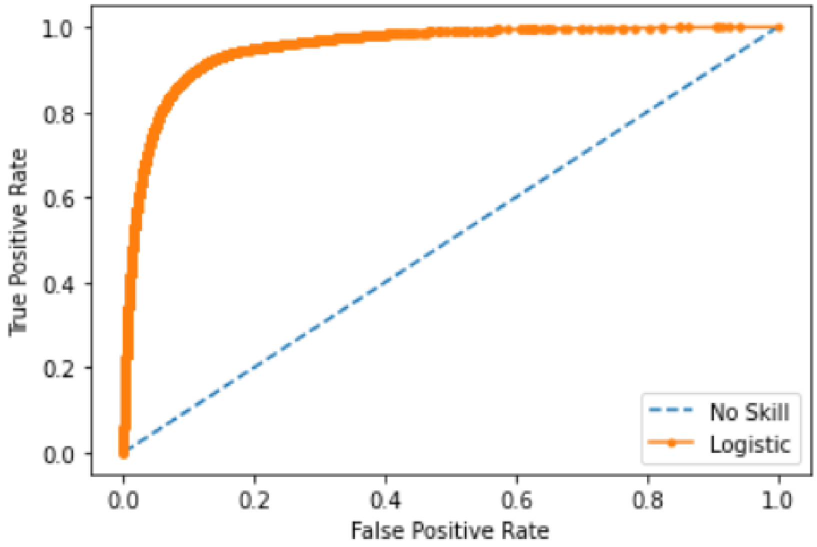
- Boosting - XGBoost and AdaBoost

Model analysis

For the logistic regression model, following is the confusion matrix:



This tells us that there are 3188 false positives and 1868 false negatives. The following is the ROC curve for the same:



- Tried to find the optimum threshold for logistic regression, instead of the default 0.5, with the above information.