In [533]:

```
# Importing Libraries
import pandas as pd
import numpy as np
import re
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
```

In [618]:

```
# Loading Data
train_df = pd.read_csv("./train1.csv")
test_df = pd.read_csv("./test1.csv")
```

Training Data Set - has 3 columns ID, Label & Tweet. Tweet columns has tweets writen by users & Label columns contains binary values 1 & 0.

In [ ]:

```

```

In [619]:

```
test_df.head(10)
```

Out[619]:

	Unnamed: 0	text_id	text
0	0	hasoc_en_902	West Bengal Doctor Crisis: Protesting doctors ...
1	1	hasoc_en_416	68.5 million people have been forced to leave ...
2	2	hasoc_en_207	You came, you saw .... we will look after the ...
3	3	hasoc_en_595	We'll get Brexit delivered by October 31st. ...
4	4	hasoc_en_568	Fuck you. Go back to the dark ages you cow @IB...
5	5	hasoc_en_953	Boris Johnson faces Supreme Court bid to make ...
6	6	hasoc_en_685	What about a refund for not serving Halala to ...
7	7	hasoc_en_672	General election, DUP dumped out, Tory power w...
8	8	hasoc_en_746	#Repost free.wicked • • • • • • #freewicked ...
9	9	hasoc_en_527	Jesus Christ Christian News. Illuminati is now...

In [620]:

```
#Training Data Set
train_df.head(10)
```

Out[620]:

	text	labels
0	@realDonaldTrump This is one of the worst time...	0
1	How about the crowd in Oval in today's #AUSvIN...	1
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0
3	#etsy shop: Benedict Donald so called presiden...	1
4	@realDonaldTrump Good build a wall around Arka...	0
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1
6	@MeredthSalenger Anything to get a war to dist...	1
7	Why the FUCK did Doris mention demar lmfaooooo...	0
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0
9	@matthewamiller Because there are no consequen...	0

In [621]:

```
#Testing Data Set
test_df.head()
print('Testing data set has no Label column')
print(test_df.head(10))
```

```
Testing data set has no Label column
   Unnamed: 0       text_id
text
0           0  hasoc_en_902  West Bengal Doctor Crisis: Protesting d
octors ...
1           1  hasoc_en_416  68.5 million people have been forced to
leave ...
2           2  hasoc_en_207  You came, you saw .... we will look aft
er the ...
3           3  hasoc_en_595  We'll get Brexit delivered by October 3
1st.   ...
4           4  hasoc_en_568  Fuck you. Go back to the dark ages you
cow @IB...
5           5  hasoc_en_953  Boris Johnson faces Supreme Court bid t
o make ...
6           6  hasoc_en_685  What about a refund for not serving Hal
ala to ...
7           7  hasoc_en_672  General election, DUP dumped out, Tory
power w...
8           8  hasoc_en_746  #Repost free.wicked  • • • • • •  #free
wicked ...
9           9  hasoc_en_527  Jesus Christ Christian News. Illuminati
is now...
```

In [622]:

```
# Training Data Set Information
print("Training Data Set Info - Total Rows | Total Columns | Total Null Values")
print(train_df.info())
```

```
Training Data Set Info - Total Rows | Total Columns | Total Null Val
ues
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5266 entries, 0 to 5265
Data columns (total 2 columns):
text      5266 non-null object
labels    5266 non-null int64
dtypes: int64(1), object(1)
memory usage: 82.4+ KB
None
```

In [623]:

```
# Testing Data Set Information
print("Test Data Set Info - Total Rows | Total Columns | Total Null Values")
print(test_df.info())
```

```
Test Data Set Info - Total Rows | Total Columns | Total Null Values
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1153 entries, 0 to 1152
Data columns (total 3 columns):
Unnamed: 0    1153 non-null int64
text_id       1153 non-null object
text          1153 non-null object
dtypes: int64(1), object(2)
memory usage: 27.1+ KB
None
```

We can see in above tweet column in both data sets Training & Testing tweets are unstructured, for better analysis we first need to structure the tweets, remove the unwanted words, replace the misspelled words with the correct ones, replace the abriviation with full words

In [624]:

```
# Merging both the data sets as tweets in both the data set is unstructured
combine_df = train_df
combine_df.head()
```

Out[624]:

	text	labels
0	@realDonaldTrump This is one of the worst time...	0
1	How about the crowd in Oval in today's #AUSvIN...	1
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0
3	#etsy shop: Benedict Donald so called presiden...	1
4	@realDonaldTrump Good build a wall around Arka...	0

In [625]:

```
# Combine (Merged) Data Set Information
print("Combine Data Set Info - Total Rows | Total Columns | Total Null Values")
print(combine_df.info())
```

```
Combine Data Set Info - Total Rows | Total Columns | Total Null Valu
es
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5266 entries, 0 to 5265
Data columns (total 2 columns):
text      5266 non-null object
labels    5266 non-null int64
dtypes: int64(1), object(1)
memory usage: 82.4+ KB
None
```

We can see above, ID & Tweet column has 49159 has values where as Label column has 31962 values.

Data processing & cleaning

- Step C : Changing all the tweets into lowercase
- Step D : Apostrophe Lookup Not done due reduce in accuracy
- Step E : Short Word Lookup Not done
- Step F : Emoticon Lookup
- Step H : Replacing Special Characters with space
- Step I : Replacing Numbers (integers) with space
- Step J : Removing words whom length is 1 not done due to reduce in accuracy

Step C : Changing all the tweets into lowercase

In [626]:

```
combine_df['clean_tweet'] = combine_df['text'].apply(lambda x: x.lower())
combine_df.head(10)
```

Out[626]:

	text	labels	clean_tweet
0	@realDonaldTrump This is one of the worst time...	0	@realdonaldtrump this is one of the worst time...
1	How about the crowd in Oval in today's #AUSvIN...	1	how about the crowd in oval in today's #ausvin...
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0	@skroskz @shossy2 @joebiden biden &amp; his so...
3	#etsy shop: Benedict Donald so called presiden...	1	#etsy shop: benedict donald so called presiden...
4	@realDonaldTrump Good build a wall around Arka...	0	@realdonaldtrump good build a wall around arka...
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1	meanwhile ....dhoni's reply to icc ...... #...
6	@MeredthSalenger Anything to get a war to dist...	1	@meredthsalenger anything to get a war to dist...
7	Why the FUCK did Doris mention demar lmfaooooo...	0	why the fuck did doris mention demar lmfaooooo...
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0	@kimkardashian #trump2020 #fucktrump maybe yo...
9	@matthewamiller Because there are no consequen...	0	@matthewamiller because there are no consequen...

In [627]:

```
test_df['clean_tweet'] = test_df['text'].apply(lambda x: x.lower())
test_df.head(10)
```

Out[627]:

	Unnamed: 0	text_id	text	clean_tweet
0	0	hasoc_en_902	West Bengal Doctor Crisis: Protesting doctors ...	west bengal doctor crisis: protesting doctors ...
1	1	hasoc_en_416	68.5 million people have been forced to leave ...	68.5 million people have been forced to leave ...
2	2	hasoc_en_207	You came, you saw .... we will look after the ...	you came, you saw .... we will look after the ...
3	3	hasoc_en_595	We'll get Brexit delivered by October 31st. ...	we'll get brexit delivered by october 31st. ...
4	4	hasoc_en_568	Fuck you. Go back to the dark ages you cow @IB...	fuck you. go back to the dark ages you cow @ib...
5	5	hasoc_en_953	Boris Johnson faces Supreme Court bid to make ...	boris johnson faces supreme court bid to make ...
6	6	hasoc_en_685	What about a refund for not serving Halala to ...	what about a refund for not serving halala to ...
7	7	hasoc_en_672	General election, DUP dumped out, Tory power w...	general election, dup dumped out, tory power w...
8	8	hasoc_en_746	#Repost free.wicked • • • • • • #freewicked ...	#repost free.wicked • • • • • • #freewicked ...
9	9	hasoc_en_527	Jesus Christ Christian News. Illuminati is now...	jesus christ christian news. illuminati is now...

Step D : Apostrophe Lookup

Step F : Emoticon Lookup

In [628]:

```
for i in range(combine_df.shape[0]):
  combine_df['clean_tweet'][i] = combine_df['clean_tweet'][i].encode('ascii', 'i
gnore').decode('ascii')
```

```
/home/gsmodi/anaconda3/lib/python3.7/site-packages/ipykernel_launche
r.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/panda
s-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-cop
y
```

In [629]:

```
for i in range(test_df.shape[0]):
  test_df['clean_tweet'][i] = test_df['clean_tweet'][i].encode('ascii', 'ignore'
).decode('ascii')
```

```
/home/gsmodi/anaconda3/lib/python3.7/site-packages/ipykernel_launche
r.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/panda
s-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-cop
y
```

Step G : ReplacingPunctuations with space

In [630]:

```
combine_df['clean_tweet'] = combine_df['clean_tweet'].apply(lambda x: re.sub(r
'[^\w\s]',' ',x))
combine_df.head(10)
```

Out[630]:

	text	labels	clean_tweet
0	@realDonaldTrump This is one of the worst time...	0	realdonaldtrump this is one of the worst time...
1	How about the crowd in Oval in today's #AUSvIN...	1	how about the crowd in oval in today s ausvin...
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0	skroskz shossy2 joebiden biden amp his so...
3	#etsy shop: Benedict Donald so called presiden...	1	etsy shop benedict donald so called presiden...
4	@realDonaldTrump Good build a wall around Arka...	0	realdonaldtrump good build a wall around arka...
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1	meanwhile dhoni s reply to icc ...
6	@MeredthSalenger Anything to get a war to dist...	1	meredthsalenger anything to get a war to dist...
7	Why the FUCK did Doris mention demar lmfaooooo...	0	why the fuck did doris mention demar lmfaooooo...
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0	kimkardashian trump2020 fucktrump maybe yo...
9	@matthewamiller Because there are no consequen...	0	matthewamiller because there are no consequen...

In [631]:

```
#test_df['clean_tweet'] = test_df['clean_tweet'].apply(lambda x: re.sub(r'[^\w
\s]',' ',x))
#test_df.head(10)
```

Step I : Replacing Numbers (integers) with space

In [632]:

```
combine_df['clean_tweet'] = combine_df['clean_tweet'].apply(lambda x: re.sub(r
'[^a-zA-Z]',' ',x))
combine_df.head(10)
```

Out[632]:

	text	labels	clean_tweet
0	@realDonaldTrump This is one of the worst time...	0	realdonaldtrump this is one of the worst time...
1	How about the crowd in Oval in today's #AUSvIN...	1	how about the crowd in oval in today s ausvin...
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0	skroskz shossy joebiden biden amp his so...
3	#etsy shop: Benedict Donald so called presiden...	1	etsy shop benedict donald so called presiden...
4	@realDonaldTrump Good build a wall around Arka...	0	realdonaldtrump good build a wall around arka...
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1	meanwhile dhoni s reply to icc ...
6	@MeredthSalenger Anything to get a war to dist...	1	meredthsalenger anything to get a war to dist...
7	Why the FUCK did Doris mention demar lmfaooooo...	0	why the fuck did doris mention demar lmfaoooo...
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0	kimkardashian trump fucktrump maybe yo...
9	@matthewamiller Because there are no consequen...	0	matthewamiller because there are no consequen...

In [633]:

```
test_df['clean_tweet'] = test_df['clean_tweet'].apply(lambda x: re.sub(r'[^\w\s]',' ',x))
test_df.head(10)
```

Out[633]:

	Unnamed: 0	text_id	text	clean_tweet
0	0	hasoc_en_902	West Bengal Doctor Crisis: Protesting doctors ...	west bengal doctor crisis protesting doctors ...
1	1	hasoc_en_416	68.5 million people have been forced to leave ...	68 5 million people have been forced to leave ...
2	2	hasoc_en_207	You came, you saw .... we will look after the ...	you came you saw we will look after the ...
3	3	hasoc_en_595	We'll get Brexit delivered by October 31st. ...	we ll get brexit delivered by october 31st ...
4	4	hasoc_en_568	Fuck you. Go back to the dark ages you cow @IB...	fuck you go back to the dark ages you cow ib...
5	5	hasoc_en_953	Boris Johnson faces Supreme Court bid to make ...	boris johnson faces supreme court bid to make ...
6	6	hasoc_en_685	What about a refund for not serving Halala to ...	what about a refund for not serving halala to ...
7	7	hasoc_en_672	General election, DUP dumped out, Tory power w...	general election dup dumped out tory power w...
8	8	hasoc_en_746	#Repost free.wicked • • • • • • #freewicked ...	repost free wicked freewicked freet...
9	9	hasoc_en_527	Jesus Christ Christian News. Illuminati is now...	jesus christ christian news illuminati is now...

In [634]:

```
# Spelling correction is a cool feature which TextBlob offers, we can be accessed using the correct function as shown below.
blob = TextBlob("Why are you stting on this bech??") # Scentence with two errors
print(blob.correct()) # Correct function giave us the best possible word simmilar to "gret"
```

```
Why are you sitting on this bench??
```

In [635]:

```
# we can see all the similar matches our first error along with the probability
 score.
blob.words[3].spellcheck()
```

Out[635]:

```
[('sitting', 0.8078078078078078),
 ('setting', 0.11411411411411411),
 ('string', 0.036036036036036036),
 ('sting', 0.02702702702702703),
 ('stating', 0.015015015015015015)]
```

Applying TextBlob on our data set - Spelling correction

In [636]:

```
# Not cleaning the just showing the spelling check as its take lot of time to pr
ocess all these tweets
## Shown sample how its must done
#combine_df['clean_tweet'] = combine_df['clean_tweet'][0:10].apply(lambda x: str
(TextBlob(x).correct()))
#combine_df.head()
```

In [637]:

```
# Not cleaning the just showing the spelling check as its take lot of time to pr
ocess all these tweets
## Shown sample how its must done
text = combine_df['clean_tweet'][0:10].apply(lambda x: str(TextBlob(x).correct
()))
text
```

Out[637]:

```
0     realdonaldtrump this is one of the worst time...
1    how about the crowd in oval in today s  austin...
2     skroskz  shows   joebiden widen  amp  his son...
3     easy shop  benedict donald so called presiden...
4     realdonaldtrump good build a wall around arka...
5    meanwhile     don s reply to ice            dh...
6     meredthsalenger anything to get a war to dist...
7    why the fuck did boris mention dear lmfaooooo ...
8     kimkardashian  tramp      fucktrump  maybe yo...
9     matthewamiller because there are no consequen...
Name: clean_tweet, dtype: object
```

In [638]:

```
# Importing stop words from NLTK coupus and word tokenizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

In [639]:

```
# Creating token for the clean tweets
combine_df['tweet_token'] = combine_df['clean_tweet'].apply(lambda x: word_token
ize(x))

## Fully formated tweets & there tokens
combine_df.head(10)
```

Out[639]:

	text	labels	clean_tweet	tweet_token
0	@realDonaldTrump This is one of the worst time...	0	realdonaldtrump this is one of the worst time...	[realdonaldtrump, this, is, one, of, the, wors...
1	How about the crowd in Oval in today's #AUSvIN...	1	how about the crowd in oval in today s ausvin...	[how, about, the, crowd, in, oval, in, today, ...
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0	skroskz shossy joebiden biden amp his so...	[skroskz, shossy, joebiden, biden, amp, his, s...
3	#etsy shop: Benedict Donald so called presiden...	1	etsy shop benedict donald so called presiden...	[etsy, shop, benedict, donald, so, called, pre...
4	@realDonaldTrump Good build a wall around Arka...	0	realdonaldtrump good build a wall around arka...	[realdonaldtrump, good, build, a, wall, around...
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1	meanwhile dhoni s reply to icc ...	[meanwhile, dhoni, s, reply, to, icc, dhonikee...
6	@MeredthSalenger Anything to get a war to dist...	1	meredthsalenger anything to get a war to dist...	[meredthsalenger, anything, to, get, a, war, t...
7	Why the FUCK did Doris mention demar lmfaooooo...	0	why the fuck did doris mention demar lmfaooooo...	[why, the, fuck, did, doris, mention, demar, l...
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0	kimkardashian trump fucktrump maybe yo...	[kimkardashian, trump, fucktrump, maybe, you, ...
9	@matthewamiller Because there are no consequen...	0	matthewamiller because there are no consequen...	[matthewamiller, because, there, are, no, cons...

In [640]:

```
test_df['tweet_token'] = test_df['clean_tweet'].apply(lambda x: word_tokenize(x))

## Fully formated tweets & there tokens
test_df.head(10)
```

Out[640]:

	Unnamed: 0	text_id	text	clean_tweet	tweet_token
0	0	hasoc_en_902	West Bengal Doctor Crisis: Protesting doctors ...	west bengal doctor crisis protesting doctors ...	[west, bengal, doctor, crisis, protesting, doc...
1	1	hasoc_en_416	68.5 million people have been forced to leave ...	68 5 million people have been forced to leave ...	[68, 5, million, people, have, been, forced, t...
2	2	hasoc_en_207	You came, you saw .... we will look after the ...	you came you saw we will look after the ...	[you, came, you, saw, we, will, look, after, t...
3	3	hasoc_en_595	We'll get Brexit delivered by October 31st. ...	we ll get brexit delivered by october 31st ...	[we, ll, get, brexit, delivered, by, october, ...
4	4	hasoc_en_568	Fuck you. Go back to the dark ages you cow @IB...	fuck you go back to the dark ages you cow ib...	[fuck, you, go, back, to, the, dark, ages, you...
5	5	hasoc_en_953	Boris Johnson faces Supreme Court bid to make ...	boris johnson faces supreme court bid to make ...	[boris, johnson, faces, supreme, court, bid, t...
6	6	hasoc_en_685	What about a refund for not serving Halala to ...	what about a refund for not serving halala to ...	[what, about, a, refund, for, not, serving, ha...
7	7	hasoc_en_672	General election, DUP dumped out, Tory power w...	general election dup dumped out tory power w...	[general, election, dup, dumped, out, tory, po...
8	8	hasoc_en_746	#Repost free.wicked • • • • • • #freewicked ...	repost free wicked freewicked freet...	[repost, free, wicked, freewicked, freethekids...
9	9	hasoc_en_527	Jesus Christ Christian News. Illuminati is now...	jesus christ christian news illuminati is now...	[jesus, christ, christian, news, illuminati, i...

In [641]:

```
# Importing stop words from NLTK corpus for english language

import nltk
nltk.download('stopwords')
stop_words = set(stopwords.words('english'))
stop_words
```

```
[nltk_data] Downloading package stopwords to /home/gsmodi/nltk_dat
a...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[641]:

```
{'a',
 'about',
 'above',
 'after',
 'again',
 'against',
 'ain',
 'all',
 'am',
 'an',
 'and',
 'any',
 'are',
 'aren',
 "aren't",
 'as',
 'at',
 'be',
 'because',
 'been',
 'before',
 'being',
 'below',
 'between',
 'both',
 'but',
 'by',
 'can',
 'couldn',
 "couldn't",
 'd',
 'did',
 'didn',
 "didn't",
 'do',
 'does',
 'doesn',
 "doesn't",
 'doing',
 'don',
 "don't",
 'down',
 'during',
 'each',
 'few',
 'for',
 'from',
 'further',
 'had',
 'hadn',
 "hadn't",
 'has',
 'hasn',
 "hasn't",
 'have',
 'haven',
 "haven't",
 'having',
 'he',
 'her',
 'here',
 'hers',
 'herself',
 'him',
 'himself',
 'his',
 'how',
 'i',
 'if',
 'in',
 'into',
 'is',
 'isn',
 "isn't",
 'it',
 "it's",
 'its',
 'itself',
 'just',
 'll',
 'm',
 'ma',
 'me',
 'mightn',
 "mightn't",
 'more',
 'most',
 'mustn',
 "mustn't",
 'my',
 'myself',
 'needn',
 "needn't",
 'no',
 'nor',
 'not',
 'now',
 'o',
 'of',
 'off',
 'on',
 'once',
 'only',
 'or',
 'other',
 'our',
 'ours',
 'ourselves',
 'out',
 'over',
 'own',
 're',
 's',
 'same',
 'shan',
 "shan't",
 'she',
 "she's",
 'should',
 "should've",
 'shouldn',
 "shouldn't",
 'so',
 'some',
 'such',
 't',
 'than',
 'that',
 "that'll",
 'the',
 'their',
 'theirs',
 'them',
 'themselves',
 'then',
 'there',
 'these',
 'they',
 'this',
 'those',
 'through',
 'to',
 'too',
 'under',
 'until',
 'up',
 've',
 'very',
 'was',
 'wasn',
 "wasn't",
 'we',
 'were',
 'weren',
 "weren't",
 'what',
 'when',
 'where',
 'which',
 'while',
 'who',
 'whom',
 'why',
 'will',
 'with',
 'won',
 "won't",
 'wouldn',
 "wouldn't",
 'y',
 'you',
 "you'd",
 "you'll",
 "you're",
 "you've",
 'your',
 'yours',
 'yourself',
 'yourselves'}
```

In [642]:

```
# Created new columns of tokens - where stop words are being removed
combine_df['tweet_token_filtered'] = combine_df['tweet_token'].apply(lambda x: [word for word in x if not word in stop_words])

## Tokens columns with stop words and without stop words
combine_df[['tweet_token', 'tweet_token_filtered']].head(10)
```

Out[642]:

	tweet_token	tweet_token_filtered
0	[realdonaldtrump, this, is, one, of, the, wors...	[realdonaldtrump, one, worst, times, american,...
1	[how, about, the, crowd, in, oval, in, today, ...	[crowd, oval, today, ausvind, holding, balidan...
2	[skroskz, shossy, joebiden, biden, amp, his, s...	[skroskz, shossy, joebiden, biden, amp, son, h...
3	[etsy, shop, benedict, donald, so, called, pre...	[etsy, shop, benedict, donald, called, preside...
4	[realdonaldtrump, good, build, a, wall, around...	[realdonaldtrump, good, build, wall, around, a...
5	[meanwhile, dhoni, s, reply, to, icc, dhonikee...	[meanwhile, dhoni, reply, icc, dhonikeeptheglo...
6	[meredthsalenger, anything, to, get, a, war, t...	[meredthsalenger, anything, get, war, distract...
7	[why, the, fuck, did, doris, mention, demar, l...	[fuck, doris, mention, demar, lmfaooooo, dickh...
8	[kimkardashian, trump, fucktrump, maybe, you, ...	[kimkardashian, trump, fucktrump, maybe, hire,...
9	[matthewamiller, because, there, are, no, cons...	[matthewamiller, consequences, individual, bas...

In [643]:

```
test_df['tweet_token_filtered'] = test_df['tweet_token'].apply(lambda x: [word for word in x if not word in stop_words])

## Tokens columns with stop words and without stop words
test_df[['tweet_token', 'tweet_token_filtered']].head(10)
```

Out[643]:

	tweet_token	tweet_token_filtered
0	[west, bengal, doctor, crisis, protesting, doc...	[west, bengal, doctor, crisis, protesting, doc...
1	[68, 5, million, people, have, been, forced, t...	[68, 5, million, people, forced, leave, homes,...
2	[you, came, you, saw, we, will, look, after, t...	[came, saw, look, fort, good, luck]
3	[we, ll, get, brexit, delivered, by, october, ...	[get, brexit, delivered, october, 31st, help, ...
4	[fuck, you, go, back, to, the, dark, ages, you...	[fuck, go, back, dark, ages, cow, ibnliverealt...
5	[boris, johnson, faces, supreme, court, bid, t...	[boris, johnson, faces, supreme, court, bid, m...
6	[what, about, a, refund, for, not, serving, ha...	[refund, serving, halala, muslims, regularly, ...
7	[general, election, dup, dumped, out, tory, po...	[general, election, dup, dumped, tory, power, ...
8	[repost, free, wicked, freewicked, freethekids...	[repost, free, wicked, freewicked, freethekids...
9	[jesus, christ, christian, news, illuminati, i...	[jesus, christ, christian, news, illuminati, c...

We will create 2 new columns

- One For Stemming
- Second For Lemmatization

The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form, whereas stemming just removes the last few characters, often leading to incorrect meanings and spelling errors.

Stemming - Stemming refers to the removal of suffices, like “ing”, “ly”, “s”, etc. by a simple rule-based approach.

In [644]:

```
# Importing library for stemming
from nltk.stem import PorterStemmer
stemming = PorterStemmer()
```

In [645]:

```
# Created one more columns tweet_stemmed it shows tweets' stemmed version
combine_df['tweet_stemmed'] = combine_df['tweet_token_filtered'].apply(lambda x:
' '.join([stemming.stem(i) for i in x]))
combine_df['tweet_stemmed'].head(10)
```

Out[645]:

```
0    realdonaldtrump one worst time american caus s...
1    crowd oval today ausvind hold balidan badg ban...
2    skroskz shossi joebiden biden amp son hunter t...
3    etsi shop benedict donald call presid traitor ...
4    realdonaldtrump good build wall around arkansa...
5    meanwhil dhoni repli icc dhonikeeptheglov dhon...
6    meredthsaleng anyth get war distract fucktrump...
7           fuck dori mention demar lmfaooooo dickhead
8    kimkardashian trump fucktrump mayb hire ex con...
9    matthewamil consequ individu basic advertis fr...
Name: tweet_stemmed, dtype: object
```

In [646]:

```
test_df['tweet_stemmed'] = test_df['tweet_token_filtered'].apply(lambda x: ' '.join([stemming.stem(i) for i in x]))
test_df['tweet_stemmed'].head(10)
```

Out[646]:

```
0    west bengal doctor crisi protest doctor agre m...
1    68 5 million peopl forc leav home read http we...
2                         came saw look fort good luck
3    get brexit deliv octob 31st help build movemen...
4    fuck go back dark age cow ibnliverealtim rape ...
5    bori johnson face suprem court bid make stand ...
6    refund serv halala muslim regularli ad onion j...
7           gener elect dup dump tori power weaken way
8    repost free wick freewick freethekid terrorist...
9    jesu christ christian news illuminati chang bi...
Name: tweet_stemmed, dtype: object
```

Lemmatization - Lemmatization is the process of converting a word to its base form.

In [647]:

```
# Importing library for lemmatizing
from nltk.stem.wordnet import WordNetLemmatizer
lemmatizing = WordNetLemmatizer()
```

In [648]:

```
# Created one more columns tweet_lemmatized it shows tweets' lemmatized version
combine_df['tweet_lemmatized'] = combine_df['tweet_token_filtered'].apply(lambda x: ' '.join([lemmatizing.lemmatize(i) for i in x]))
combine_df['tweet_lemmatized'].head(10)
```

Out[648]:

```
0    realdonaldtrump one worst time american causin...
1    crowd oval today ausvind holding balidan badge...
2    skroskz shossy joebiden biden amp son hunter t...
3    etsy shop benedict donald called president tra...
4    realdonaldtrump good build wall around arkansa...
5    meanwhile dhoni reply icc dhonikeeptheglove dh...
6    meredthsalenger anything get war distract fuck...
7          fuck doris mention demar lmfaooooo dickhead
8    kimkardashian trump fucktrump maybe hire ex co...
9    matthewamiller consequence individual basicall...
Name: tweet_lemmatized, dtype: object
```

In [649]:

```
test_df['tweet_lemmatized'] = test_df['tweet_token_filtered'].apply(lambda x: ' '.join([lemmatizing.lemmatize(i) for i in x]))
test_df['tweet_lemmatized'].head(10)
```

Out[649]:

```
0    west bengal doctor crisis protesting doctor ag...
1    68 5 million people forced leave home read htt...
2                        came saw look fort good luck
3    get brexit delivered october 31st help build m...
4    fuck go back dark age cow ibnliverealtime rape...
5    boris johnson face supreme court bid make stan...
6    refund serving halala muslim regularly adding ...
7    general election dup dumped tory power weakene...
8    repost free wicked freewicked freethekids terr...
9    jesus christ christian news illuminati changin...
Name: tweet_lemmatized, dtype: object
```

In [650]:

```
# Our final dataframe - Fully formatted, Processed, Noise less, Cleaned, ready t
o analyse
## for further analysis we consider 2 columns i.e. "tweet_stemmed" & "tweet_lema
tized"
### We are using 2 columns to see which of them give us better score.
combine_df.head(10)
```

Out[650]:

	text	labels	clean_tweet	tweet_token	tweet_token_filtered	tweet_s
0	@realDonaldTrump This is one of the worst time...	0	realdonaldtrump this is one of the worst time...	[realdonaldtrump, this, is, one, of, the, wors...	[realdonaldtrump, one, worst, times, american,...	realdon: one w americ
1	How about the crowd in Oval in today's #AUSvIN...	1	how about the crowd in oval in today s ausvin...	[how, about, the, crowd, in, oval, in, today, ...	[crowd, oval, today, ausvind, holding, balidan...	crowd ov ausv balic
2	@skroskz @shossy2 @JoeBiden Biden &amp; his so...	0	skroskz shossy joebiden biden amp his so...	[skroskz, shossy, joebiden, biden, amp, his, s...	[skroskz, shossy, joebiden, biden, amp, son, h...	skrosk joebid amp so
3	#etsy shop: Benedict Donald so called presiden...	1	etsy shop benedict donald so called presiden...	[etsy, shop, benedict, donald, so, called, pre...	[etsy, shop, benedict, donald, called, preside...	e benedic call pres
4	@realDonaldTrump Good build a wall around Arka...	0	realdonaldtrump good build a wall around arka...	[realdonaldtrump, good, build, a, wall, around...	[realdonaldtrump, good, build, wall, around, a...	realdon: good b around a
5	Meanwhile ....Dhoni's Reply To ICC ...... #...	1	meanwhile dhoni s reply to icc ...	[meanwhile, dhoni, s, reply, to, icc, dhonikee...	[meanwhile, dhoni, reply, icc, dhonikeeptheglo...	meanw dhonikee
6	@MeredthSalenger Anything to get a war to dist...	1	meredthsalenger anything to get a war to dist...	[meredthsalenger, anything, to, get, a, war, t...	[meredthsalenger, anything, get, war, distract...	mered anyth fucl
7	Why the FUCK did Doris mention demar lmfaooooo...	0	why the fuck did doris mention demar lmfaooooo...	[why, the, fuck, did, doris, mention, demar, l...	[fuck, doris, mention, demar, lmfaooooo, dickh...	fuck dori demar lm (
8	@KimKardashian #trump2020 #fucktrump Maybe yo...	0	kimkardashian trump fucktrump maybe yo...	[kimkardashian, trump, fucktrump, maybe, you, ...	[kimkardashian, trump, fucktrump, maybe, hire,...	kimka trump fu may
9	@matthewamiller Because there are no consequen...	0	matthewamiller because there are no consequen...	[matthewamiller, because, there, are, no, cons...	[matthewamiller, consequences, individual, bas...	matt consequ basic adv

A - Will see the most commonly used words for both the columns i.e. "tweet_stemmed" & "tweet_lematized"

A - Bag-of-Words Features

In [569]:

```
# Importing library
from sklearn.feature_extraction.text import CountVectorizer
bow_vectorizer = CountVectorizer(max_df=0.90, min_df=2, max_features=1000, stop_
words='english')
bow_vectorizer
```

Out[569]:

```
CountVectorizer(analyzer='word', binary=False, decode_error='stric
t',
                dtype=<class 'numpy.int64'>, encoding='utf-8', input
='content',
                lowercase=True, max_df=0.9, max_features=1000, min_d
f=2,
                ngram_range=(1, 1), preprocessor=None, stop_words='e
nglish',
                strip_accents=None, token_pattern='(?u)\\b\\w\\w+
\\b',
                tokenizer=None, vocabulary=None)
```

A.1 Bag-Of-Words feature matrix - For columns "combine_df['tweet_stemmed']"

In [661]:

```
# bag-of-words feature matrix - For columns "combine_df['tweet_stemmed']"
bow_stem = bow_vectorizer.fit_transform(combine_df['tweet_stemmed'])
bow_stem
```

Out[661]:

```
<5266x1000 sparse matrix of type '<class 'numpy.int64'>'
        with 46358 stored elements in Compressed Sparse Row format>
```

A.2 Bag-Of-Words feature matrix - For column - combine_df['tweet_lemmatized']

In [662]:

```
# bag-of-words feature matrix - For column - combine_df['tweet_lemmatized']
bow_lemm = bow_vectorizer.fit_transform(combine_df['tweet_lemmatized'])
bow_lemm
```

Out[662]:

```
<5266x1000 sparse matrix of type '<class 'numpy.int64'>'
        with 43001 stored elements in Compressed Sparse Row format>
```

B - TF-IDF Features

In [651]:

```
# Importing library
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer(max_df=0.90, min_df=1, max_features=1000, sto
p_words='english')
tfidf_vectorizer
```

Out[651]:

```
TfidfVectorizer(analyzer='word', binary=False, decode_error='stric
t',
                dtype=<class 'numpy.float64'>, encoding='utf-8',
                input='content', lowercase=True, max_df=0.9, max_fea
tures=1000,
                min_df=1, ngram_range=(1, 1), norm='l2', preprocesso
r=None,
                smooth_idf=True, stop_words='english', strip_accents
=None,
                sublinear_tf=False, token_pattern='(?u)\\b\\w\\w+
\\b',
                tokenizer=None, use_idf=True, vocabulary=None)
```

B.1 TF-IDF feature matrix - For columns "combine_df['tweet_stemmed']"

In [652]:

```
 TF-IDF feature matrix - For columns "combine_df['tweet_stemmed']"
combine_df.head(3)
tfidf_stem3 = tfidf_vectorizer.fit_transform(combine_df['tweet_stemmed'])
tfidf_stem3
```

Out[652]:

```
<5266x1000 sparse matrix of type '<class 'numpy.float64'>'
	with 46353 stored elements in Compressed Sparse Row format>
```

In [653]:

```
tfidf_stem2 = tfidf_vectorizer.fit_transform(test_df['tweet_stemmed'])
tfidf_stem2
```

Out[653]:

```
<1153x1000 sparse matrix of type '<class 'numpy.float64'>'
	with 10521 stored elements in Compressed Sparse Row format>
```

B.2 TF-IDF feature matrix - For columns "combine_df['tweet_lemmatized']"

In [654]:

```
# TF-IDF feature matrix - For columns "combine_df['tweet_lemmatized']"
tfidf_lemm1 = tfidf_vectorizer.fit_transform(combine_df['tweet_lemmatized'])
tfidf_lemm1
```

Out[654]:

```
<5266x1000 sparse matrix of type '<class 'numpy.float64'>'
	with 42995 stored elements in Compressed Sparse Row format>
```

In [655]:

```
# TF-IDF feature matrix - For columns "combine_df['tweet_lemmatized']"
tfidf_lemm2 = tfidf_vectorizer.fit_transform(test_df['tweet_lemmatized'])
tfidf_lemm2
```

Out[655]:

```
<1153x1000 sparse matrix of type '<class 'numpy.float64'>'
	with 9601 stored elements in Compressed Sparse Row format>
```

Logistic Regression Model Building: Twitter Sentiment Analysis

In [656]:

```
# Importing Libraries
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import f1_score
```

A Building model using Bag-of-Words features

A.1 For columns "combine_df['tweet_stemmed']"

In [663]:

```
# A.1 For columns "combine_df['tweet_stemmed']"
train_bow = bow_stem[:5866,:]
test_bow = bow_stem[5266:,:]

# splitting data into training and validation set
xtrain_bow, xvalid_bow, ytrain, yvalid = train_test_split(train_bow, train_df['labels'], random_state=42, test_size=0.25)

lreg = LogisticRegression()
lreg.fit(xtrain_bow, ytrain) # training the model

prediction = lreg.predict_proba(xvalid_bow) # predicting on the validation set
prediction_int = prediction[:,1] >= 0.3 # if prediction is greater than or equal to 0.3 than 1 else 0
prediction_int = prediction_int.astype(np.int)

A1 = f1_score(yvalid, prediction_int) # calculating f1 score
print(A1)
```

```
0.7667862634546386
```

```
/home/gsmodi/anaconda3/lib/python3.7/site-packages/sklearn/linear_mo
del/logistic.py:432: FutureWarning: Default solver will be changed t
o 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)
```

A.2 For columns "combine_df['tweet_lemmatized']"

In [664]:

```
# A.2 For columns "combine_df['tweet_lemmatized']"
train_bow = bow_lemm[:5266,:]
test_bow = bow_lemm[5266:,:]

# splitting data into training and validation set
xtrain_bow, xvalid_bow, ytrain, yvalid = train_test_split(train_bow, train_df['labels'], random_state=42, test_size=0.25)

lreg = LogisticRegression()
lreg.fit(xtrain_bow, ytrain) # training the model

prediction = lreg.predict_proba(xvalid_bow) # predicting on the validation set
prediction_int = prediction[:,1] >= 0.3 # if prediction is greater than or equal to 0.3 than 1 else 0
prediction_int = prediction_int.astype(np.int)

A2 = f1_score(yvalid, prediction_int) # calculating f1 score
print(A2)




from sklearn.naive_bayes import MultinomialNB # import Multinomial Naive Bayes model from sklearn.naive_bayes
nb = MultinomialNB(alpha = 10) # get object of Multinomial naive bayes model with alpha parameter = 10
nb.fit(xtrain_bow, ytrain)# fit our both traing data tweets as well as its sentiments to the multinomial naive bayes model
```

```
0.7680491551459294
```

Out[664]:

```
MultinomialNB(alpha=10, class_prior=None, fit_prior=True)
```

In [500]:

```
y_pred_nb = nb.predict(xvalid_bow)
A9 = f1_score(yvalid, y_pred_nb) # calculating f1 score
print(A9)
```

```
0.7247278382581648
```

B Building model using TF-IDF features

B.1 For columns "combine_df['tweet_stemmed']"

In [657]:

```
# B.1 For columns "combine_df['tweet_stemmed']"
train_tfidf = tfidf_stem3[:5852,:]
test_tfidf = tfidf_stem2[1153:,:]


xtrain_tfidf, xvalid_tfidf, ytrain, yvalid = train_test_split(train_tfidf, train
_df['labels'], random_state=42, test_size=0.25)

lreg.fit(xtrain_tfidf, ytrain)

prediction = lreg.predict_proba(xvalid_tfidf)
prediction_int = prediction[:,1] >= 0.3
prediction_int = prediction_int.astype(np.int)

B1 = f1_score(yvalid, prediction_int) # calculating f1 score
from sklearn.metrics import accuracy_score
b2 = accuracy_score(yvalid, prediction_int)
print(b2)
print(B1)


from sklearn.metrics import f1_score

from sklearn.svm import SVC
clf = SVC(kernel='linear')
clf.fit(xtrain_tfidf, ytrain)
prediction = clf.predict(xvalid_tfidf)
B2 = f1_score(yvalid, prediction_int) # calculating f1 score
from sklearn.metrics import accuracy_score
b2 = accuracy_score(yvalid, prediction)
print(b2)
print(B2)
```

```
/home/gsmodi/anaconda3/lib/python3.7/site-packages/sklearn/linear_mo
del/logistic.py:432: FutureWarning: Default solver will be changed t
o 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)
```

```
0.6590736522399393
0.7834056922334781
0.6719817767653758
0.7834056922334781
```

B.2 For columns "combine_df['tweet_lemmatized']"

In [658]:

```
# B.2 For columns "combine_df['tweet_lemmatized']"
train_tfidf = tfidf_lemm1[:5266,:]
test_tfidf = tfidf_lemm[5266:,:]

xtrain_tfidf = train_tfidf[ytrain.index]
xvalid_tfidf = train_tfidf[yvalid.index]

lreg.fit(xtrain_tfidf, ytrain)

prediction = lreg.predict_proba(xvalid_tfidf)
prediction_int = prediction[:,1] >= 0.3
prediction_int = prediction_int.astype(np.int)




B2 = f1_score(yvalid, prediction_int) # calculating f1 score
from sklearn.metrics import accuracy_score
b2 = accuracy_score(yvalid, prediction_int)
print(b2)
print(B2)





from sklearn.metrics import f1_score

from sklearn.svm import SVC
clf = SVC(kernel='linear')
clf.fit(tfidf_lemm1,combine_df['labels'])
prediction3 = clf.predict(tfidf_lemm2)
print('done')
```

```
/home/gsmodi/anaconda3/lib/python3.7/site-packages/sklearn/linear_mo
del/logistic.py:432: FutureWarning: Default solver will be changed t
o 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)

0.6522399392558846
0.7802303262955855
done
```

In [435]:

```
print("F1 - Score Chart")
print("* F1-Score - Model using Bag-of-Words features")
print("   F1-Score = ",A1," - For column tweets are stemmed")
print("   F1-Score = ",A2," - For column tweets are Lemmatized")
print("* F1-Score - Model using TF-IDF features")
print("   F1-Score = ",B1," - For column tweets are stemmed")
print("   F1-Score = ",B2," - For column tweets are Lemmatized")
```

```
F1 - Score Chart
* F1-Score - Model using Bag-of-Words features
   F1-Score =  0.7695262483994878  - For column tweets are stemmed
   F1-Score =  0.7639188605955978  - For column tweets are Lemmatize
d
* F1-Score - Model using TF-IDF features
   F1-Score =  0.773462783171521  - For column tweets are stemmed
   F1-Score =  0.7719580983078163  - For column tweets are Lemmatize
d
```

In [660]:

```
y_pred=pd.DataFrame(data=prediction3,columns=['labels']);
print(y_pred)
y_pred.to_csv("./submissionmodig6.csv",index=True)
```

```
      labels
0          1
1          1
2          0
3          1
4          1
...      ...
1148       1
1149       0
1150       1
1151       1
1152       1

[1153 rows x 1 columns]
```

Conclusion

In above code we try different preprocessing method and the also implemented all possible algorithm and selects linear SVC which gives maximum accuracy

Also we tried TFIDF as well as Bag of words both technique to check which one is more fruitful

In all combinations Tf -idf with SVC having linear Kernel gives maximum accuracy

In [ ]:

```

```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: