Exploratory Data Analysis

Hate Speech Detection Using Transformers

Team - 8bit(Prajesh Tejani & Jigar Borad)

15-August-2022

Agenda

Executive Summary

Problem Statement

Approach

EDA

EDA Summary

Recommendations

Executive Summary

- As a part of final internship project, we want to create a machine learning model which can classify the tweets in two parts whether it is hateful tweet or not.

Problem Statement

- Objective : The term hate speech is understood as any type of verbal, written or behavioural communication that attacks or uses derogatory or discriminatory language against a person or group based on what they are, in other words, based on their religion, ethnicity, nationality, race, colour, ancestry, sex or another identity factor. In this problem, We will take you through a hate speech detection model with Machine Learning and Python. But, before that we are going to do analysis of data and based on analysis we will modify for better performance of our model.

Approach

- Data Understanding
- Data Cleaning and manipulation
- Analysing and visualizing data
- Recommendations of machine learning models

Data Exploration

- 2 datasets: Training and Test
- 3 features in total in training data with 2 input features and 1 target
- 2 features in total in test data with 2 input features
- Total Data:

  1) training dataset : 31962

  2) test dataset : 17197

Data Analysis

Finding Empty data in dataset

- in Training Data

```
id       0
label    0
tweet    0
dtype: int64
```

- in Test Data

```
id       0
tweet    0
dtype: int64
```

- So, here as you can see there is no empty data in our both datasets. So, we do not need to fill up that space.

Data Modification

Removing unnecessary words

- In our datasets we have some unnecessary words such as user names , special characters , website links or numbers which do not have any impact in predictions. So, we need to remove that to create better and faster machine learning model.

Data Modification

Removing Duplicate Data (I)

- First we are going to find that whether we have any duplicate data(tweets) or not. And, in below picture we can see tweet and occurrence of that tweet in whole data.

```
[('model love u take u time ur', 325),
 ('final found way delet old tweet might find use well deletetweet', 83),
 ('aww yeah good bing bong bing bong', 75),
 ('might libtard libtard sjw liber polit', 72),
 ('grate affirm', 57),
 ('love instagood photooftheday top tag tbt cute beauti followm follow', 36),
 ('happi work confer right mindset lead cultur develop organ work mindset',
  35),
 ('father day', 32),
 ('lighttherapi help depress altwaystoh healthi happi', 31),
 ('', 31),
 ('lover stop angri visit us gt gt gt lover friend astrolog love', 26),
 ('best essentialoil anxieti healthi peac altwaystoh', 26),
 ('sikh templ vandalis calgari wso condemn act', 26),
 ('lighttherapi help sad depress altwaystoh healthi', 24),
 ('black amp feel like stomp listen retweet tampa miami', 23),
 ('flagday2016 flag day 2016 30 photo buy thing flag day 2016', 22),
 ('get get get enjoy music today free app free music', 21),
 ('feminismiscanc feminismisterror feminismmuktbharat malevot ignor', 20),
 ('sea shepherd suppoer racist antirac seashepherd', 17),
 ('save login x broker chang meme love educ univers', 17),
 ('magnettherapi realli work altwaystoh heal healthi', 15),
 ('detoxdiet altwaystoh healthi', 15),
```

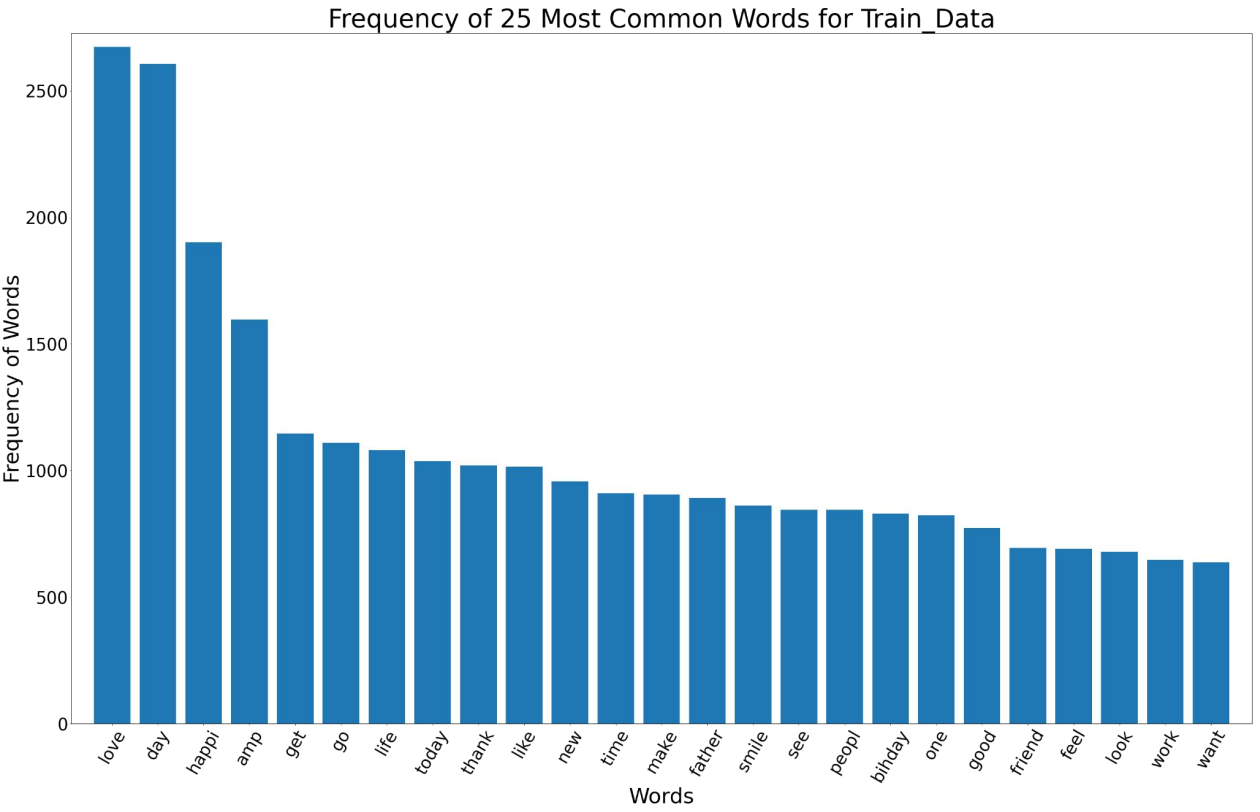
Data Modification

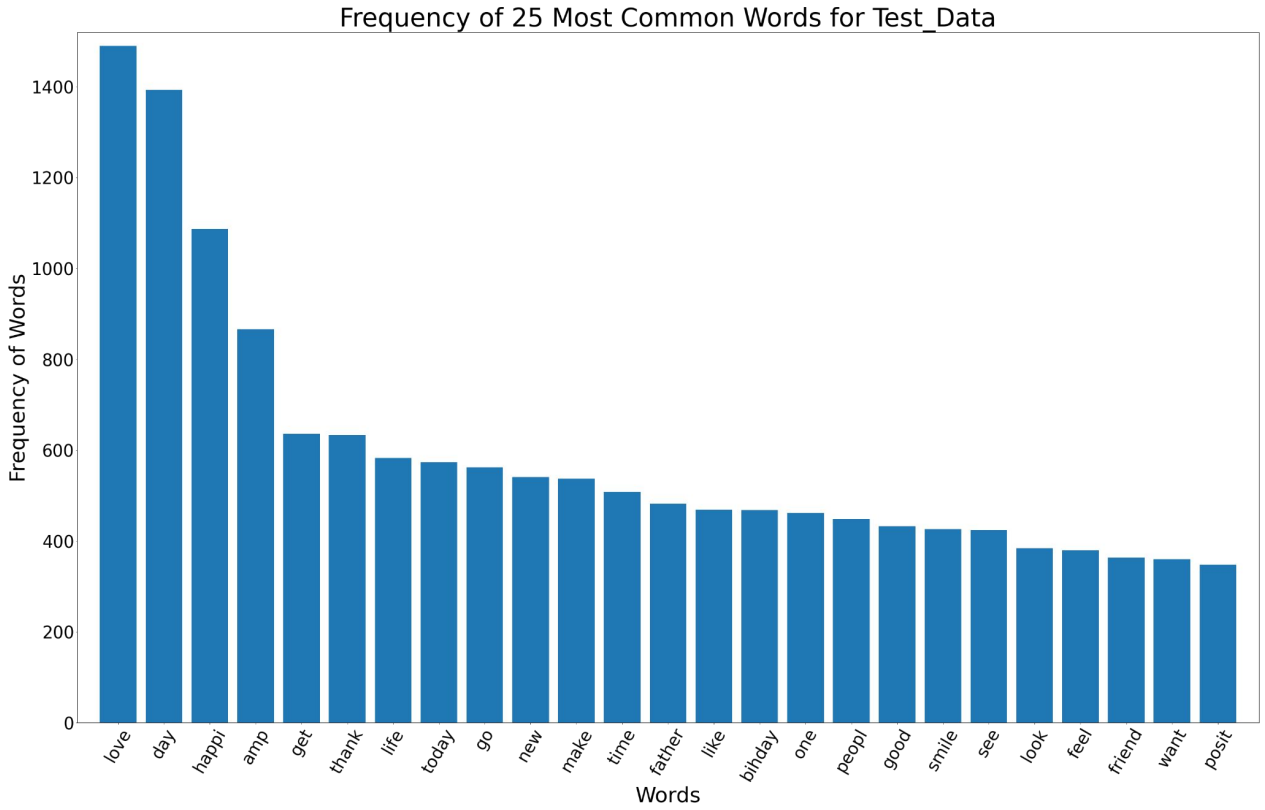
Removing Duplicate Data (II)

- Now , you can see below the result after removing duplicate data(tweets). Every tweet has only 1 occurrence in dataset.

```
[('father dysfunct selfish drag kid dysfunct run', 1),
 ('thank lyft credit use caus offer wheelchair van pdx disapoint getthank', 1),
 ('bihday majesti', 1),
 ('model love u take u time ur', 1),
 ('factsguid societi motiv', 1),
 ('2 2 huge fan fare big talk leav chao pay disput get allshowandnogo', 1),
 ('camp tomorrow danni', 1),
 ('next school year year exam think school exam hate imagin actorslif revolutionschool girl',
  1),
 ('love land allin cav champion cleveland clevelandcavali', 1),
 ('welcom gr8', 1),
 ('ireland consum price index mom climb previou 0 2 0 5 may blog silver gold forex',
  1),
 ('selfish orlando standwithorlando pulseshoot orlandoshoot biggerproblem selfish heabreak valu love',
  1),
 ('get see daddi today 80day gettingf', 1),
 ('cnn call michigan middl school build wall chant tcot', 1),
 ('comment australia opkillingbay seashepherd helpcovedolphin thecov helpcovedolphin',
  1),
 ('ouch junior angri got7 junior yugyoem omg', 1),
 ('thank paner thank posit', 1),
 ('retweet agre', 1),
 ('friday smile around via ig user cooki make peopl', 1),
```

Frequency of word in data



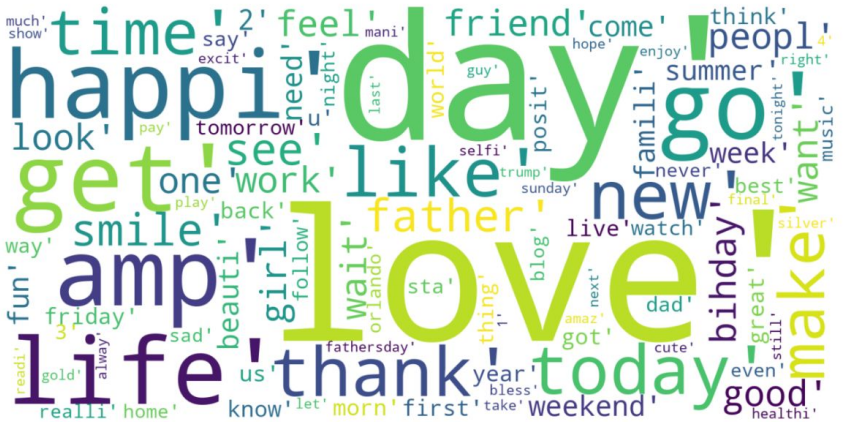


- In above graph we can see the 25 most common words in Training dataset and Test Dataset.
- We can see that both the datasets have almost same most common words. For this reason we might get accurately trained model through which we can get high accuracy while testing and predicting results.

Frequency of word in data

Top 100 Most Common Words for Test_Data

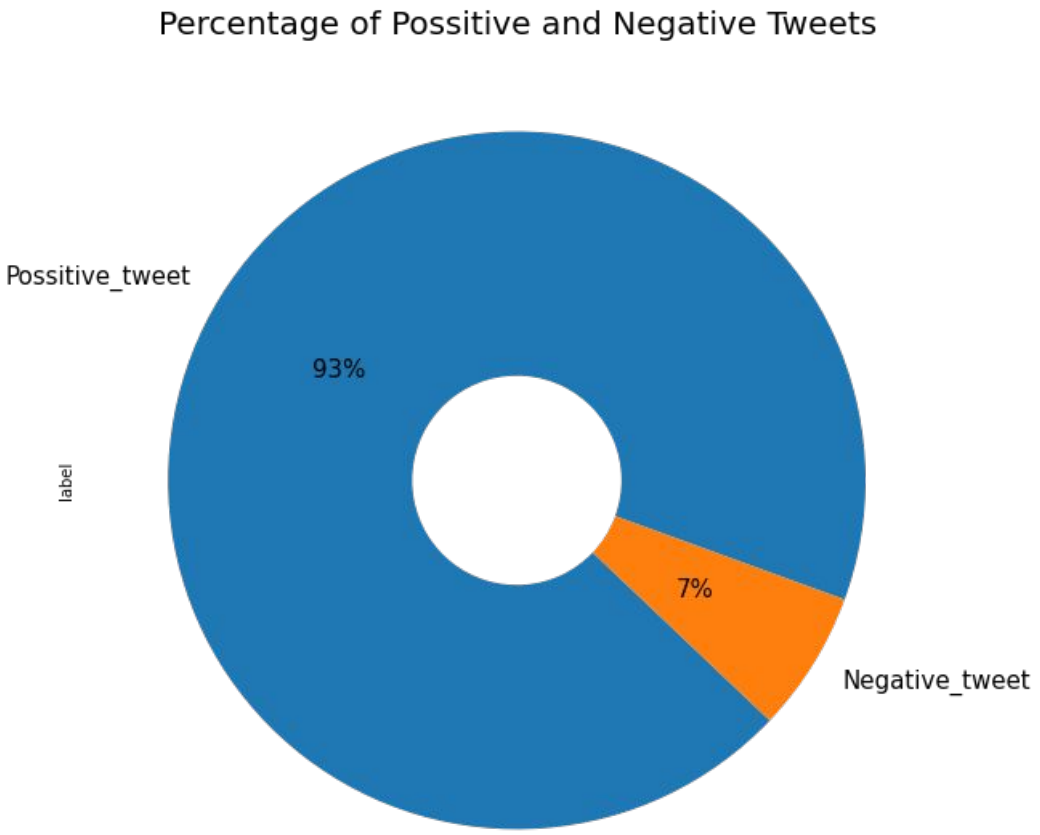
Top 100 Most Common Words for Train_Data



- Here we can 100 most common words in picture. Bigger the word most common the word.

Data Exploration and visualization

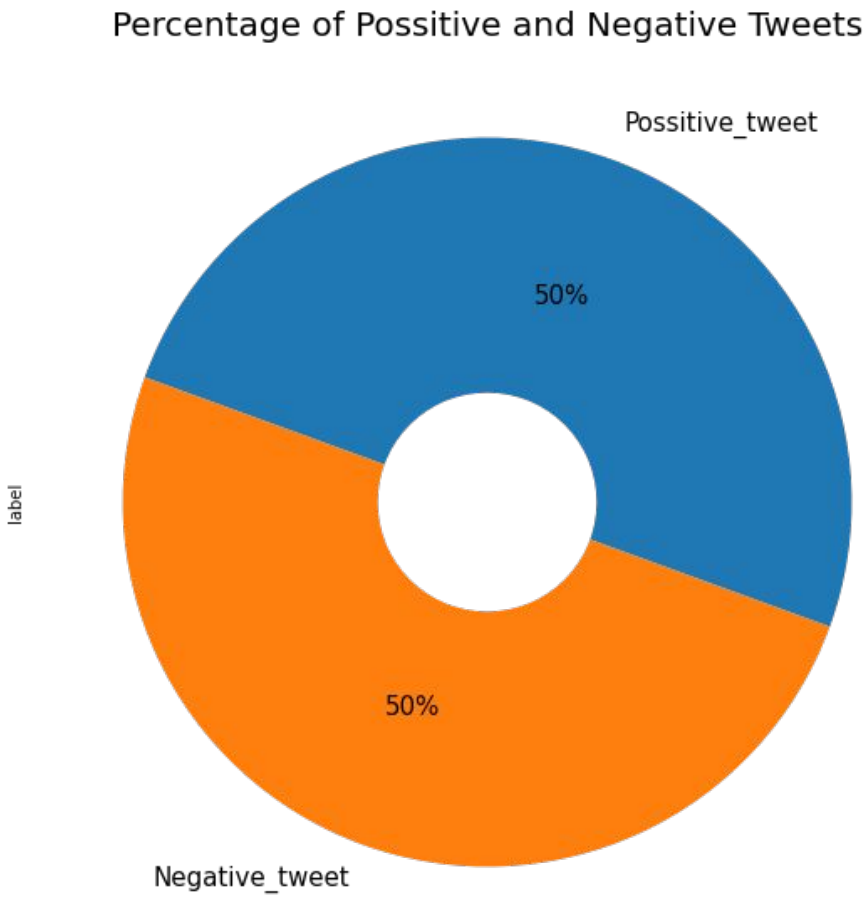
- **Distribution ratio of positive and negative tweets in training data**



- From above chart we can see that we have extremely high number of positive(Non hateful) tweets. This can cause problem in prediction of our test data. Model trained with this dataset will be heavily biased to positive tweets means in most of the cases our model will predict that is is positive tweet even if it is not. So, we have make it 1:1 ratio with sampling method.
- In sampling there is upsampling method in which it will generate data from our old dataset for e.g Here we have less negative tweets so it will create negative tweets to match number of positive tweets.

Data Exploration and visualization

- **Distribution ratio of positive and negative tweets in training data**



- After upsampling this is the result where we have equal number of positive and negative tweets.

Recommendation and Analysis

- Recommended Machine Learning Models:
  - From analysing datasets we can clearly see that we have to **classify** that whether tweet is positive or negative. So, we have to use modern and advance classification machine learning models such as XGboost, Stochastic Gradient descent , Decision Tree , Random Forest Model etc.
- Other Recommendation:
  - But , before that we need to create bag of word model with count vectoriser or TV-IDF vectoriser from scikit learn library and transformer with TV-IDF transformer. It will generate bag of word and transform it into 1s and 0s data which is understandable for ML models.
  - Also, for testing the or finding accuracy of model split the training data into train and test data with the help of scikit learn library(train_test_split() method).
  - I would also like recommend that to make code more smaller we can use pipelines as well in which we can do whole above mentioned process with one line of code.

Thank You