Import Packages

In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
import matplotlib.pyplot as plt
from scipy.stats import pearsonr
import sklearn.ensemble
from nltk.corpus import stopwords
from textblob import TextBlob
from nltk.stem import PorterStemmer
from textblob import Word
```

In [47]:

```
df = pd.read_csv('C:/Data Science/Greyatom/TwitterSentimentAnalysis/train.csv')
df1 = pd.read_csv('C:/Data Science/Greyatom/TwitterSentimentAnalysis/test.csv')# use / not
```

In [3]:

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

Out[3]:

	tweet_id	tweet	sentiment
0	1701	#sxswnui #sxsw #apple defining language of tou...	1
1	1851	Learning ab Google doodles! All doodles should...	1
2	2689	one of the most in-your-face ex. of stealing t...	2
3	4525	This iPhone #SXSW app would b pretty awesome i...	0
4	3604	Line outside the Apple store in Austin waiting...	1
5	966	#technews One lone dude awaits iPad 2 at Apple...	1
6	1395	SXSW Tips, Prince, NPR Videos, Toy Shopping Wi...	1
7	8182	NU user RT @mention New #UberSocial for #iPhon...	1
8	8835	Free #SXSW sampler on iTunes {link} #FreeMusic	2
9	883	I think I might go all weekend without seeing ...	2

Analyzing The Data

In [4]:

```
df.info()
```

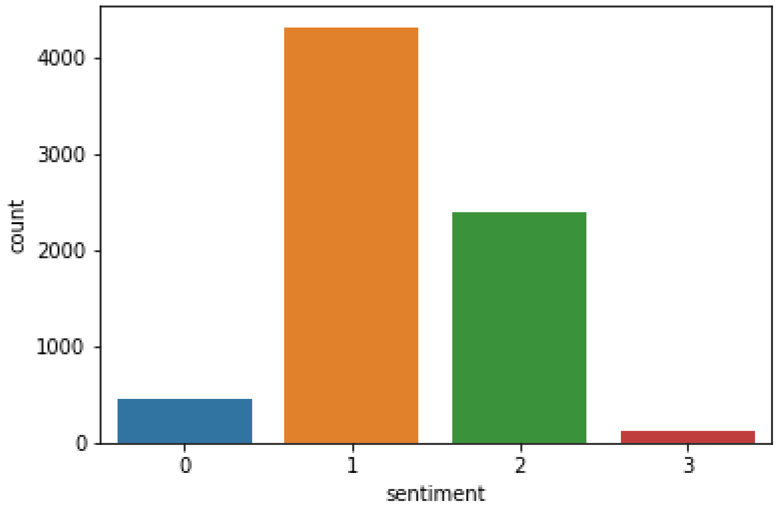
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7274 entries, 0 to 7273
Data columns (total 3 columns):
tweet_id     7274 non-null int64
tweet        7273 non-null object
sentiment    7274 non-null int64
dtypes: int64(2), object(1)
memory usage: 170.6+ KB
```

In [5]:

```
sns.countplot(x="sentiment",data=df)
```

Out[5]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1b3fe067c50>
```



Data Wrangling

In [6]:

```
df.isnull().sum()
```

Out[6]:

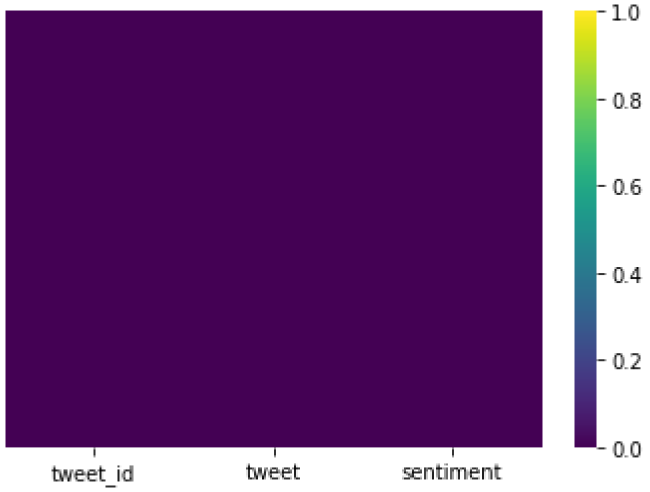
```
tweet_id     0
tweet        1
sentiment    0
dtype: int64
```

In [7]:

```
sns.heatmap(df.isnull(),yticklabels=False,cmap="viridis")
#1 Missing Value
```

Out[7]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1b38436c7b8>
```



In [8]:

```
df.drop("tweet_id",axis=1,inplace=True)
```

In [9]:

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

Out[9]:

	tweet	sentiment
0	#sxswnui #sxsw #apple defining language of tou...	1
1	Learning ab Google doodles! All doodles should...	1
2	one of the most in-your-face ex. of stealing t...	2
3	This iPhone #SXSW app would b pretty awesome i...	0
4	Line outside the Apple store in Austin waiting...	1
5	#technews One lone dude awaits iPad 2 at Apple...	1
6	SXSW Tips, Prince, NPR Videos, Toy Shopping Wi...	1
7	NU user RT @mention New #UberSocial for #iPhon...	1
8	Free #SXSW sampler on iTunes {link} #FreeMusic	2
9	I think I might go all weekend without seeing ...	2

In [10]:

```
df.dropna(inplace=True)
#Dropping The Column
```

In [11]:

```
df.isnull().sum()
```

Out[11]:

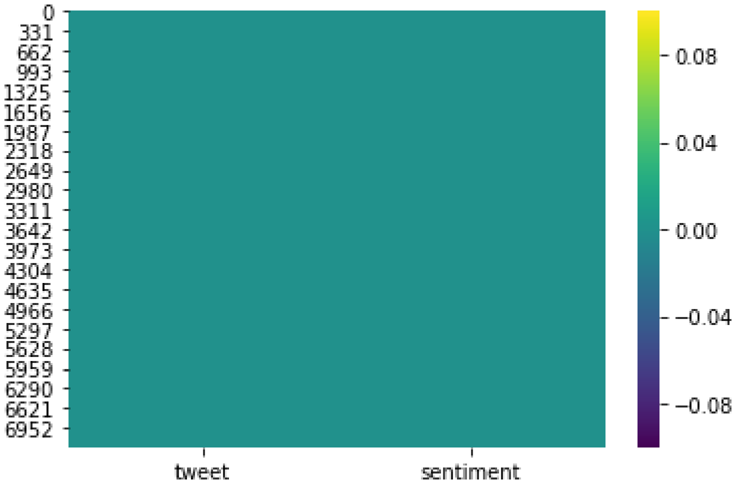
```
tweet        0
sentiment    0
dtype: int64
```

In [12]:

```
sns.heatmap(df.isnull(), linecolor="red",cmap="viridis")
# Perfectly Clean Data
```

Out[12]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1b384420cc0>
```



In [13]:

```
df.head(5)
```

Out[13]:

	tweet	sentiment
0	#sxswnui #sxsw #apple defining language of tou...	1
1	Learning ab Google doodles! All doodles should...	1
2	one of the most in-your-face ex. of stealing t...	2
3	This iPhone #SXSW app would b pretty awesome i...	0
4	Line outside the Apple store in Austin waiting...	1

Exploratory Data Analysis

In [14]:

```
df['word_count'] = df['tweet'].apply(lambda x: len(str(x).split(" ")))
df[['tweet','word_count']].head()
```

Out[14]:

	tweet	word_count
0	#sxswnui #sxsw #apple defining language of tou...	12
1	Learning ab Google doodles! All doodles should...	19
2	one of the most in-your-face ex. of stealing t...	23
3	This iPhone #SXSW app would b pretty awesome i...	19
4	Line outside the Apple store in Austin waiting...	15

In [15]:

```
df['char_count'] = df['tweet'].str.len() ## this also includes spaces
df[['tweet','char_count']].head()
```

Out[15]:

	tweet	char_count
0	#sxswnui #sxsw #apple defining language of tou...	89
1	Learning ab Google doodles! All doodles should...	143
2	one of the most in-your-face ex. of stealing t...	132
3	This iPhone #SXSW app would b pretty awesome i...	125
4	Line outside the Apple store in Austin waiting...	77

In [16]:

```
def avg_word(sentence):
  words = sentence.split()
  return (sum(len(word) for word in words)/len(words))

df['avg_word'] = df['tweet'].apply(lambda x: avg_word(x))
df[['tweet','avg_word']].head()
```

Out[16]:

	tweet	avg_word
0	#sxswnui #sxsw #apple defining language of tou...	6.500000
1	Learning ab Google doodles! All doodles should...	6.578947
2	one of the most in-your-face ex. of stealing t...	5.000000
3	This iPhone #SXSW app would b pretty awesome i...	5.631579
4	Line outside the Apple store in Austin waiting...	4.500000

In [17]:

```
stop = stopwords.words('english')
df['stopwords'] = df['tweet'].apply(lambda x: len([x for x in x.split() if x in stop]))
df[['tweet','stopwords']].head()
```

Out[17]:

	tweet	stopwords
0	#sxswnui #sxsw #apple defining language of tou...	2
1	Learning ab Google doodles! All doodles should...	4
2	one of the most in-your-face ex. of stealing t...	7
3	This iPhone #SXSW app would b pretty awesome i...	4
4	Line outside the Apple store in Austin waiting...	4

In [18]:

```
df['hastags'] = df['tweet'].apply(lambda x: len([x for x in x.split() if x.startswith('#')]
df[['tweet','hastags']].head()
```

Out[18]:

	tweet	hastags
0	#sxswnui #sxsw #apple defining language of tou...	3
1	Learning ab Google doodles! All doodles should...	2
2	one of the most in-your-face ex. of stealing t...	1
3	This iPhone #SXSW app would b pretty awesome i...	3
4	Line outside the Apple store in Austin waiting...	1

In [19]:

```
df['numerics'] = df['tweet'].apply(lambda x: len([x for x in x.split() if x.isdigit()]))
df[['tweet','numerics']].head()
#Total Number Present
```

Out[19]:

	tweet	numerics
0	#sxswnui #sxsw #apple defining language of tou...	0
1	Learning ab Google doodles! All doodles should...	0
2	one of the most in-your-face ex. of stealing t...	0
3	This iPhone #SXSW app would b pretty awesome i...	0
4	Line outside the Apple store in Austin waiting...	0

In [20]:

```
df['upper'] = df['tweet'].apply(lambda x: len([x for x in x.split() if x.isupper()]))
df[['tweet','upper']].head()
#Upper Case Characters Presnt in Datset
```

Out[20]:

	tweet	upper
0	#sxswnui #sxsw #apple defining language of tou...	0
1	Learning ab Google doodles! All doodles should...	0
2	one of the most in-your-face ex. of stealing t...	2
3	This iPhone #SXSW app would b pretty awesome i...	1
4	Line outside the Apple store in Austin waiting...	1

Data Preprocessing And Cleaning

In [21]:

```
df['tweet'] = df['tweet'].apply(lambda x: " ".join(x.lower() for x in x.split()))
df['tweet'].head()
#Making Everything in LowerCase No Repeatations
```

Out[21]:

```
0    #sxswnui #sxsw #apple defining language of tou...
1    learning ab google doodles! all doodles should...
2    one of the most in-your-face ex. of stealing t...
3    this iphone #sxsw app would b pretty awesome i...
4    line outside the apple store in austin waiting...
Name: tweet, dtype: object
```

In [22]:

```
df['tweet'] = df['tweet'].str.replace('[^\w\s]','')
df['tweet'].head()
#REMOVING THE PUNCTUCATION
```

Out[22]:

```
0    sxswnui sxsw apple defining language of touch ...
1    learning ab google doodles all doodles should ...
2    one of the most inyourface ex of stealing the ...
3    this iphone sxsw app would b pretty awesome if...
4    line outside the apple store in austin waiting...
Name: tweet, dtype: object
```

In [23]:

```
stop = stopwords.words('english')
df['tweet'] = df['tweet'].apply(lambda x: " ".join(x for x in x.split() if x not in stop))
df['tweet'].head()
# Removing Stopwords
```

Out[23]:

```
0    sxswnui sxsw apple defining language touch dif...
1    learning ab google doodles doodles light funny...
2    one inyourface ex stealing show yrs rt mention...
3    iphone sxsw app would b pretty awesome didnt c...
4    line outside apple store austin waiting new ip...
Name: tweet, dtype: object
```

In [24]:

```
freq = pd.Series(' '.join(df['tweet']).split()).value_counts()[:10]
#Commonly Used Words And Thier Count
```

In [25]:

```
freq
```

Out[25]:

```
sxsw       7540
mention    5512
link       3427
rt         2344
ipad       1912
google     1862
apple      1729
iphone     1215
store      1188
new         862
dtype: int64
```

In [26]:

```
freq = list(freq.index)
df['tweet'] = df['tweet'].apply(lambda x: " ".join(x for x in x.split() if x not in freq))
df['tweet'].head()
#Removing the Common Words
```

Out[26]:

```
0    sxswnui defining language touch different dial...
1    learning ab doodles doodles light funny amp in...
2    one inyourface ex stealing show yrs quotat sch...
3    app would b pretty awesome didnt crash every 1...
4                              line outside austin waiting
Name: tweet, dtype: object
```

In [27]:

```
freq1 = pd.Series(' '.join(df['tweet']).split()).value_counts()[-10:]
# Rare Words From Dataset
```

In [28]:

```
freq1
```

Out[28]:

```
soccomp           1
hooking           1
emily             1
mkesxsw           1
edreform          1
suggestionskind   1
guerrilla         1
sehugg            1
ipadssxswû        1
beforetwitter     1
dtype: int64
```

In [29]:

```
freq1 = list(freq1.index)
df['tweet'] = df['tweet'].apply(lambda x: " ".join(x for x in x.split() if x not in freq1))
df['tweet'].head()
#Removing Rare Words From Dataset
```

Out[29]:

```
0    sxswnui defining language touch different dial...
1    learning ab doodles doodles light funny amp in...
2    one inyourface ex stealing show yrs quotat sch...
3    app would b pretty awesome didnt crash every 1...
4                               line outside austin waiting
Name: tweet, dtype: object
```

In [30]:

```
df['tweet'][:5].apply(lambda x: str(TextBlob(x).correct()))
#Words Correction analytics and analtycs
```

Out[30]:

```
0    sxswnui defining language touch different dial...
1    learning ab doubles doubles light funny amp in...
2    one inyourface ex stealing show yes quotas sch...
3    pp would b pretty awesome didn crash every 10m...
4                               line outside austin waiting
Name: tweet, dtype: object
```

In [31]:

```
TextBlob(df['tweet'][1]).words
```

Out[31]:

```
WordList(['learning', 'ab', 'doodles', 'doodles', 'light', 'funny', 'amp',
'innovative', 'exceptions', 'significant', 'occasions', 'googledoodle'])
```

In [32]:

```
st = PorterStemmer()
df['tweet'][:5].apply(lambda x: " ".join([st.stem(word) for word in x.split()]))
#removal of suffices, like “ing”, “ly”, “s”, etc.
```

Out[32]:

```
0    sxswnui defin languag touch differ dialect bec...
1    learn ab doodl doodl light funni amp innov exc...
2    one inyourfac ex steal show yr quotat school m...
3    app would b pretti awesom didnt crash everi 10...
4                              line outsid austin wait
Name: tweet, dtype: object
```

Advanced Text Processing

In [34]:

```
TextBlob(df['tweet'][0]).ngrams(2)
#N-grams are the combination of multiple words used together.
```

Out[34]:

```
[WordList(['sxswnui', 'defining']),
 WordList(['defining', 'language']),
 WordList(['language', 'touch']),
 WordList(['touch', 'different']),
 WordList(['different', 'dialects']),
 WordList(['dialects', 'becoming']),
 WordList(['becoming', 'smaller'])]
```

In [36]:

```
tf1 = (df['tweet'][1:2]).apply(lambda x: pd.value_counts(x.split(" "))).sum(axis = 0).reset
tf1.columns = ['words','tf']
tf1
#Term frequency is simply the ratio of the count of a word present in a sentence, to the le
```

Out[36]:

	words	tf
0	doodles	2
1	innovative	1
2	googledoodle	1
3	significant	1
4	learning	1
5	ab	1
6	light	1
7	exceptions	1
8	amp	1
9	occasions	1
10	funny	1

In [39]:

```
for i,word in enumerate(tf1['words']):
  tf1.loc[i, 'idf'] = np.log(df.shape[0]/(len(df[df['tweet'].str.contains(word)])))
tf1
#The intuition behind inverse document frequency (IDF) is that a word is not of much use to
#in all the documents.
```

Out[39]:

	words	tf	idf
0	doodles	2	5.800882
1	innovative	1	7.793312
2	googledoodle	1	6.183874
3	significant	1	8.891924
4	learning	1	6.326975
5	ab	1	2.787131
6	light	1	4.687232
7	exceptions	1	8.891924
8	amp	1	2.349452
9	occasions	1	8.891924
10	funny	1	5.896192

In [41]:

```
tf1['tfidf'] = tf1['tf'] * tf1['idf']
tf1
#TF-IDF is the multiplication of the TF and IDF which we calculated above.
```

Out[41]:

	words	tf	idf	tfidf
0	doodles	2	5.800882	11.601763
1	innovative	1	7.793312	7.793312
2	googledoodle	1	6.183874	6.183874
3	significant	1	8.891924	8.891924
4	learning	1	6.326975	6.326975
5	ab	1	2.787131	2.787131
6	light	1	4.687232	4.687232
7	exceptions	1	8.891924	8.891924
8	amp	1	2.349452	2.349452
9	occasions	1	8.891924	8.891924
10	funny	1	5.896192	5.896192

Model

In [43]:

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf = TfidfVectorizer(max_features=1000, lowercase=True, analyzer='word',
 stop_words= 'english',ngram_range=(1,1))
train_vect = tfidf.fit_transform(df['tweet'])

train_vect
```

Out[43]:

```
<7273x1000 sparse matrix of type '<class 'numpy.float64'>'
	with 36485 stored elements in Compressed Sparse Row format>
```

In [44]:

```
from sklearn.feature_extraction.text import CountVectorizer
bow = CountVectorizer(max_features=1000, lowercase=True, ngram_range=(1,1),analyzer = "word
train_bow = bow.fit_transform(df['tweet'])
train_bow
#Bag of Words
```

Out[44]:

```
<7273x1000 sparse matrix of type '<class 'numpy.int64'>'
	with 40110 stored elements in Compressed Sparse Row format>
```

In [45]:

```
df['tweet'][:5].apply(lambda x: TextBlob(x).sentiment)
```

Out[45]:

```
0          (0.15, 0.65)
1    (0.38125, 0.89375)
2            (0.0, 0.0)
3          (0.625, 1.0)
4           (0.0, 0.05)
Name: tweet, dtype: object
```

In [46]:

```
df['sentiment'] = df['tweet'].apply(lambda x: TextBlob(x).sentiment[0] )
df[['tweet','sentiment']].head()
```

Out[46]:

	tweet	sentiment
0	sxswnui defining language touch different dial...	0.15000
1	learning ab doodles doodles light funny amp in...	0.38125
2	one inyourface ex stealing show yrs quotat sch...	0.00000
3	app would b pretty awesome didnt crash every 1...	0.62500
4	line outside austin waiting	0.00000

Testing

In [49]:

```
df1.head(4)
```

Out[49]:

	tweet_id	tweet
0	7506	Audience Q: What prototyping tools do you use?...
1	7992	At SXSW? Send Your Best Photos &amp; Videos to...
2	247	@mention and here's a pic of you winning your...
3	7688	Google Marissa Mayer: mobile phone as a cursor...

In [51]:

```
df1['sentiment'] = df1['tweet'].apply(lambda x: TextBlob(x).sentiment[0] )
df[['tweet','sentiment']].head()
```

Out[51]:

	tweet	sentiment
0	#sxswnui #sxsw #apple defining language of tou...	1
1	Learning ab Google doodles! All doodles should...	1
2	one of the most in-your-face ex. of stealing t...	2
3	This iPhone #SXSW app would b pretty awesome i...	0
4	Line outside the Apple store in Austin waiting...	1

In [ ]:

```

```