WHY SO HARSH?

SUBMITTED BY:

MT2022130- VARTIKA CHATURVEDI

MT2022137- YASHA DAYAL

Mounting the Google Drive

```
from google.colab import drive
drive.mount('/content/drive')
```

```
     Mounted at /content/drive
```

Reading the Dataset

```
#importing the necessary libraries
import pandas as pd
import numpy as np
```

```
#Reading the data into a dataframe
df = pd.read_csv("/content/drive/MyDrive/train.csv")
```

Exploring the Dataset

```
df.shape
```

```
     (89359, 8)
```

The dataset has 89359 rows and 8 columns

```
df.info()
```

```
     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 89359 entries, 0 to 89358
     Data columns (total 8 columns):
      #   Column           Non-Null Count  Dtype 
     ---  ------           --------------  ----- 
      0   id               89359 non-null  object
      1   text             89359 non-null  object
      2   harsh            89359 non-null  int64 
      3   extremely_harsh  89359 non-null  int64 
      4   vulgar           89359 non-null  int64 
      5   threatening      89359 non-null  int64 
      6   disrespect       89359 non-null  int64 
      7   targeted_hate    89359 non-null  int64 
```

```
dtypes: int64(6), object(2)
memory usage: 5.5+ MB
```

No null entries are present in the dataset

```
df.describe()
```

	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
count	89359.000000	89359.000000	89359.000000	89359.000000	89359.000000	89359.000000
mean	0.095782	0.010262	0.053067	0.002999	0.049150	0.008975
std	0.294294	0.100781	0.224168	0.054683	0.216182	0.094311
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

```
df.corr()
```

	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
harsh	1.000000	0.312860	0.677991	0.156696	0.645257	0.271428
extremely_harsh	0.312860	1.000000	0.409329	0.134532	0.378011	0.206952
vulgar	0.677991	0.409329	1.000000	0.146781	0.736406	0.286603
threatening	0.156696	0.134532	0.146781	1.000000	0.158877	0.114129
disrespect	0.645257	0.378011	0.736406	0.158877	1.000000	0.343374
targeted_hate	0.271428	0.206952	0.286603	0.114129	0.343374	1.000000

```
df.head()
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	", 6 December 2007 (UTC)\nI am interested, not...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	I added about three missing parameters to temp...	0	0	0	0	0	0
2	db934381501872ba6f38	SANDBOX?? \n\nI DID YOUR MADRE DID IN THE SANDBOX	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	why good sir? Why? \n\nYou, sir, obviously do ...	1	0	1	1	1	0

```
import nltk
```

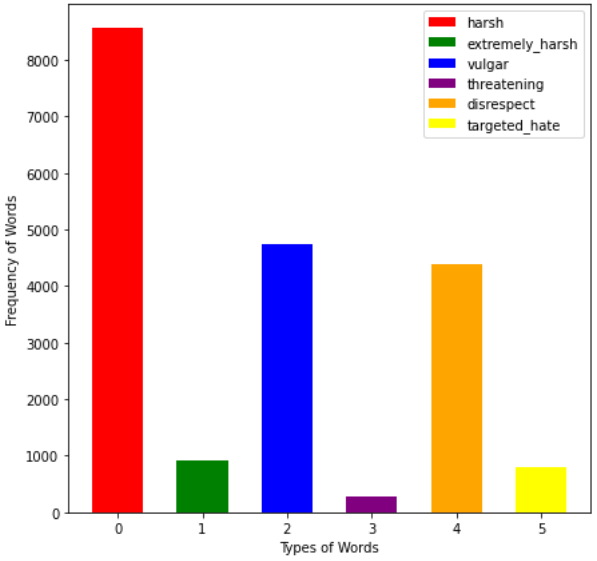
```
nltk.download('all')
```

```
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping corpora/product_reviews_2.zip.
[nltk_data]    | Downloading package propbank to /root/nltk_data...
[nltk_data]    | Downloading package pros_cons to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/pros_cons.zip.
[nltk_data]    | Downloading package ptb to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/ptb.zip.
[nltk_data]    | Downloading package punkt to /root/nltk_data...
[nltk_data]    |   Unzipping tokenizers/punkt.zip.
[nltk_data]    | Downloading package qc to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/qc.zip.
[nltk_data]    | Downloading package reuters to /root/nltk_data...
[nltk_data]    | Downloading package rslp to /root/nltk_data...
[nltk_data]    |   Unzipping stemmers/rslp.zip.
[nltk_data]    | Downloading package rte to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/rte.zip.
[nltk_data]    | Downloading package sample_grammars to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping grammars/sample_grammars.zip.
[nltk_data]    | Downloading package semcor to /root/nltk_data...
[nltk_data]    | Downloading package senseval to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/senseval.zip.
[nltk_data]    | Downloading package sentence_polarity to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping corpora/sentence_polarity.zip.
[nltk_data]    | Downloading package sentiwordnet to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping corpora/sentiwordnet.zip.
[nltk_data]    | Downloading package shakespeare to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/shakespeare.zip.
[nltk_data]    | Downloading package sinica_treebank to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping corpora/sinica_treebank.zip.
[nltk_data]    | Downloading package smultron to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/smultron.zip.
[nltk_data]    | Downloading package snowball_data to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    | Downloading package spanish_grammars to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping grammars/spanish_grammars.zip.
[nltk_data]    | Downloading package state_union to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/state_union.zip.
[nltk_data]    | Downloading package stopwords to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/stopwords.zip.
[nltk_data]    | Downloading package subjectivity to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Unzipping corpora/subjectivity.zip.
[nltk_data]    | Downloading package swadesh to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/swadesh.zip.
[nltk_data]    | Downloading package switchboard to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/switchboard.zip.
[nltk_data]    | Downloading package tagsets to /root/nltk_data...
[nltk_data]    |   Unzipping help/tagsets.zip.
[nltk_data]    | Downloading package timit to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/timit.zip.
[nltk_data]    | Downloading package toolbox to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/toolbox.zip.
[nltk_data]    | Downloading package treebank to /root/nltk_data...
[nltk_data]    |   Unzipping corpora/treebank.zip.
```

```
import matplotlib.pyplot as plt


count=df['text'].value_counts()
#Creating a function to plot the counts using matplotlib
def plot_counts(count_harsh,count_extremlyharsh,count_vulgar,count_threatening,count_disrepect,count_targetedhate):
    plt.rcParams['figure.figsize']=(7,7)
    plt.bar(0,count_harsh,width=0.6,label='harsh',color='red')
    plt.legend()
    plt.bar(1,count_extremlyharsh,width=0.6,label='extremely_harsh',color='green')
    plt.legend()
    plt.bar(2,count_vulgar,width=0.6,label='vulgar',color='blue')
    plt.legend()
    plt.bar(3,count_threatening,width=0.6,label='threatening',color='purple')
    plt.legend()
    plt.bar(4,count_disrepect,width=0.6,label='disrespect',color='orange')
    plt.legend()
    plt.bar(5,count_targetedhate,width=0.6,label='targeted_hate',color='yellow')
    plt.legend()
    plt.ylabel('Frequency of Words')
    plt.xlabel('Types of Words')
    plt.show()

count_harsh=df[df['harsh']== 1]
count_extremlyharsh=df[df['extremely_harsh']== 1]
count_vulgar=df[df['vulgar']== 1]
count_threatening=df[df['threatening']== 1]
count_disrepect=df[df['disrespect']== 1]
count_targetedhate=df[df['targeted_hate']== 1]
plot_counts(len(count_harsh),len(count_extremlyharsh),len(count_vulgar),len(count_threatening),len(count_disrepect),len(count_targetedhate))
```



Cleaning the Dataset to gain maximum information out of it

Removing the punctuation marks from the dataset

```
df['text'] = df['text'].str.replace(r'[^\w\s]+', '') #w = word and #s = space
```

```
    /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: The default value of regex will change from True to False in a future version.
      """Entry point for launching an IPython kernel.
```

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	6 December 2007 UTC\nI am interested not in a...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	I added about three missing parameters to temp...	0	0	0	0	0	0
2	db934381501872ba6f38	SANDBOX \n\nI DID YOUR MADRE DID IN THE SANDBOX	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	why good sir Why \n\nYou sir obviously do not ...	1	0	1	1	1	0
4	b18f26cfa1408b52e949	\n\n Source \n\nIncase I forget or someone els...	0	0	0	0	0	0
5	6729341b01ab895388d7	\n Neither of your arguments are persuasive Y...	0	0	0	0	0	0
6	a36cf2a3d3cf833492ec	I knew this was a left wing blog and the above...	0	0	0	0	0	0

Removing some special characters like \n

```
df['text'] = df['text'].str.replace('\n', '')
```

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	6 December 2007 UTCI am interested not in arg...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	I added about three missing parameters to temp	0	0	0	0	0	0

Converting the text data into lowercase

DID IN THE SANDBOX

```
#lowercase is done to not differentiate between He and he. Treat them as same
df['text'] = df['text'].str.lower()
```

Source incase I forget or someone

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	6 december 2007 utci am interested not in arg...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	i added about three missing parameters to temp...	0	0	0	0	0	0
2	db934381501872ba6f38	sandbox i did your madre did in the sandbox	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	why good sir why you sir obviously do not comp...	1	0	1	1	1	0
4	b18f26cfa1408b52e949	source incase i forget or someone else wants ...	0	0	0	0	0	0
5	6729341b01ab895388d7	neither of your arguments are persuasive you...	0	0	0	0	0	0
6	a36cf2a3d3cf833492ec	i knew this was a left wing blog and the above	0	0	0	0	0	0

Removing the numerical data

```
df['text'] = df['text'].str.replace('\d+', '') #d = digits
```

```
    /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: The default value of regex will change from True to False in a future version.
      """Entry point for launching an IPython kernel.
```

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	december utci am interested not in arguing ...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	i added about three missing parameters to temp...	0	0	0	0	0	0
2	db934381501872ba6f38	sandbox i did your madre did in the	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	why good sir why you sir obviously do not comp	1	0	1	1	1	0
4	b18f26cfa1408b52e949	else wants	0	0	0	0	0	0
6	a36cf2a3d3cf833492ec	i knew this was a left wing blog and	0	0	0	0	0	0

Removing the repeated characters

```
import re
```

```
def solve(s):
     return re.sub(r'(\S)\1+', r'\1', s)
df['text'] = df['text'].apply(lambda x : solve(x))
```

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	december utci am interested not in arguing ...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	i aded about thre mising parameters to templat...	0	0	0	0	0	0
2	db934381501872ba6f38	sandbox i did your madre did in the sandbox	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	why god sir why you sir obviously do not compr...	1	0	1	1	1	0
4	b18f26cfa1408b52e949	source incase i forget or someone else wants ...	0	0	0	0	0	0
5	6729341b01ab895388d7	neither of your arguments are persuasive you...	0	0	0	0	0	0
6	a36cf2a3d3cf833492ec	i knew this was a left wing blog and the above	0	0	0	0	0	0

Removing the stop words

```
import nltk
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
```

```
import nltk
nltk.download('stopwords')
```

```
     [nltk_data] Downloading package stopwords to /root/nltk_data...
     [nltk_data]   Package stopwords is already up-to-date!
```

```
      True
```

```
new_stopwords = ["january","february","march","april","may","june","july","august","september","october","november","december","also","zero","one","two","thre","four","five","six","seven","eight","nine
stpwrd = nltk.corpus.stopwords.words('english')
stpwrd.extend(new_stopwords)
```

```
stpwrd
```

```
        'three',
       'four',
       'five',
       'six',
       'seven',
       'eight',
       'nine']

#removing stop words
stop_words = set(stpwrd)
def remove_stop(x) :
    return " ".join([word for word in str(x).split() if word not in stop_words]) #splitting on the basis of " " and then joined with " "
df['text'] = df['text'].apply(lambda x : remove_stop(x))#apply lambda for each row-> sending each sentence to remove_stop
```

```
import nltk
from nltk.stem import WordNetLemmatizer

# Init the Wordnet Lemmatizer
lemmatizer = WordNetLemmatizer()
```

```
nltk.download('wordnet')
nltk.download('omw-1.4')
```

```
     [nltk_data] Downloading package wordnet to /root/nltk_data...
     [nltk_data]   Package wordnet is already up-to-date!
     [nltk_data] Downloading package omw-1.4 to /root/nltk_data...
     [nltk_data]   Package omw-1.4 is already up-to-date!
     True
```

```
#checking lemmatization
def lemmatize(text):
  output = ""
  text = text.split(" ")
  for word in text :
    word1 = wordnetlemmatizer.lemmatize(word,pos="n")
    word2 = wordnetlemmatizer.lemmatize(word1,pos="v")
    word3 = wordnetlemmatizer.lemmatize(word2,pos="a")
    word4 = wordnetlemmatizer.lemmatize(word3,pos="r")
    output = output + " "+word4

  return str(output.strip())
```

```
df['text'] = df['text'].apply(lambda x : lemmatize(x))
```

```
#lemmatization
#stop_words = set(stpwrd)
#def remove_stop(x) :
#    return " ".join([lemmatizer.lemmatize(word) for word in str(x).split() if word not in stop_words]) #splitting on the basis of " " and then joined with " "
##apply lambda for each row-> sending each sentence to remove_stop
```

```
df.head(10)
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	utci interest argue policy resolve ongoing con...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	aded mising parameter templateinfobox organiza...	0	0	0	0	0	0
2	db934381501872ba6f38	sandbox madre sandbox	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	god sir sir obviously comprehend importance sc...	1	0	1	1	1	0
4	b18f26cfa1408b52e949	source incase forget someone else want pick gr...	0	0	0	0	0	0
5	6729341b01ab895388d7	neither argument persuasive dismis separate ar...	0	0	0	0	0	0
		know leave wing blog statement prof						

```
dfs=df
```

```
import nltk
from nltk.corpus import wordnet
```

List of unique words in dataset

```
from sklearn.feature_extraction.text import CountVectorizer
```

```
vect = CountVectorizer()
vect.fit(df['text'])
print("Dictionary has no.of unique words: ", len(vect.vocabulary_))
```

```
    Dictionary has no.of unique words:  172102
```

```
#converting dictionary to list
list = [(k, v) for k, v in vect.vocabulary_.items()]
```

```
list.sort()
list
```

```
 ('_many_', 31),
 ('_maximum_', 32),
 ('_miley_cyrus_', 33),
 ('_minerals_', 34),
 ('_monitor', 35),
 ('_n_w_regionie_typecity', 36),
 ('_nevermind', 37),
 ('_ninety', 38),
 ('_noeditsection_', 39),
 ('_noeditsection_nonewsectionlink_narayana', 40),
 ('_not_', 41),
 ('_notoc_', 42),
 ('_o', 43),
 ('_one_', 44),
 ('_only_', 45),
 ('_philipe', 46),
 ('_please', 47),
 ('_ps', 48),
 ('_px', 49),
 ('_reasons_', 50),
 ('_require_', 51),
 ('_sanka', 52),
 ('_should_', 53),
 ('_sobok', 54),
 ('_suposed_', 55),
 ('_thanks', 56),
 ('_that', 57),
 ('_then_', 58),
 ('_toc_', 59),
 ('_toc_deleted', 60),
 ('_toc_totoro', 61),
 ('_user', 62),
 ('_vitaines', 63),
 ('_war_', 64),
 ('_wikia_', 65),
 ('_yes', 66),
 ('_you', 67),
 ('_you_', 68),
 ('_youre', 69),
 ('_zero', 70),
 ('a_cardboard_microwave', 71),
 ('a_hero_sits_next_dor', 72),
 ('a_holemothr', 73),
 ('a_picture_is_worth_a_bucks', 74)
```

```
#frequency of words
vector = vect.transform(df['text'])
print(vector)
#in the tuple, the first element is number of rows i.e., 89359 called training examples and the second element is the index of the word
```

```
  (0, 2429)	1
  (0, 3093)	1
  (0, 5093)	1
  (0, 6391)	2
  (0, 8550)	1
  (0, 9034)	1
  (0, 19769)	1
  (0, 20823)	1
  (0, 25970)	1
  (0, 26473)	1
  (0, 29821)	1
  (0, 35727)	1
  (0, 37114)	1
  (0, 37904)	1
  (0, 38054)	1
  (0, 39375)	1
  (0, 41652)	1
  (0, 42581)	1
  (0, 43767)	1
  (0, 56956)	1
  (0, 65019)	1
  (0, 69283)	1
  (0, 73198)	1
  (0, 82109)	1
  (0, 93721)	3
  :	:
  (89356, 83977)	1
  (89356, 87289)	1
  (89356, 90609)	1
  (89356, 105853)	1
  (89356, 106362)	5
  (89356, 132034)	1
  (89356, 160409)	1
  (89356, 163385)	1
  (89356, 163840)	1
  (89356, 163880)	2
  (89356, 163883)	1
  (89356, 166146)	1
  (89356, 166152)	1
  (89357, 36393)	1
  (89357, 85132)	2
  (89357, 95873)	1
  (89357, 120039)	1
  (89357, 154736)	1
  (89358, 12257)	1
  (89358, 26202)	1
  (89358, 52822)	1
  (89358, 97521)	1
  (89358, 128739)	1
  (89358, 162647)	1
  (89358, 167764)	1
```

Pickling

```
#to avoid doing pre-processing multiple times and thus saving RAM for further tasks
import pickle
```

```
filename = 'train.pkl'
```

```
#Run this cell only one time... and from the next time comment this cell and only import pickle and upload the pkl file, no need to run above cells
pickle.dump(df,open(filename,'wb'))
```

```
main_df = pickle.load(open(filename,'rb'))
```

```
main_df
```

	id	text	harsh	extremely_harsh	vulgar	threatening	disrespect	targeted_hate
0	a8be7c5d4527adbbf15f	utci interest argue policy resolve ongoing con...	0	0	0	0	0	0
1	0b7ca73f388222aad64d	aded mising parameter templateinfobox organiza...	0	0	0	0	0	0
2	db934381501872ba6f38	sandbox madre sandbox	1	0	0	0	0	0
3	228015c4a87c4b1f09a7	god sir sir obviously comprehend importance sc...	1	0	1	1	1	0
4	b18f26cfa1408b52e949	source incase forget someone else want pick gr...	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...
89354	748a13233c1ea91c4584	becuase critic actualy read boks	0	0	0	0	0	0
89355	e49b832cc766ee220113	youre go technical boyd never post goglegroups...	0	0	0	0	0	0
89356	ff4751b348157ac2b585	join u fb helo pakistani	0	0	0	0	0	0

Pre-processing of Test Dataset

```
from google.colab import drive
drive.mount('/content/drive')
```

```
    Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

Reading the Dataset

```
#importing the necessary libraries
import pandas as pd
import numpy as np
```

```
#Reading the data into a dataframe
tdf = pd.read_csv("/content/drive/MyDrive/test.csv")
```

Exploring the Dataset

```
tdf.shape
```

```
    (38297, 2)
```

The dataset has 38297 rows and 2 columns

```
tdf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38297 entries, 0 to 38296
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   id      38297 non-null  object
 1   text    38297 non-null  object
dtypes: object(2)
memory usage: 598.5+ KB
```

No null entries are present in the dataset

```
tdf.describe()
```

	id	text
count	38297	38297
unique	38297	38297
top	e0ae9d9474a5689a5791	in an interview before his execution
freq	1	1

```
tdf.head()
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	He knew what he was doing. The below posts are...
2	5e1953d9ae04bdc66408	Zzzzzzz... youre a real bore. Now go bore some...
3	23128f98196c8e8f7b90	"\n\nYet, it remains confusion because the 910...
4	2d3f1254f71472bf2b78	I was referring to them losing interest in van...

Cleaning the Dataset to gain maximum information out of it

Removing the punctuation marks from the dataset

```
tdf['text'] = tdf['text'].str.replace(r'[^\w\s]+', '')
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: The default value of regex will change from True to False in a future version.
  """Entry point for launching an IPython kernel.
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	He knew what he was doing The below posts are ...
2	5e1953d9ae04bdc66408	Zzzzzzz youre a real bore Now go bore someone ...
3	23128f98196c8e8f7b90	\n\nYet it remains confusion because the 910 i...
4	2d3f1254f71472bf2b78	I was referring to them losing interest in van...
5	21f4f0f4812a08ea6c28	5 March 2009 UTC\n\nThat wasnt an attack ad h...
6	733b43d534c67c1be948	1 Youre not reading properly Ive asked you wha...
7	aad47a397f7ddc629d5d	\nplease look at the discussion here and here ...
8	d19fcde8a3af2e472d74	2011 UTC\nCall Of Duty has never made any clai...
9	7d4de482c60f1c8a79c6	You too man take care

Removing some special characters like \n

```
tdf['text'] = tdf['text'].str.replace('\n', '')
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	He knew what he was doing The below posts are ...
2	5e1953d9ae04bdc66408	Zzzzzzz youre a real bore Now go bore someone ...
3	23128f98196c8e8f7b90	Yet it remains confusion because the 910 is ju...
4	2d3f1254f71472bf2b78	I was referring to them losing interest in van...
5	21f4f0f4812a08ea6c28	5 March 2009 UTCThat wasnt an attack ad homin...
6	733b43d534c67c1be948	1 Youre not reading properly Ive asked you wha...
7	aad47a397f7ddc629d5d	please look at the discussion here and here Ly...
8	d19fcde8a3af2e472d74	2011 UTCCall Of Duty has never made any claims...
9	7d4de482c60f1c8a79c6	You too man take care

Converting the text data into lowercase

```
tdf['text'] = tdf['text'].str.lower()
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	he knew what he was doing the below posts are ...
2	5e1953d9ae04bdc66408	zzzzzzz youre a real bore now go bore someone ...
3	23128f98196c8e8f7b90	yet it remains confusion because the 910 is ju...
4	2d3f1254f71472bf2b78	i was referring to them losing interest in van...
5	21f4f0f4812a08ea6c28	5 march 2009 utcthat wasnt an attack ad homin...
6	733b43d534c67c1be948	1 youre not reading properly ive asked you wha...
7	aad47a397f7ddc629d5d	please look at the discussion here and here ly...
8	d19fcde8a3af2e472d74	2011 utccall of duty has never made any claims

Removing the numerical data

```
tdf['text'] = tdf['text'].str.replace('\d+', '')
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: The default value of regex will change from True to False in a future version.
  """Entry point for launching an IPython kernel.
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	he knew what he was doing the below posts are ...
2	5e1953d9ae04bdc66408	zzzzzzz youre a real bore now go bore someone ...
3	23128f98196c8e8f7b90	yet it remains confusion because the is just ...
4	2d3f1254f71472bf2b78	i was referring to them losing interest in van...
5	21f4f0f4812a08ea6c28	march utcthat wasnt an attack ad hominem th...
6	733b43d534c67c1be948	youre not reading properly ive asked you what...
7	aad47a397f7ddc629d5d	please look at the discussion here and here ly...
8	d19fcde8a3af2e472d74	utccall of duty has never made any claims of ...
9	7d4de482c60f1c8a79c6	you too man take care

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	in an interview before his execution
1	b64a191301cad4f11287	he knew what he was doing the below posts are ...
2	5e1953d9ae04bdc66408	zzzzzzz youre a real bore now go bore someone ...
3	23128f98196c8e8f7b90	yet it remains confusion because the is just ...
4	2d3f1254f71472bf2b78	i was referring to them losing interest in van...
5	21f4f0f4812a08ea6c28	march utcthat wasnt an attack ad hominem th...

Removing the stop words

8 d19fcde8a3af2e472d74 utccall of duty has never made any claims of

```
#removing stop words
stop_words = set(stpwrd)
def remove_stop(x) :
    return " ".join([word for word in str(x).split() if word not in stop_words]) #splitting on the basis of " " and then joined with " "
tdf['text'] = tdf['text'].apply(lambda x : remove_stop(x))#apply lambda for each row-> sending each sentence to remove_stop
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	interview execution
1	b64a191301cad4f11287	knew posts truthful admins hate wont anything ...
2	5e1953d9ae04bdc66408	zzzzzzz youre real bore go bore someone else twt
3	23128f98196c8e8f7b90	yet remains confusion mentioned sac withdrew b...
4	2d3f1254f71472bf2b78	referring losing interest vandalising talk pag...
5	21f4f0f4812a08ea6c28	utcthat wasnt attack ad hominem constructive c...
6	733b43d534c67c1be948	youre reading properly ive asked evidence cont...
7	aad47a397f7ddc629d5d	please look discussion lygophile spoken
8	d19fcde8a3af2e472d74	utccall duty never made claims accuracy whilst...
9	7d4de482c60f1c8a79c6	man take care

```
import nltk
from nltk.stem import WordNetLemmatizer

# Init the Wordnet Lemmatizer
lemmatizer = WordNetLemmatizer()
```

```
#checking lemmatization
def lemmatize(text):
  output = ""
  text = text.split(" ")
  for word in text :
    word1 = wordnetlemmatizer.lemmatize(word,pos="n")
    word2 = wordnetlemmatizer.lemmatize(word1,pos="v")
    word3 = wordnetlemmatizer.lemmatize(word2,pos="a")
    word4 = wordnetlemmatizer.lemmatize(word3,pos="r")
    output = output + " "+word4

  return str(output.strip())
```

```
tdf['text'] = tdf['text'].apply(lambda x : lemmatize(x))
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	interview execution
1	b64a191301cad4f11287	know post truthful admins hate wont anything e...
2	5e1953d9ae04bdc66408	zzzzzzz youre real bore go bore someone else twt
3	23128f98196c8e8f7b90	yet remain confusion mention sac withdraw berg...
4	2d3f1254f71472bf2b78	refer lose interest vandalise talk page dark
5	21f4f0f4812a08ea6c28	utcthat wasnt attack ad hominem constructive c...
6	733b43d534c67c1be948	youre read properly ive ask evidence continue ...
7	aad47a397f7ddc629d5d	please look discussion lygophile speak
8	d19fcde8a3af2e472d74	utccall duty never make claim accuracy whilst ...
9	7d4de482c60f1c8a79c6	man take care

```
import re
def solve(s):
      return re.sub(r'(\S)\1+', r'\1', s)
tdf['text'] = tdf['text'].apply(lambda x : solve(x))
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	interview execution
2	5e1953d9ae04bdc66408	z youre real bore go bore someone else twt

```
filename = 'test.pkl'
```

```
pickle.dump(tdf,open(filename,'wb'))
```

```
tdf = pickle.load(open(filename,'rb'))
```

```
tdf.head(10)
```

	id	text
0	e0ae9d9474a5689a5791	interview execution
1	b64a191301cad4f11287	know post truthful admins hate wont anything e...
2	5e1953d9ae04bdc66408	z youre real bore go bore someone else twt
3	23128f98196c8e8f7b90	yet remain confusion mention sac withdraw berg...
4	2d3f1254f71472bf2b78	refer lose interest vandalise talk page dark
5	21f4f0f4812a08ea6c28	utcthat wasnt atack ad hominem constructive cr...
6	733b43d534c67c1be948	youre read properly ive ask evidence continue ...
7	aad47a397f7ddc629d5d	please lok discusion lygophile speak
8	d19fcde8a3af2e472d74	utcal duty never make claim acuracy whilst lar...
9	7d4de482c60f1c8a79c6	man take care

```
# merging data of both test and train to count the unique words in both
merge_data = main_df.append(tdf)
```

```
merge_data.shape
```

```
(127656, 8)
```

Feature Engineering : Extracting features

Bag of Words

```
from sklearn.feature_extraction.text import CountVectorizer
```

```
#reducing number of features to avoid system crash
vects = CountVectorizer(max_features = 1500)
vects.fit(merge_data['text'])
print("Dictionary has no.of unique words: ", len(vects.vocabulary_))
```

```
Dictionary has no.of unique words:  1500
```

```
list = [(k, v) for k, v in vects.vocabulary_.items()]
```

```
list.sort()
list
```

```
[('ability', 0),
 ('able', 1),
 ('absolutely', 2),
 ('abuse', 3),
 ('academic', 4),
 ('according', 5),
 ('account', 6),
 ('ace', 7),
 ('acept', 8),
 ('aceptable', 9),
 ('acepted', 10),
 ('acording', 11),
 ('acount', 12),
 ('acounts', 13),
 ('acros', 14),
 ('act', 15),
 ('acting', 16),
 ('action', 17),
 ('active', 18),
 ('activity', 19),
 ('actual', 20),
 ('actually', 21),
 ('actualy', 22),
 ('acurate', 23),
 ('acusations', 24),
 ('ad', 25),
 ('add', 26),
 ('added', 27),
 ('adding', 28),
 ('address', 29),
 ('aded', 30),
 ('ading', 31),
 ('adition', 32),
 ('admin', 33),
 ('administrator', 34),
 ('admins', 35),
 ('admit', 36),
 ('adres', 37),
 ('advertising', 38),
 ('advice', 39),
 ('afd', 40),
 ('afraid', 41),
 ('african', 42),
 ('age', 43),
 ('agenda', 44),
 ('ago', 45),
 ('agre', 46),
 ('agred', 47),
 ('agree', 48),
 ('ahead', 49),
 ('aid', 50),
 ('air', 51),
 ('al', 52),
 ('album', 53),
 ('almost', 54),
 ('alone', 55),
 ('along', 56),
 ('alow', 57),
```

```
vectors = vects.transform(merge_data['text'])
print(vectors)
```

```
  (0, 40)	1
  (0, 74)	2
  (0, 103)	1
  (0, 191)	1
  (0, 196)	1
  (0, 293)	1
  (0, 363)	1
  (0, 373)	1
  (0, 377)	1
  (0, 391)	1
  (0, 402)	1
  (0, 423)	1
  (0, 564)	1
  (0, 631)	1
  (0, 640)	1
  (0, 677)	1
  (0, 730)	1
  (0, 980)	1
  (0, 1007)	1
  (0, 1043)	1
  (0, 1112)	1
  (0, 1113)	1
  (0, 1115)	1
  (0, 1162)	3
  (0, 1256)	1
  :	:
  (127653, 928)	1
  (127653, 1339)	1
  (127653, 1478)	1
  (127654, 360)	1
  (127654, 403)	2
  (127654, 539)	2
  (127654, 935)	1
  (127655, 298)	1
  (127655, 361)	2
  (127655, 463)	1
  (127655, 492)	1
  (127655, 509)	1
  (127655, 511)	1
  (127655, 643)	3
  (127655, 704)	1
  (127655, 769)	1
  (127655, 779)	1
  (127655, 864)	1
  (127655, 883)	1
  (127655, 991)	1
  (127655, 1150)	1
  (127655, 1207)	1
  (127655, 1305)	1
  (127655, 1318)	1
  (127655, 1478)	1
```

TF-IDF

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
# create object
tfidf = TfidfVectorizer()
```

```
# get tf-df values
result = tfidf.fit_transform(merge_data['text'])
```

```
# get tf-df values
X_train = tfidf.fit_transform(df['text'])
```

```
type(X_train)
```

```
scipy.sparse.csr.csr_matrix
```

```
print('\ntf-idf value:')
print(result)
#in the tuple the first element is document index(number of rows), the second element is the word index in the dictonary
```

```
tf-idf value:
  (0, 164536)	0.08162775639600003
  (0, 36786)	0.1383773708083291
  (0, 136998)	0.19106284977877197
  (0, 7327)	0.14072578045886092
  (0, 52485)	0.11364055855427882
  (0, 171134)	0.1104795335496869
  (0, 91605)	0.07359448309361268
  (0, 222184)	0.17375229925882316
  (0, 159822)	0.09645707502466795
  (0, 115633)	0.0871770542829718
  (0, 207737)	0.05986187818574381
  (0, 58819)	0.09141723055603182
  (0, 3540)	0.1027873062317165
  (0, 195389)	0.12950429257158305
  (0, 29485)	0.08873955328070617
  (0, 12815)	0.044448108987265714
  (0, 147590)	0.1334978718782289
  (0, 172398)	0.1269141100226697
  (0, 50643)	0.07559533978629705
  (0, 80239)	0.08174036019704016
  (0, 168269)	0.15414304661840048
  (0, 4527)	0.1214460613343614
  (0, 27918)	0.11454439090631352
  (0, 9118)	0.18000913638360114
  (0, 53556)	0.07468020764173589
  :	:
  (127655, 98262)	0.30925957022580264
  (127655, 81506)	0.30925957022580264
  (127655, 166247)	0.2988654694301841
  (127655, 39729)	0.20495004179078455
  (127655, 30777)	0.20584178411250223
  (127655, 10916)	0.2003493658888452
  (127655, 157829)	0.16183305443925797
  (127655, 120440)	0.13627010425289823
  (127655, 166642)	0.1803881717193178
  (127655, 72238)	0.1419716035965556
  (127655, 67464)	0.16245178938537874
  (127655, 69513)	0.16472031004643536
  (127655, 137955)	0.11301357913812755
  (127655, 204574)	0.141617239911727
  (127655, 108373)	0.14783313011159807
  (127655, 185462)	0.12208594938800997
  (127655, 98040)	0.3731737934075189
  (127655, 176794)	0.10996725154009733
  (127655, 119893)	0.16588679912426466
  (127655, 135633)	0.10496503047279561
  (127655, 50522)	0.21752395276511993
  (127655, 42624)	0.13882941456492373
  (127655, 199365)	0.07514404953934736
  (127655, 72192)	0.10271676501252043
  (127655, 232553)	0.07745164970328033
```

```
corpus = merge_data['text']
```

```
merge_data = pd.DataFrame(merge_data)
```

```
X = vects.fit_transform(corpus).toarray()
y = merge_data.iloc[:, 1].values
```

```
X
```

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

```
y
```

```
array(['utci interested arguing policies resolve ongoing content dispute se wikipedia wikiproject united states presidential elections il working moneybomb closer selfreverted diferent requests echoed would requested wil rephrase didnt se answer building agrement moneybomb redlink given deletion reversion outline article caled moneybomb submited afd due time later se previous version however version wil require detailed answer ambiguity wil necesitate clarifying questions',
       'aded thre mising parameters templateinfobox organization converted ca articles information lost least articles company switched use infobox company template listed merge section wptfdh typicaly means redirect merging redirects help users find apropriate infobox se list redirect would like review conversion find edit history around midle page edits',
       'sandbox madre sandbox', ...,
       'listing girls alphabetical order keeps moving victoria bottom girls listed alphabetically thus making first even married would first maiden name adams keep getting moved bottom',
       'dumb fuck delete angry nintendo nerds page dumb fuck',
       'fine job would never find log look carefully image said deleted th gonzo fan talk contribs deleted imagecircumcision countrypng image exists commons rasterb image name appeared rd simply possible'],
      dtype=object)
```

```
# using binary relevance
from skmultilearn.problem_transform import BinaryRelevance
from sklearn.naive_bayes import GaussianNB
```

```
# initialize binary relevance multi-label classifier
# with a gaussian naive bayes base classifier
classifier = BinaryRelevance(GaussianNB())


import nltk
from nltk.stem import WordNetLemmatizer

# Init the Wordnet Lemmatizer
wordnetlemmatizer = WordNetLemmatizer()


#checking lemmatization
def lemmatize(text):
  output = ""
  text = text.split(" ")
  for word in text :
    word1 = wordnetlemmatizer.lemmatize(word,pos="n")
    word2 = wordnetlemmatizer.lemmatize(word1,pos="v")
    word3 = wordnetlemmatizer.lemmatize(word2,pos="a")
    word4 = wordnetlemmatizer.lemmatize(word3,pos="r")
    output = output + " "+word4

  return str(output.strip())


text = "going to happened easily policies "
lem = lemmatize(text)
lem
```

```
    'go to happen easily policy'
```