SENTIMENT ANALYSIS FOR MARKETING

PHASE 3

NAME	SATTANATHAN V
TEAM ID	proj_212173_Team_2

Project : Sentient Analysis For Marketing

DATA VISUALIZATION:

Data visualization in sentiment analysis is the combination of these two processes, where the results of sentiment analysis are displayed in a visual form that can facilitate analysis and decision making. For example, data visualization in sentiment analysis can help to

- Compare the overall sentiment (positive, negative, or neutral) of different groups of customers, products, topics, or time periods.
- Identify the most common words or phrases that are associated with positive or negative sentiment.
- Explore the distribution and variation of sentiment scores across different categories or dimensions.
- Track the changes and trends of sentiment over time

PROGRAM:

SENTIMENTAL ANALSIS FOR MARKETING

Importing Libraries:

```
import pandas as pd
import seaborn as sns
import re, nltk
nltk.download('punkt')
import matplotlib.pyplot as plt
from sklearn import model_selection, naive_bayes, svm
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.model_selection import GridSearchCV
```

```
from matplotlib import pyplot
import string
from nltk.corpus import stopwords
nltk.download('stopwords')
import numpy as np
from lime import lime_tabular
from tensorflow.keras.layers import Embedding
from tensorflow.keras.layers import LSTM, Bidirectional
from tensorflow.keras.layers import Dense, Dropout

import warnings
warnings.filterwarnings('ignore')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\ELCOT\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\ELCOT\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

```
#DATA LOADING

tweets_df =pd.read_csv('Tweets.csv')
tweets= tweets_df.copy()
tweets_df.head()
```

```
             tweet_id airline_sentiment  airline_sentiment_confidence
\
0  570306133677760513           neutral                        1.0000
1  570301130888122368          positive                        0.3486
2  570301083672813571           neutral                        0.6837
3  570301031407624196          negative                        1.0000
4  570300817074462722          negative                        1.0000

  negativereason  negativereason_confidence         airline  \
0            NaN                        NaN  Virgin America
1            NaN                     0.0000  Virgin America
2            NaN                        NaN  Virgin America
3     Bad Flight                     0.7033  Virgin America
4     Can't Tell                     1.0000  Virgin America

  airline_sentiment_gold        name negativereason_gold
retweet_count  \
0                    NaN     cairdin                 NaN
0
1                    NaN    jnardino                 NaN
```

```
0
2                    NaN  yvonnalynn                NaN
0
3                    NaN    jnardino                NaN
0
4                    NaN    jnardino                NaN
0

                                                    text tweet_coord  \
0                @VirginAmerica What @dhepburn said.         NaN
1  @VirginAmerica plus you've added commercials t...         NaN
2  @VirginAmerica I didn't today... Must mean I n...         NaN
3  @VirginAmerica it's really aggressive to blast...         NaN
4  @VirginAmerica and it's a really big bad thing...         NaN

              tweet_created tweet_location
user_timezone
0  2015-02-24 11:35:52 -0800            NaN  Eastern Time (US &
Canada)
1  2015-02-24 11:15:59 -0800            NaN  Pacific Time (US &
Canada)
2  2015-02-24 11:15:48 -0800      Lets Play  Central Time (US &
Canada)
3  2015-02-24 11:15:36 -0800            NaN  Pacific Time (US &
Canada)
4  2015-02-24 11:14:45 -0800            NaN  Pacific Time (US &
Canada)
```

```
#Data columns
```

```
tweets_df.columns
```

```
Index(['tweet_id', 'airline_sentiment',
'airline_sentiment_confidence',
       'negativereason', 'negativereason_confidence', 'airline',
       'airline_sentiment_gold', 'name', 'negativereason_gold',
       'retweet_count', 'text', 'tweet_coord', 'tweet_created',
       'tweet_location', 'user_timezone'],
      dtype='object')
```

```
tweets_df['airline_sentiment'].unique()
```

```
array(['neutral', 'positive', 'negative'], dtype=object)
```

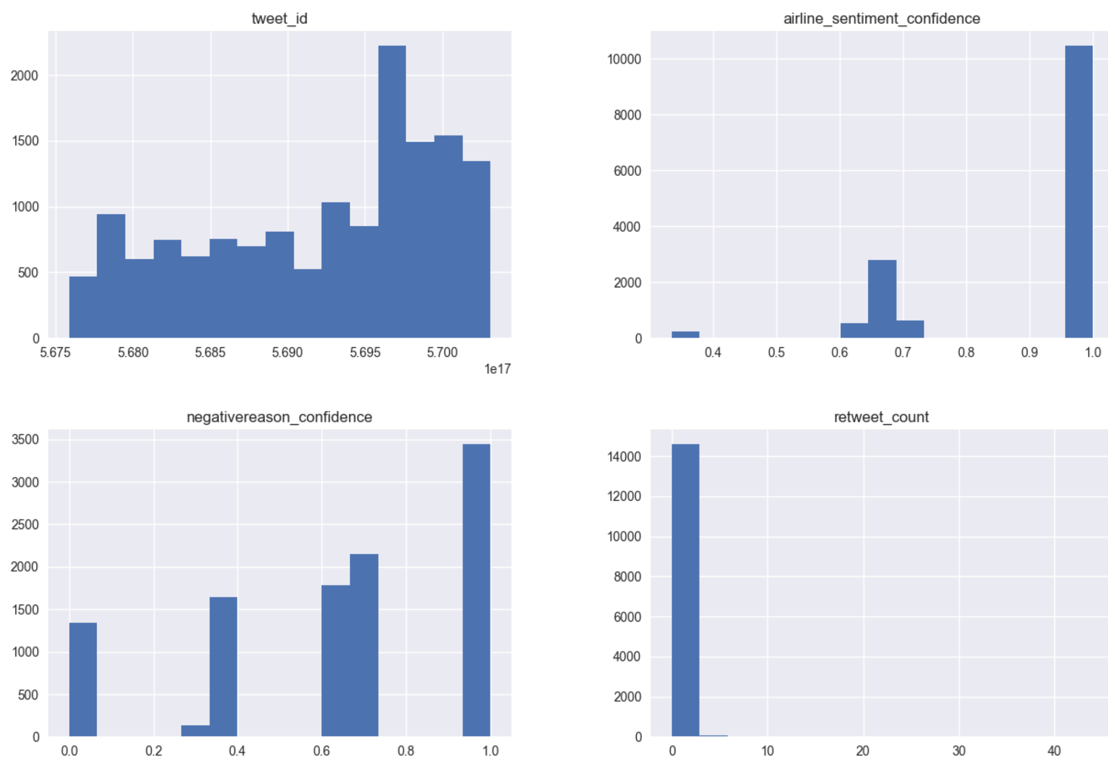
```
tweets_df['airline_sentiment'].value_counts()
```

```
negative    9178
neutral     3099
```

```
positive    2363
Name: airline_sentiment, dtype: int64
```

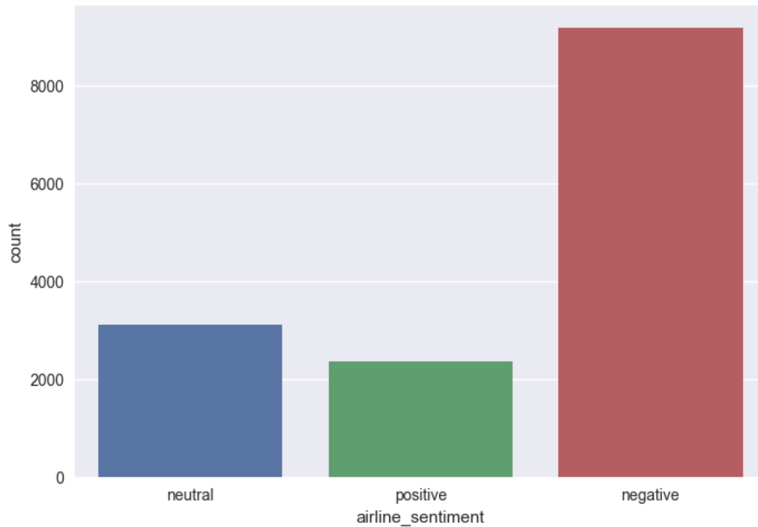
```
#Data Visualization
plt.style.use("seaborn")
tweets_df.hist(figsize=(15,10),bins=15)
```

```
array([[<AxesSubplot: title={'center': 'tweet_id'}>,
        <AxesSubplot: title={'center': 'airline_sentiment_confidence'}
>],
       [<AxesSubplot: title={'center': 'negativereason_confidence'}>,
        <AxesSubplot: title={'center': 'retweet_count'}>]],
dtype=object)
```



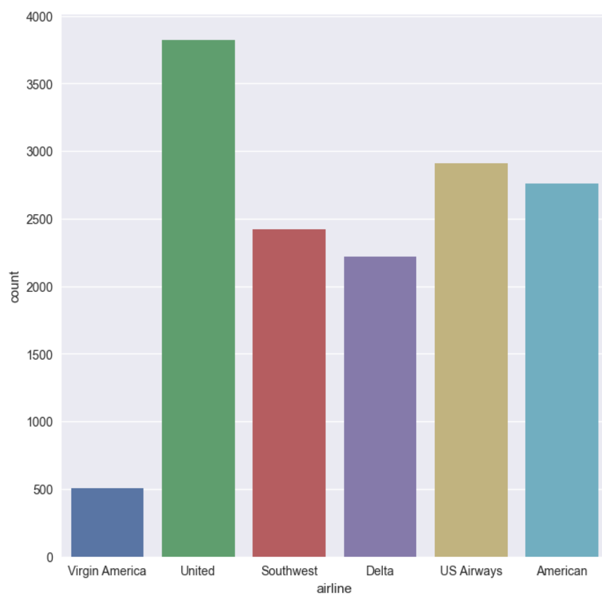

```
#COUNT PLOT

sns.countplot(x="airline_sentiment", data=tweets_df)
```

```
<AxesSubplot: xlabel='airline_sentiment', ylabel='count'>
```



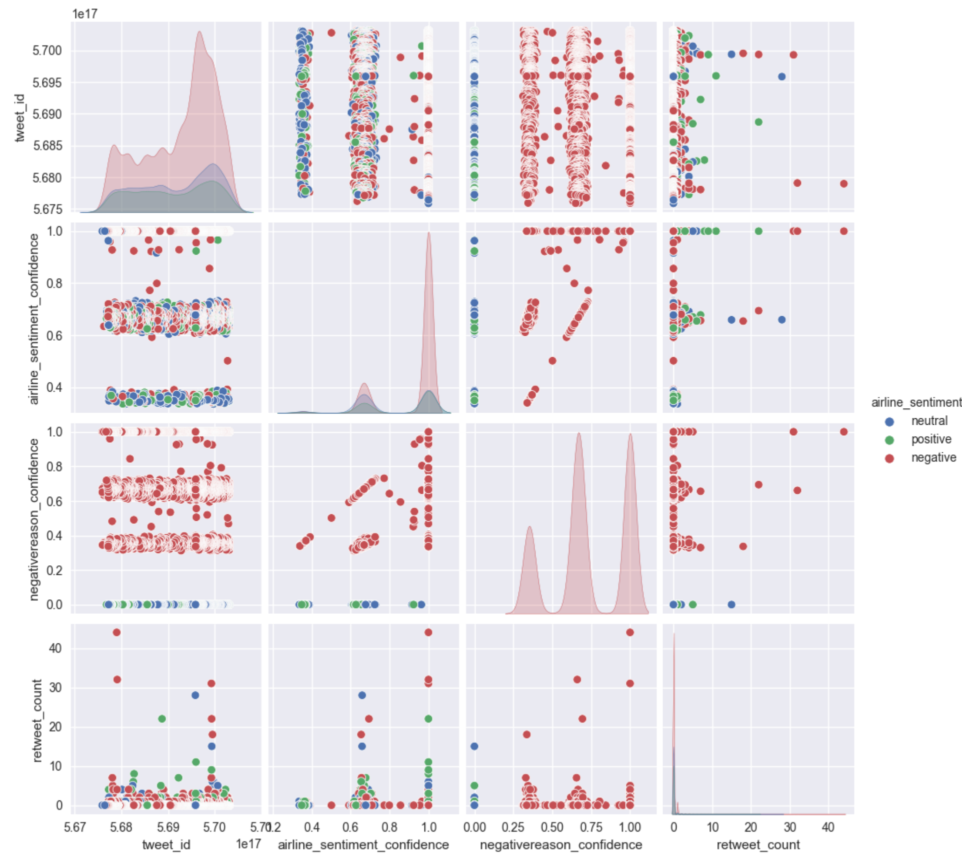

```
plt.figure(figsize=(8,8))
sns.countplot(x="airline", data=tweets_df)
```

```
<AxesSubplot: xlabel='airline', ylabel='count'>
```




```
sns.pairplot(tweets_df,hue='airline_sentiment')
```

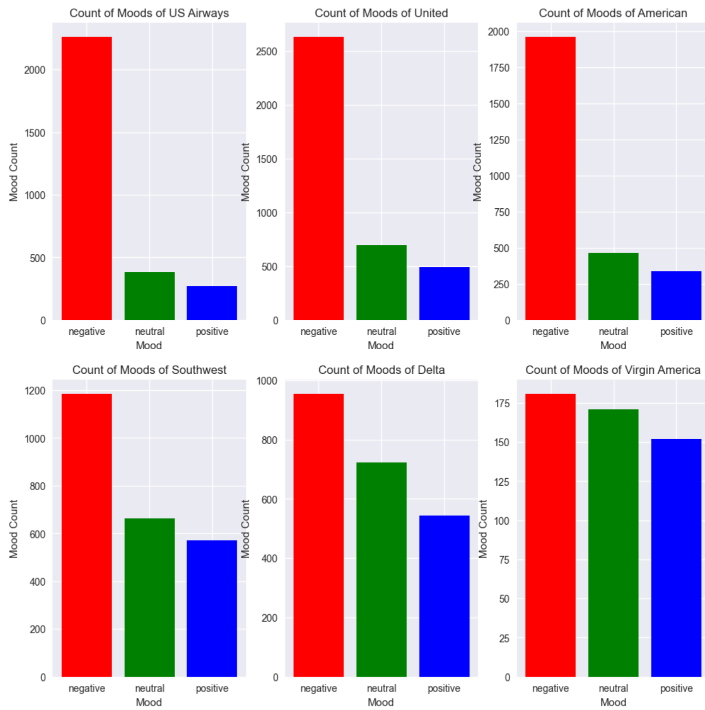
```
<seaborn.axisgrid.PairGrid at 0x211c88598d0>
```




```
print("Total number of tweets for each airline \n
",tweets_df.groupby('airline')
['airline_sentiment'].count().sort_values(ascending=False))
airlines= ['US
Airways','United','American','Southwest','Delta','Virgin America']
plt.figure(1,figsize=(12, 12))
for i in airlines:
    indices= airlines.index(i)
    plt.subplot(2,3,indices+1)
    new_df=tweets_df[tweets_df['airline']==i]
    count=new_df['airline_sentiment'].value_counts()
    Index = [1,2,3]
    plt.bar(Index,count, color=['red', 'green', 'blue'])
    plt.xticks(Index,['negative','neutral','positive'])
    plt.ylabel('Mood Count')
    plt.xlabel('Mood')
    plt.title('Count of Moods of '+i)
```

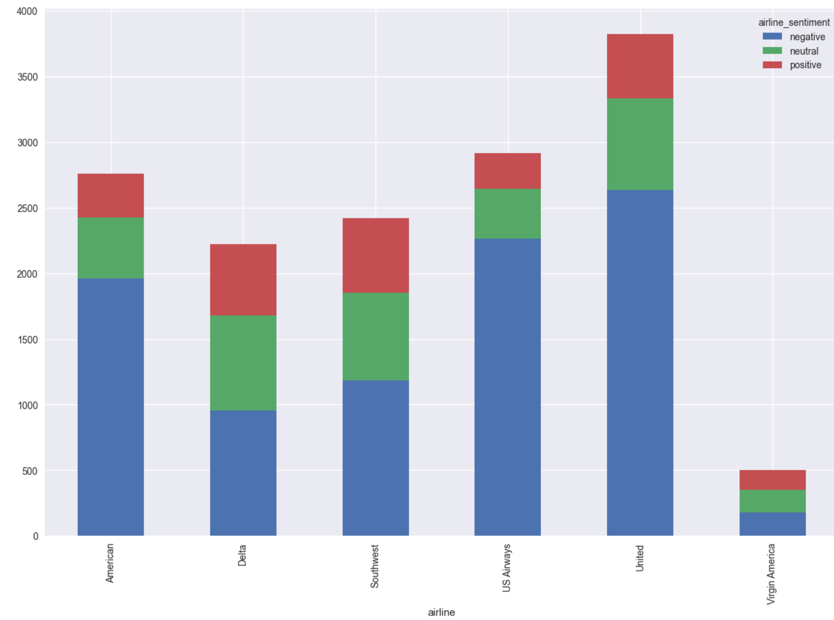
```
Out:

Total number of tweets for each airline
  airline
United            3822
```

```
US Airways        2913
American          2759
Southwest         2420
Delta             2222
Virgin America     504
Name: airline_sentiment, dtype: int64
```

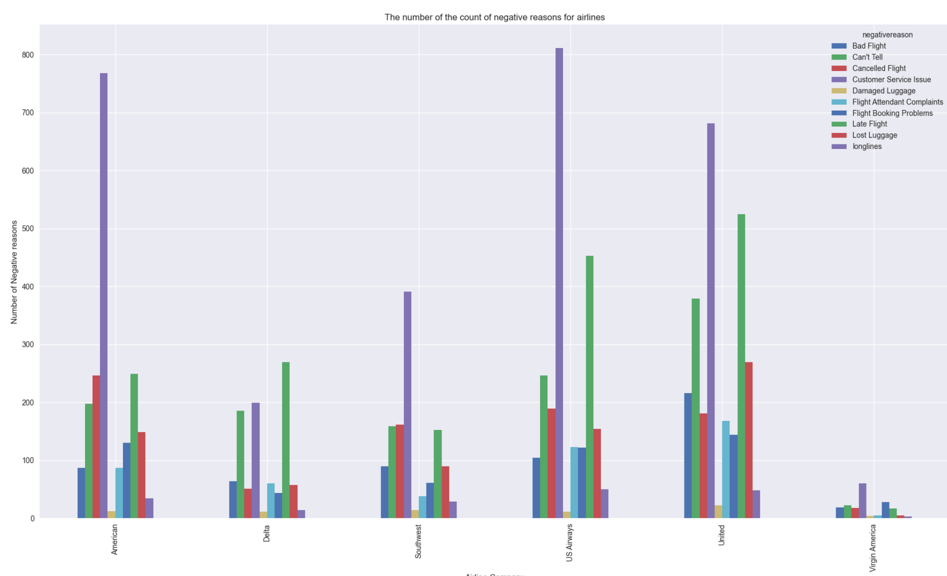



```
figure_2 = tweets_df.groupby(['airline', 'airline_sentiment']).size()
figure_2.unstack().plot(kind='bar', stacked=True, figsize=(15,10))
```

```
<AxesSubplot: xlabel='airline'>
```

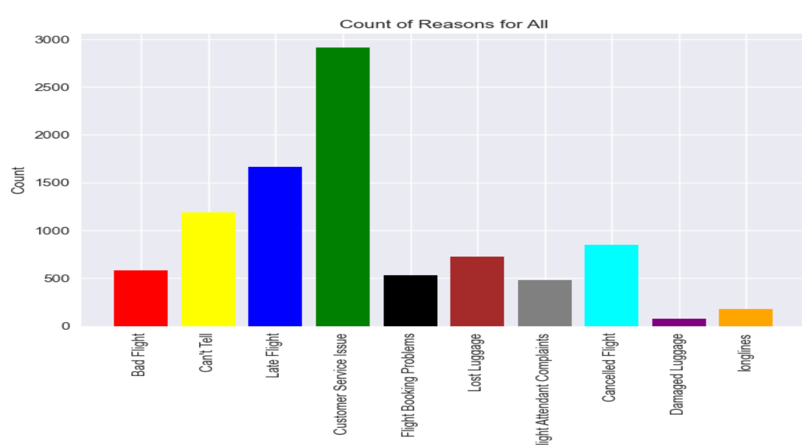


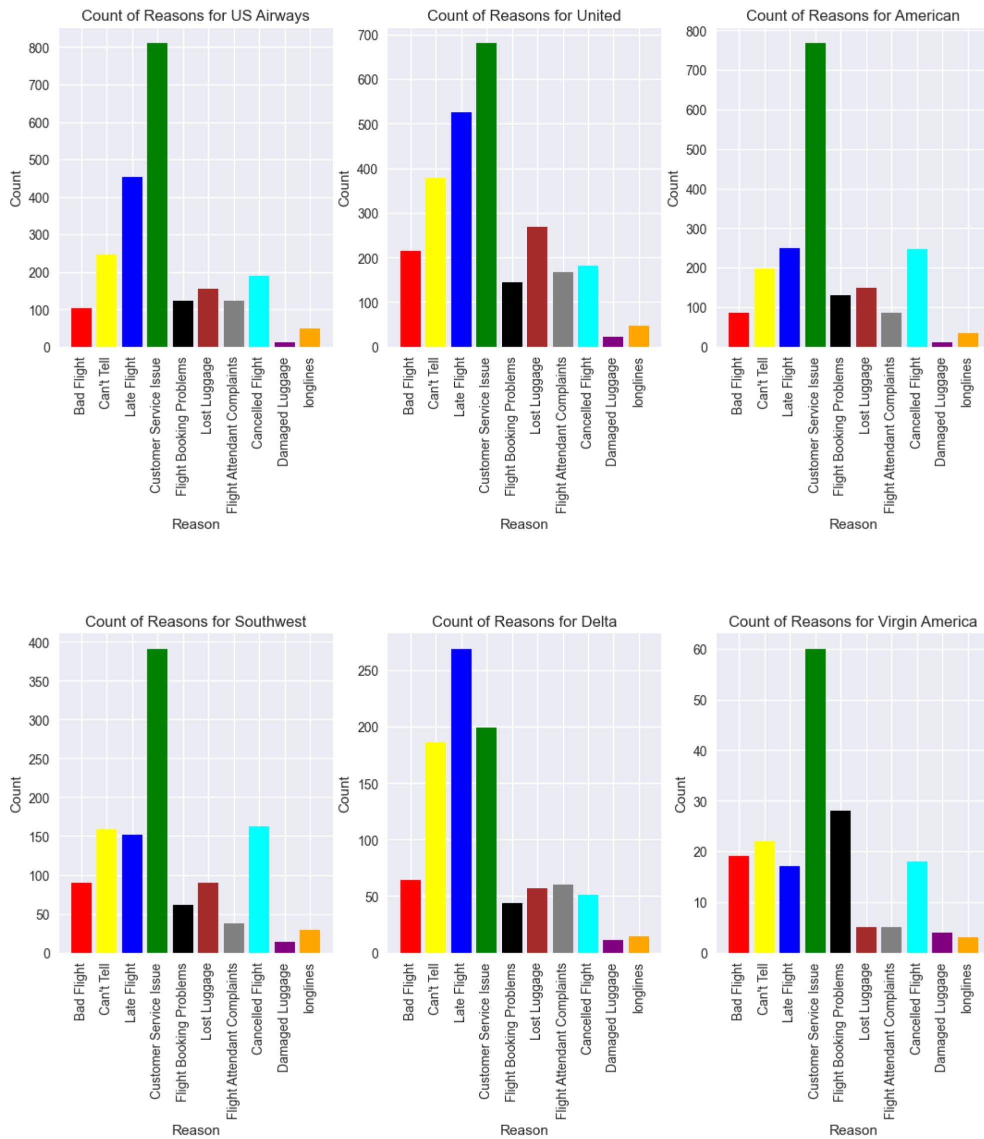

```
negative_reasons = tweets_df.groupby('airline')
['negativereason'].value_counts(ascending=True)
negative_reasons.groupby(['airline','negativereason']).sum().unstack()
.plot(kind='bar',figsize=(22,12))
plt.xlabel('Airline Company')
plt.ylabel('Number of Negative reasons')
plt.title("The number of the count of negative reasons for airlines")
plt.show()
```




```
tweets_df['negativereason'].nunique()
```

```
NR_Count=dict(tweets_df['negativereason'].value_counts(sort=False))
def NR_Count(Airline):
    if Airline=='All':
        a=tweets_df
    else:
        a=tweets_df[tweets_df['airline']==Airline]
    count=dict(a['negativereason'].value_counts())
    Unique_reason=list(tweets_df['negativereason'].unique())
    Unique_reason=[x for x in Unique_reason if str(x) != 'nan']
    Reason_frame=pd.DataFrame({'Reasons':Unique_reason})
    Reason_frame['count']=Reason_frame['Reasons'].apply(lambda x:
count[x])
    return Reason_frame
def plot_reason(Airline):

    a=NR_Count(Airline)
    count=a['count']
    Index = range(1,(len(a)+1))
    plt.bar(Index,count,
color=['red','yellow','blue','green','black','brown','gray','cyan','pu
rple','orange'])
    plt.xticks(Index,a['Reasons'],rotation=90)
    plt.ylabel('Count')
    plt.xlabel('Reason')
    plt.title('Count of Reasons for '+Airline)

plot_reason('All')
plt.figure(2,figsize=(13, 13))
for i in airlines:
    indices= airlines.index(i)
    plt.subplot(2,3,indices+1)
    plt.subplots_adjust(hspace=0.9)
    plot_reason(i)
```



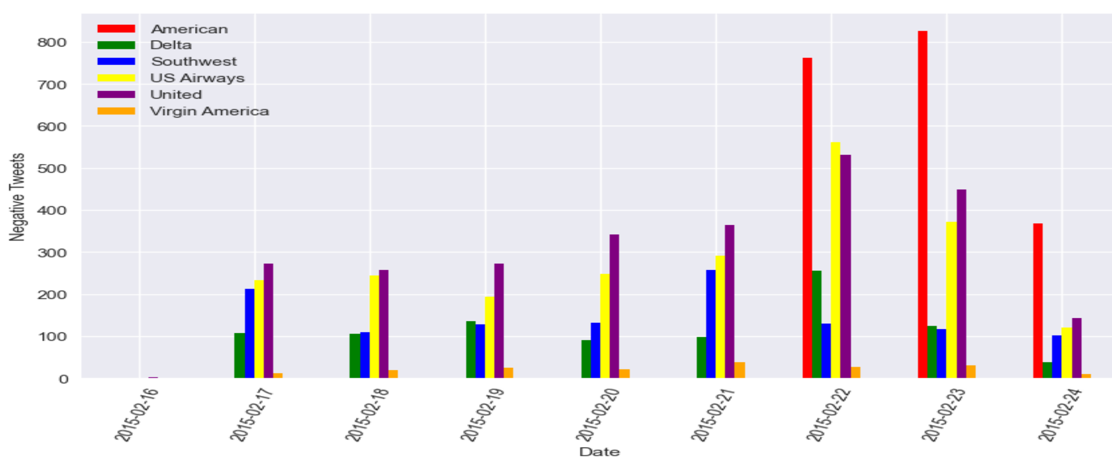



```
date = tweets_df.reset_index()
#convert the Date column to pandas datetime
date.tweet_created = pd.to_datetime(date.tweet_created)
#Reduce the dates in the date column to only the date and no time
stamp using the 'dt.date' method
date.tweet_created = date.tweet_created.dt.date
date.tweet_created.head()
df = date
day_df =
df.groupby(['tweet_created','airline','airline_sentiment']).size()
day_df
```

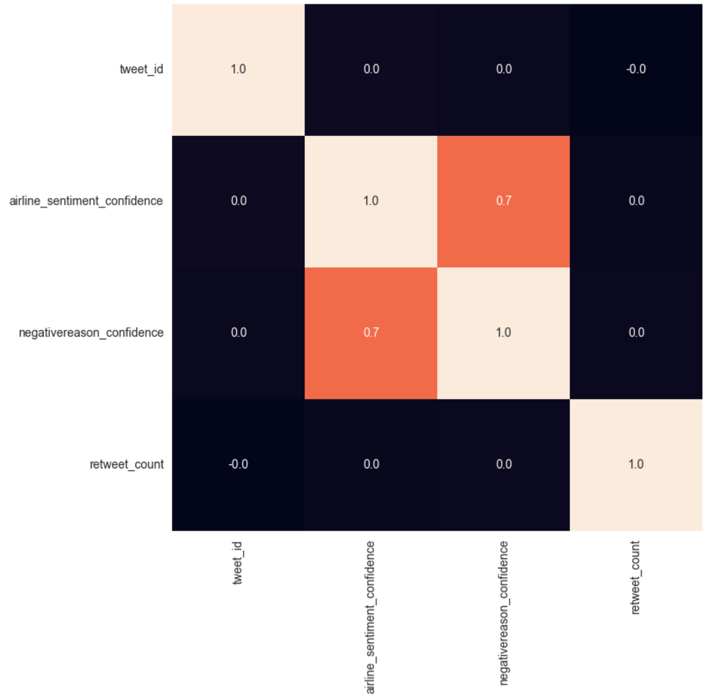
```
tweet_created   airline          airline_sentiment
2015-02-16      Delta            negative              1
                                 neutral               1
```

```
                United          negative            2
2015-02-17      Delta           negative          108
                                neutral            86
                                                  ...
2015-02-24      United          neutral            49
                                positive           25
                Virgin America  negative           10
                                neutral             6
                                positive           13
Length: 136, dtype: int64
```

```
day_df = day_df.loc(axis=0)[:,:,'negative']

#groupby and plot data
ax2 =
day_df.groupby(['tweet_created','airline']).sum().unstack().plot(kind
= 'bar', color=['red', 'green', 'blue','yellow','purple','orange'],
figsize = (10,6), rot = 70)
labels = ['American','Delta','Southwest','US Airways','United','Virgin
America']
ax2.legend(labels = labels)
ax2.set_xlabel('Date')
ax2.set_ylabel('Negative Tweets')
plt.show()
```

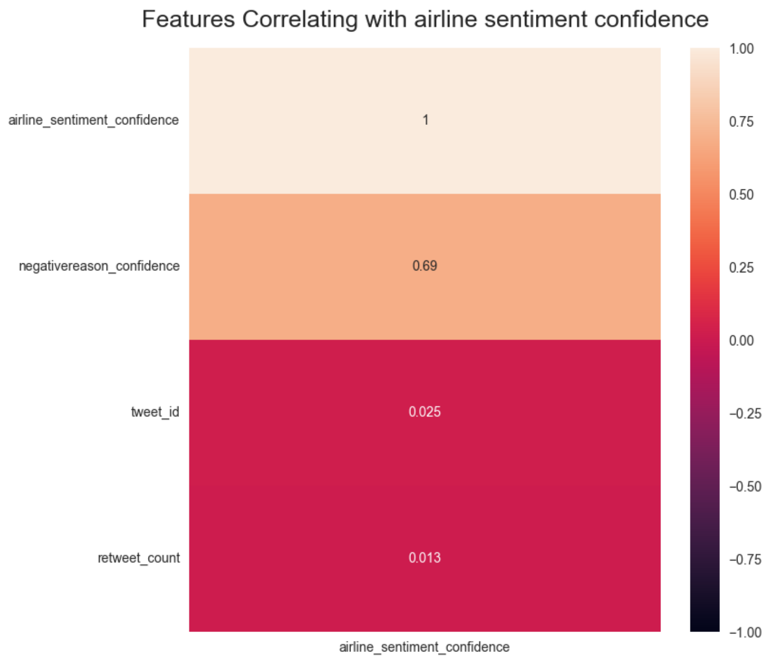



```
#Heatmap
plt.figure(figsize=(8,8))
sns.heatmap(tweets_df.corr(),annot=True,cbar=False,fmt='.1f')
plt.show()
```

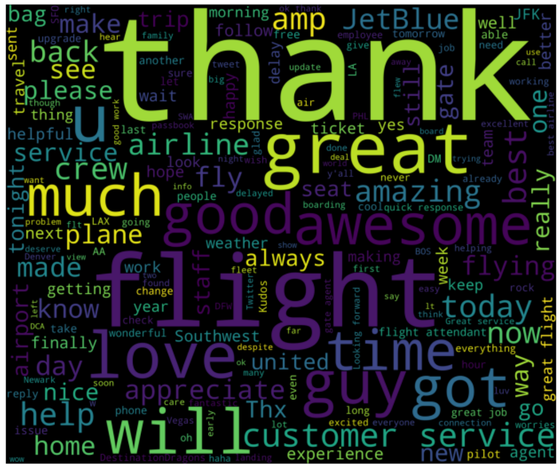



```
plt.figure(figsize=(8, 8))
heatmap = sns.heatmap(tweets_df.corr()
[['airline_sentiment_confidence']].sort_values(by='airline_sentiment_c
onfidence', ascending=False), vmin=-1, vmax=1, annot=True)
heatmap.set_title('Features Correlating with airline sentiment
confidence', fontdict={'fontsize':18}, pad=16)
```

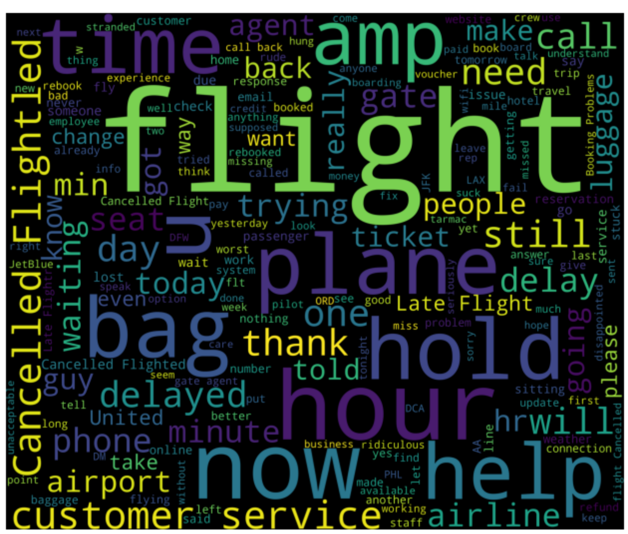
```
Text(0.5, 1.0, 'Features Correlating with airline sentiment
confidence')
```




```
from wordcloud import WordCloud,STOPWORDS
new_df=tweets_df[tweets_df['airline_sentiment']=='positive']
words = ' '.join(new_df['text'])
cleaned_word = " ".join([word for word in words.split()
                            if 'http' not in word
                                and not word.startswith('@')
                                and word != 'RT'
                            ])
wordcloud = WordCloud(stopwords=STOPWORDS,
            background_color='black',
width=3000,height=2500).generate(cleaned_word)
plt.figure(1,figsize=(8, 8))
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```

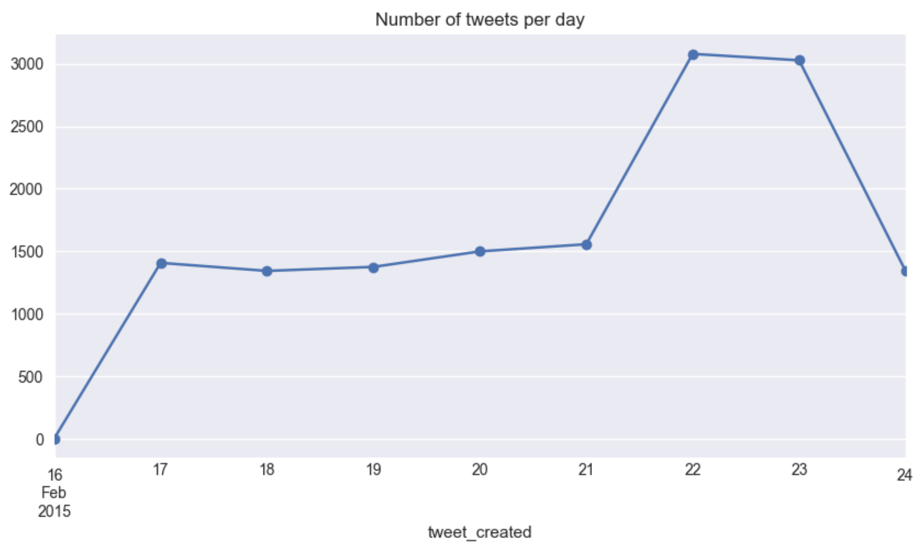



```
new_df=tweets_df[tweets_df['airline_sentiment']=='negative']
words = ' '.join(new_df['text'])
cleaned_word = " ".join([word for word in words.split()
                            if 'http' not in word
                                and not word.startswith('@')
                                and word != 'RT'
                            ])
wordcloud = WordCloud(stopwords=STOPWORDS,
                      background_color='black',
width=3000,height=2500).generate(cleaned_word)
plt.figure(1,figsize=(8,8))
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```



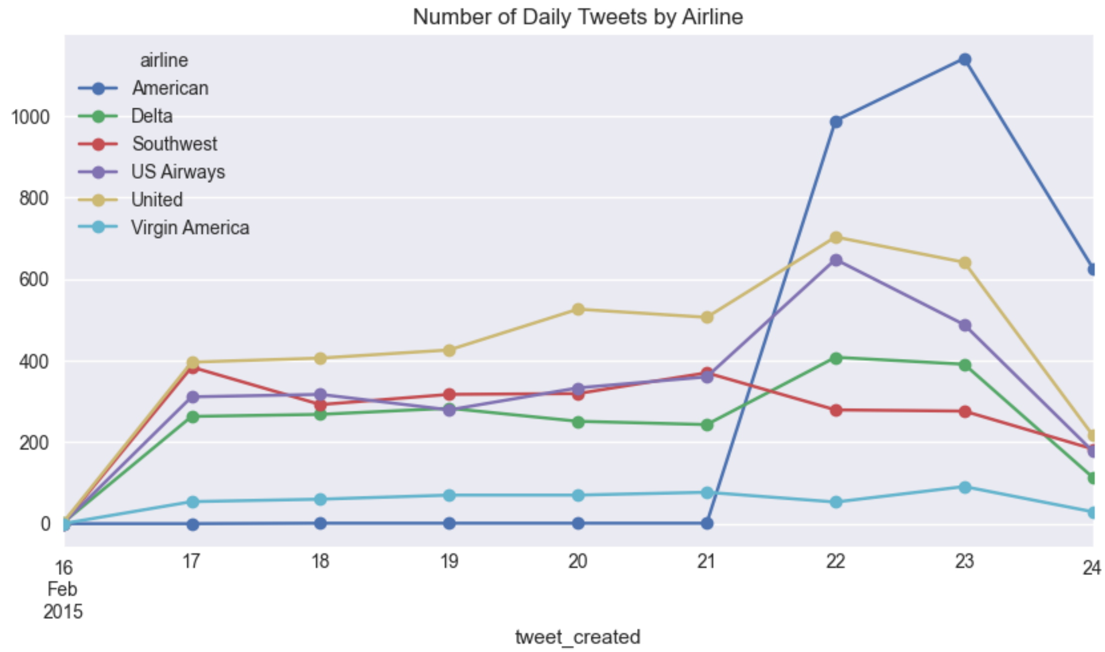
```python
tweets['tweet_created'] = pd.to_datetime(tweets['tweet_created'])
tweets_time_index = tweets.copy()
tweets_time_index.set_index("tweet_created", inplace=True)

tweets_time_index.resample("D")['tweet_id'].count().plot(style="-o",
figsize=(8, 5), title="Number of tweets per day")
```

```
<AxesSubplot: title={'center': 'Number of tweets per day'},
xlabel='tweet_created'>
```




```python
tweets_time_index =
tweets_time_index.pivot_table(index="tweet_created",columns="airline",
values="tweet_id", aggfunc=np.count_nonzero, fill_value=0)
tweets_time_index.resample("D").sum().plot(style="-o", figsize=(10,
5),title="Number of Daily Tweets by Airline")
```

```
<AxesSubplot: title={'center': 'Number of Daily Tweets by Airline'},
xlabel='tweet_created'>
```



Conclusion:

In the quest to build a sentiment analysis for marketing, we have embarked on a critical journey that begins with loading and preprocessing the dataset. We have traversed through essential steps, starting with importing the necessary libraries to facilitate data manipulation and analysis.

Understanding the data's structure, characteristics, and any potential issues through exploratory data analysis (EDA) is essential for informed decision-making.

Data preprocessing emerged as a pivotal aspect of this process. It involves cleaning, transforming, and refining the dataset to ensure that it aligns with the requirements of machine learning algorithms.