Phase:3

Sentiment analysis for marketing

Date	18-10-2023
Team ID	Proj_212173 Team 1
Project Name	Sentiment analysis for marketing

program

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
import string
import re
!pip install demoji
import demoji
```

Collecting demoji
Downloading demoji-1.1.0-py3-none-any.whl (42 kB)
━━
42.9/42.9 kB 2.1 MB/s eta 0:00:00
Installing collected packages: demoji
Successfully installed demoji-1.1.0
CodeText

```
df=pd.read_csv('Tweets.csv')
df.head()
```

	negativereason_confidence	airline	airline_sentiment_gold	name	negativereason_gold	retweet_count	text	tweet_coord	tweet_created	tweet_location	user_timezone
0	NaN	Virgin America	NaN	cairdin	NaN	0	@VirginAmerica Yhwr @dhepburn said.	NaN	2015-02-24 11:35:52 -0800	NaN	Eastern Time (US & Canada)
1	0.3000	Virgin America	NaN	jnardino	NaN	0	@VirginAmerica plus you've added commercials t...	NaN	2015-02-24 11:15:59 -0800	NaN	Pacific Time (US & Canada)
2	NaN	Virgin America	NaN	yorunalynn	NaN	0	@VirginAmerica I didn't today... Must mean I n...	NaN	2015-02-24 11:15:48 -0800	Lets Play	Central Time (US & Canada)
3	0.7000	Virgin America	NaN	jnardino	NaN	0	@VirginAmerica It's really aggressive to blast...	NaN	2015-02-24 11:15:36 -0800	NaN	Pacific Time (US & Canada)
4	1.0000	Virgin America	NaN	jnardino	NaN	0	@VirginAmerica and it's a really big bad thing...	NaN	2015-02-24 11:14:45 -0800	NaN	Pacific Time (US & Canada)

df.tail()

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	airline_sentiment_gold	r
14635	569587686496825344	positive	0.3487	NaN	0.0000	American	NaN	KristenReenders
14636	569587371693355008	negative	1.0000	Customer Service Issue	1.0000	American	NaN	itsropes
14637	569587242672398336	neutral	1.0000	NaN	NaN	American	NaN	sanyabun
14638	569587188687634433	negative	1.0000	Customer Service Issue	0.6659	American	NaN	SraJackson
14639	569587140490866689	neutral	0.6771	NaN	0.0000	American	NaN	daviddtwu

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>

RangeIndex: 14640 entries, 0 to 14639
Data columns (total 15 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   tweet_id                      14640 non-null  int64
 1   airline_sentiment             14640 non-null  object
 2   airline_sentiment_confidence  14640 non-null  float64
 3   negativereason                9178 non-null   object
 4   negativereason_confidence     10522 non-null  float64
 5   airline                       14640 non-null  object
 6   airline_sentiment_gold        40 non-null     object
 7   name                          14640 non-null  object
 8   negativereason_gold           32 non-null     object
 9   retweet_count                 14640 non-null  int64
 10  text                          14640 non-null  object
 11  tweet_coord                   1019 non-null   object
 12  tweet_created                 14640 non-null  object
 13  tweet_location                9907 non-null   object
 14  user_timezone                 9820 non-null   object
dtypes: float64(2), int64(2), object(11)
memory usage: 1.7+ MB
```

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

```
tweet_id                      0

airline_sentiment             0

airline_sentiment_confidence 0

negativereason               5462

negativereason_confidence    4118

airline                       0

airline_sentiment_gold       14600

 name                         0
```

```
negativereason_gold 14608
retweet_count          0
text 0
tweet_coord          13621
tweet_created          0
tweet_location       4733
user_timezone        4820
dtype: int64
```

```
df['airline_sentiment_confidence'].fillna(df['airline_sentiment_confidence'].mean(), inplace=True)
df['negativereason_confidence'].fillna(df['negativereason_confidence'].median(), inplace=True)
df['negativereason'].fillna(df['negativereason'].mode(),inplace=True)
df['user_timezone'].fillna(method='ffill', inplace=True)
col=["negativereason_gold","airline_sentiment_gold","tweet_coord","tweet_location"]
df.drop(col,axis=1,inplace=True)
df['negativereason'].fillna('No text', inplace=True)
df.isnull().sum()
```

```
tweet_id                      0
airline_sentiment             0
airline_sentiment_confidence  0
negativereason                0
negativereason_confidence     0
airline                       0
name                          0
retweet_count                 0
text                          0
tweet_created                 0
user_timezone                 0
dtype: int64
```

```
df['text']
```

```
0              @VirginAmerica What @dhepburn said.
1      @VirginAmerica plus you've added commercials t...
2      @VirginAmerica I didn't today... Must mean I n...
3      @VirginAmerica it's really aggressive to blast...
4      @VirginAmerica and it's a really big bad thing...
                    ...
14635   @AmericanAir thank you we got on a different f...
14636   @AmericanAir leaving over 20 minutes Late Flig...
14637   @AmericanAir Please bring American Airlines to...
14638   @AmericanAir you have my money, you change my ...
14639   @AmericanAir we have 8 ppl so we need 2 know h...
Name: text, Length: 14640, dtype: object
```

```
def clean_txt(text):
    # Remove all non-alphanumeric characters (except spaces)
    text=re.sub(r'@[a-zA-Z0-9]+','',text)#removes user
    #text=re.sub(r'[\s]', '', text)
    text=re.sub(r'#\w+','',text)
    text=re.sub(r'https?:/\/\s+','',text)#removes URL
    text=re.sub(r'RT[\s]+','',text)#removes retweet
    return text
df['new_text']=df['new_text'].astype(str).apply(clean_txt)
df['new_text']
```

```
0                          what said.
1      plus you've added commercials to the experien...
2      i didn't today... must mean i need to take an...
3      it's really aggressive to blast obnoxious "en...
4           and it's a really big bad thing about it
                  ...
14635    thank you we got on a different flight to chi...
14636    leaving over 20 minutes late flight. no warni...
14637              please bring american airlines to
14638    you have my money, you change my flight, and ...
14639    we have 8 ppl so we need 2 know how many seat...
Name: new_text, Length: 14640, dtype: object
```

CodeText

```
def remove_punctuation(text):
    return ''.join([char for char in text if char not in
string.punctuation])

df['new_text'] = df['new_text'].apply(remove_punctuation)
df['new_text']
```

```
0      what said
1      plus youve added commercials to the experienc...
2      i didnt today must mean i need to take another...
3      its really aggressive to blast obnoxious enter...
4      and its a really big bad thing about it ...
14635 thank you we got on a different flight to chi...
14636 leaving over 20 minutes late flight no warning... 14637
5      please bring american airlines to
14638 you have my money you change my flight and do... 14639
6     we have 8 ppl so we need 2 know how many seat...
Name: new_text, Length: 14640, dtype: object
```

```
nltk.download('punkt')
from nltk.tokenize import word_tokenize
def tokenize_text(text):

    tokens = word_tokenize(text)
    return tokens

df['new_text'] = df['new_text'].astype(str).apply(word_tokenize)

df['new_text']
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
0                                [what, said]
1          [plus, youve, added, commercials, to, the, exp...
2          [i, didnt, today, must, mean, i, need, to, tak...
3          [its, really, aggressive, to, blast, obnoxious...
4          [and, its, a, really, big, bad, thing, about, it]
                        ...
14635   [thank, you, we, got, on, a, different, flight...
14636   [leaving, over, 20, minutes, late, flight, no,...
14637          [please, bring, american, airlines, to]
14638   [you, have, my, money, you, change, my, flight...
14639   [we, have, 8, ppl, so, we, need, 2, know, how,...
Name:   new_text, Length: 14640, dtype: object
```

```
nltk.download('stopwords')
from nltk.corpus import stopwords
stop_words=stopwords.words('english')
# Define a function to remove stopwords from a text
def remove_stopwords(text):
    words = nltk.word_tokenize(text)
    filtered_words = [word for word in words if word.lower() not in
stopwords.words('english')]
    return ' '.join(filtered_words)

# Apply the remove_stopwords function to the 'text_column'
df['new_text'] = df['new_text'].astype(str).apply(remove_stopwords)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
df['new_text']
```

```
                [ 'what ' , 'said ' ]
1                [ 'plus ' , 'youve ' , 'added ' , 'commercials...
2                 [ ' ' , 'didnt ' , 'today ' , 'must ' , 'mean ...
0               3 [ 'its ' , 'really ' , 'aggressive ' , 'to ' ,...
1               4 [ 'and ' , 'its ' , ' ' , 'really ' , 'big ' ,... ...
                14635 [ 'thank ' , 'you ' , 'we ' , 'got ' , 'on ' ,...
                14636 [ 'leaving ' , 'over ' , '20 ' , 'minutes ' , ...
                14637 [ 'please ' , 'bring ' , 'american ' , 'airlin...
                14638 [ 'you ' , 'have ' , 'my ' , 'money ' , 'you '...
                14639 [ 'we ' , 'have ' , ' 8 ' , 'ppl ' , 'so ' , '
   Name: new_text, Length: 14640, dtype: object
```

```
nltk.download('wordnet')
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
def lemmatize_text(text):
    words = nltk.word_tokenize(text)
    lemmatized_words = [lemmatizer.lemmatize(word) for word in words]
    return ' '.join(lemmatized_words)
df['new_text'] = df['new_text'].apply(lemmatize_text)
df['new_text']
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
0                  [ 'what ' , 'said ' ]
1     [ 'plus ' , 'youve ' , 'added ' , 'commercials...
```

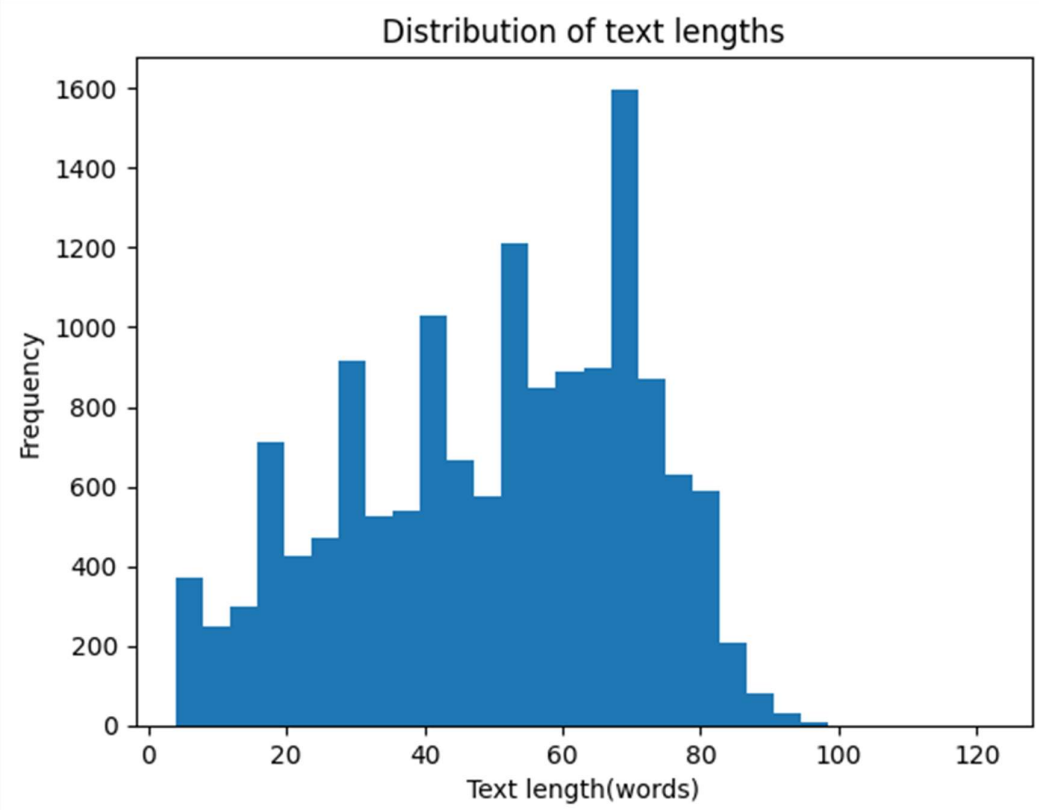
```
2      [ ' ' , 'didnt ' , 'today ' , 'must ' , 'mean ...
3      [ 'its ' , 'really ' , 'aggressive ' , 'to ' ,...
4      [ 'and ' , 'its ' , ' ' , 'really ' , 'big ' ,...
                    ...
14635   [ 'thank ' , 'you ' , 'we ' , 'got ' , 'on ' ,...
14636   [ 'leaving ' , 'over ' , '20 ' , 'minutes ' , ...
14637   [ 'please ' , 'bring ' , 'american ' , 'airlin...
14638   [ 'you ' , 'have ' , 'my ' , 'money ' , 'you '...
14639   [ 'we ' , 'have ' , ' 8 ' , 'ppl ' , 'so ' , '...
Name: new_text, Length: 14640, dtype: object
```

```python
demoji.download_codes()

def remove_emojis(text):

    return demoji.replace(text, '')

df['new_text'] = df['new_text'].apply(remove_emojis)

df['new_text']
```

```
<ipython-input-17-f7f5c0ee2554>:1: FutureWarning: The demoji.download_codes attribute is deprecated and will be removed from demoji in a future version. It is an unused attribute as emoji codes are now distributed directly with the demoji package.
  demoji.download_codes()
0      [ 'what ' , 'said ' ]
1      [ 'plus ' , 'youve ' , 'added ' , 'commercials...
2      [ ' ' , 'didnt ' , 'today ' , 'must ' , 'mean ...
3      [ 'its ' , 'really ' , 'aggressive ' , 'to ' ,...
4      [ 'and ' , 'its ' , ' ' , 'really ' , 'big ' ,...
                    ...
14635   [ 'thank ' , 'you ' , 'we ' , 'got ' , 'on ' ,...
14636   [ 'leaving ' , 'over ' , '20 ' , 'minutes ' , ...
14637   [ 'please ' , 'bring ' , 'american ' , 'airlin...
14638   [ 'you ' , 'have ' , 'my ' , 'money ' , 'you '...
14639   [ 'we ' , 'have ' , ' 8 ' , 'ppl ' , 'so ' , '...
Name: new_text, Length: 14640, dtype: object
```

```
df['text_length_words']=df['new_text'].apply(lambda x: len(x.split()))
plt.hist(df['text_length_words'],bins=30)
plt.xlabel("Text length(words)")
plt.ylabel("Frequency")
plt.title("Distribution of text lengths")
plt.show()
```

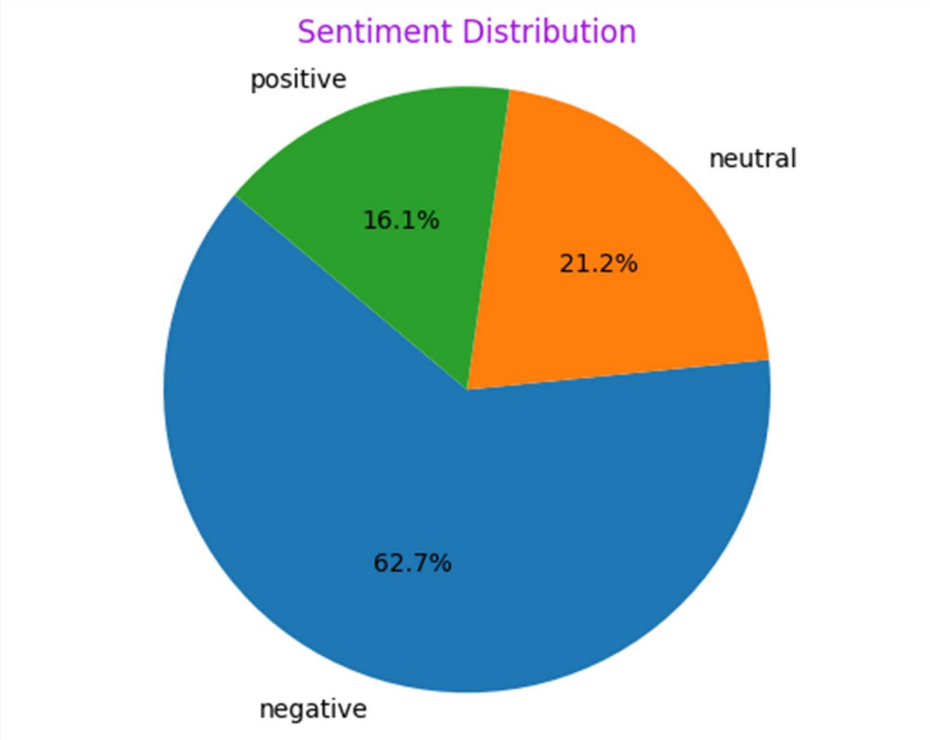
```
threshold=300
```



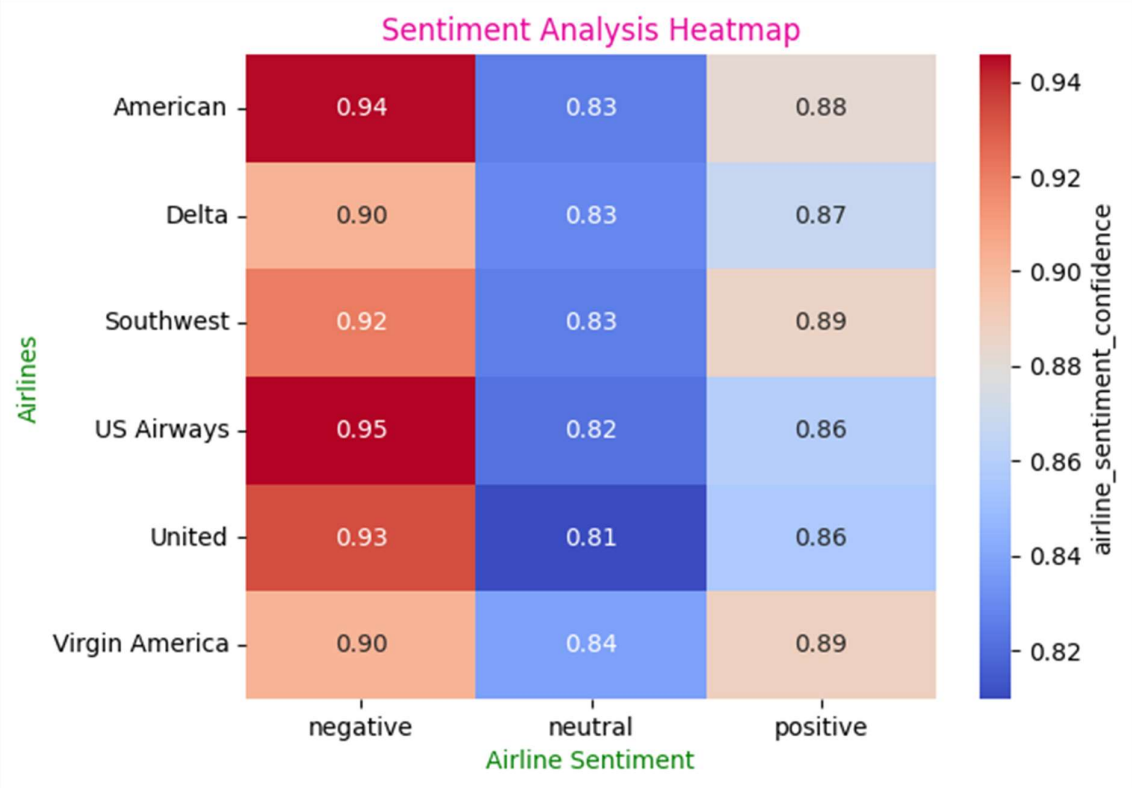

```
df['outlier_flag']=False
df.loc[df['text_length_words'] > threshold,'outlier_flag']=True
df.head()
```

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	name	retweet_count	text	tweet_create
0	570306133677760513	neutral	1.0000	Customer Service Issue	0.6706	Virgin America	cairdin	0	@VirginAmerica What @dhepburn said.	2015-02-2 11:35:52 -080
1	570301130888122368	positive	0.3486	No text	0.0000	Virgin America	jnardino	0	@VirginAmerica plus you've added commercials t...	2015-02-2 11:15:59 -080
2	570301083672813571	neutral	0.6837	No text	0.6706	Virgin America	yvonnalynn	0	@VirginAmerica I didn't today... Must mean I n...	2015-02-2 11:15:48 -080
3	570301031407624196	negative	1.0000	Bad Flight	0.7033	Virgin America	jnardino	0	@VirginAmerica it's really aggressive to blast...	2015-02-2 11:15:36 -080
4	570300817074462722	negative	1.0000	Can't Tell	1.0000	Virgin America	jnardino	0	@VirginAmerica and it's a really big bad thing...	2015-02-2 11:14:45 -080

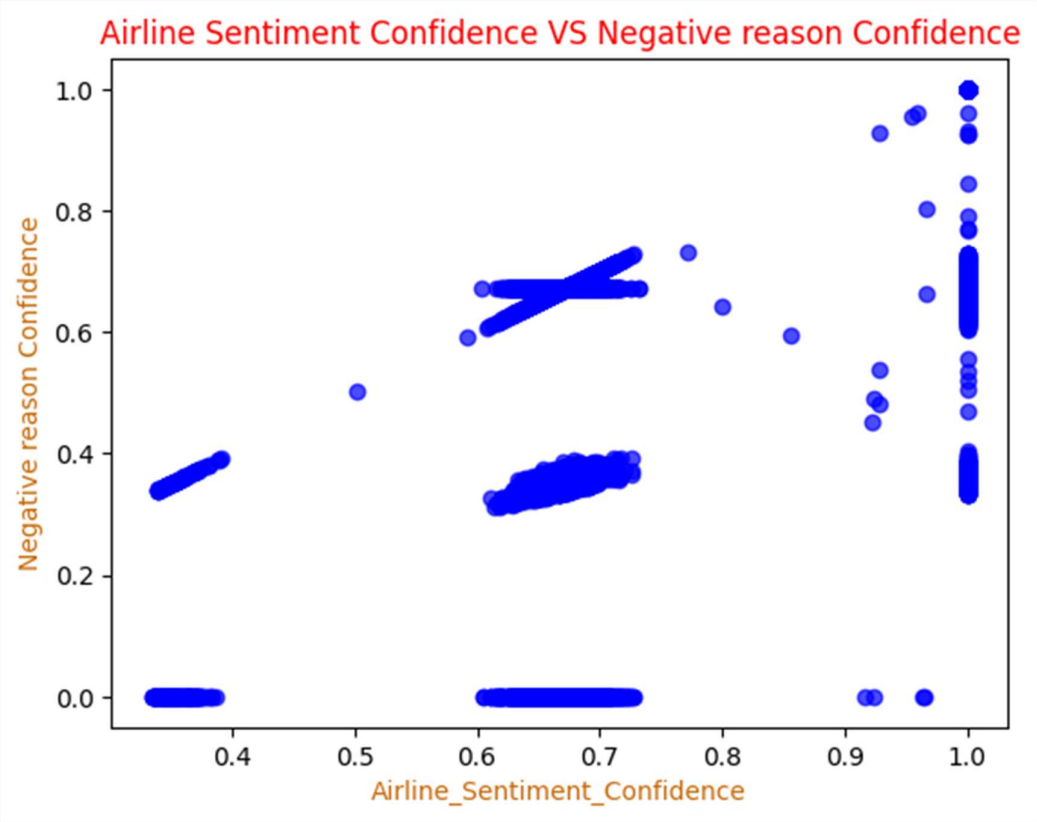
```
sentiment_counts = df['airline_sentiment'].value_counts()
labels = sentiment_counts.index
sizes = sentiment_counts.values
plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=140)
plt.axis('equal')
plt.title("Sentiment Distribution",color='#a114de')
plt.show()
```



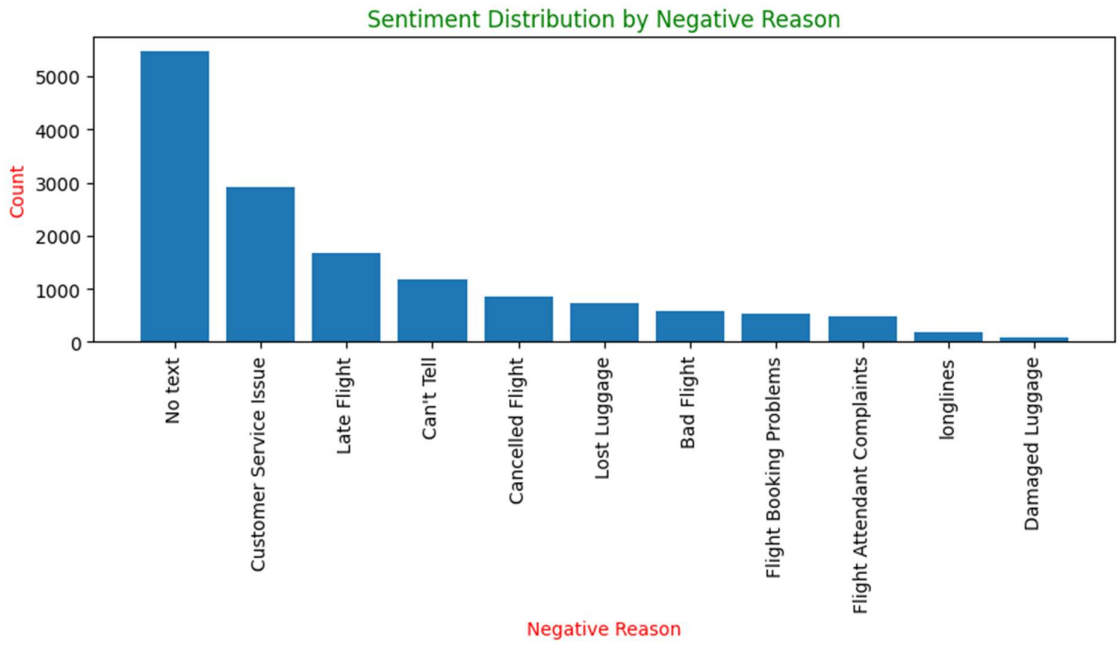
```
heatmap_data = df.pivot_table(index='airline',
columns='airline_sentiment', values='airline_sentiment_confidence',
aggfunc='mean')
sns.heatmap(heatmap_data, cmap="coolwarm", annot=True, fmt=".2f",
cbar_kws={'label': 'airline_sentiment_confidence'})
plt.xlabel('Airline Sentiment',color='green')
plt.ylabel('Airlines',color='green')
plt.title('Sentiment Analysis Heatmap',color='#e6079b')
plt.show()
```



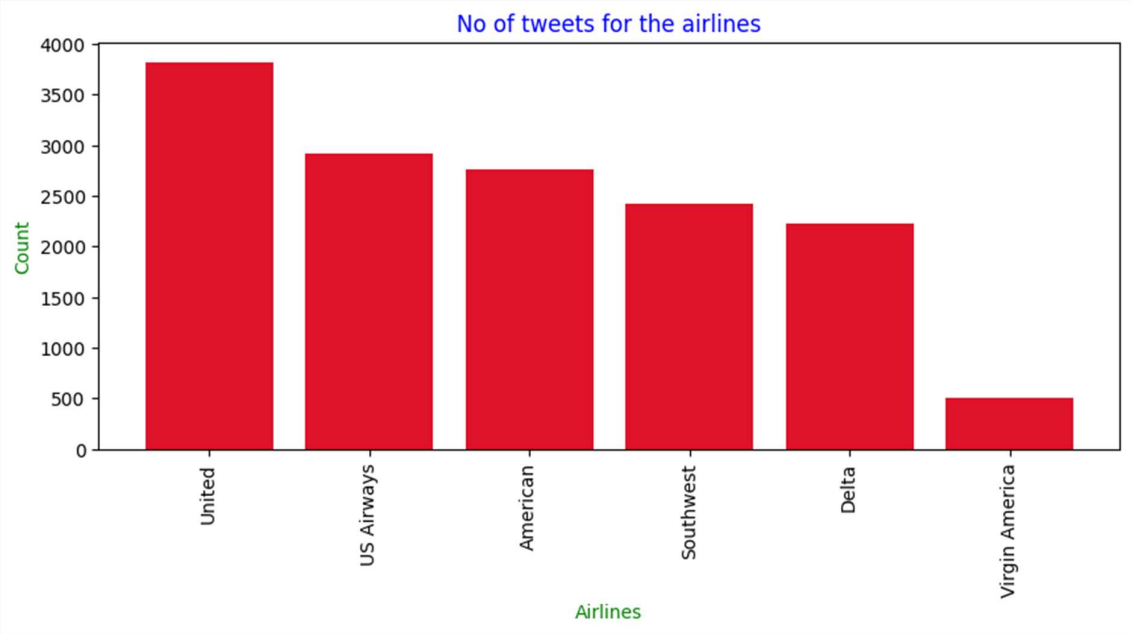
```
x=df['airline_sentiment_confidence']
y=df['negativereason_confidence']
plt.scatter(x, y, marker='o', color='blue', alpha=0.7)
plt.xlabel('Airline_Sentiment_Confidence',color='#c96806')
plt.ylabel('Negative reason Confidence',color='#c96806')
plt.title('Airline Sentiment Confidence VS Negative reason
Confidence',color='red')
plt.show()
```



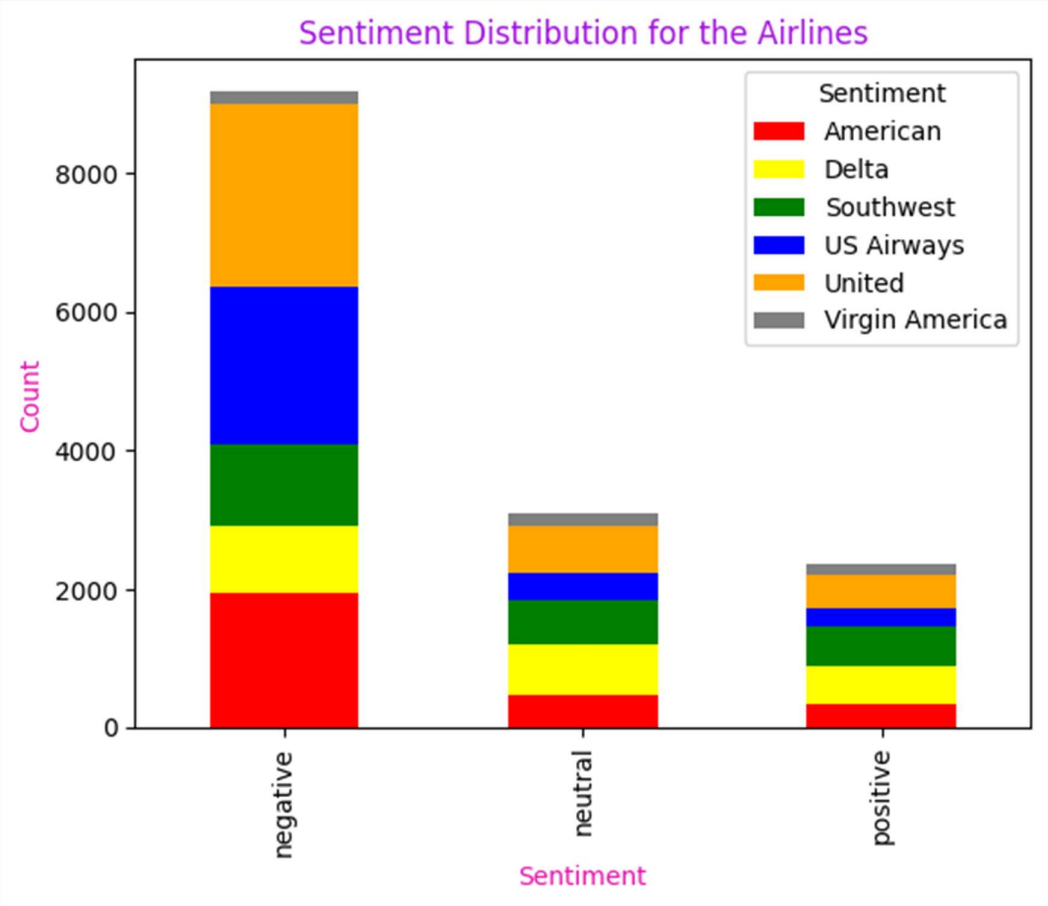
```
negative_reason_counts = df['negativereason'].value_counts()
x = negative_reason_counts.index
y = negative_reason_counts.values
plt.figure(figsize=(10, 3))
plt.bar(x,y)
plt.xlabel('Negative Reason',color='red')
plt.ylabel('Count',color='red')
plt.title('Sentiment Distribution by Negative Reason',color='green')
plt.xticks(rotation=90)
plt.show()
```



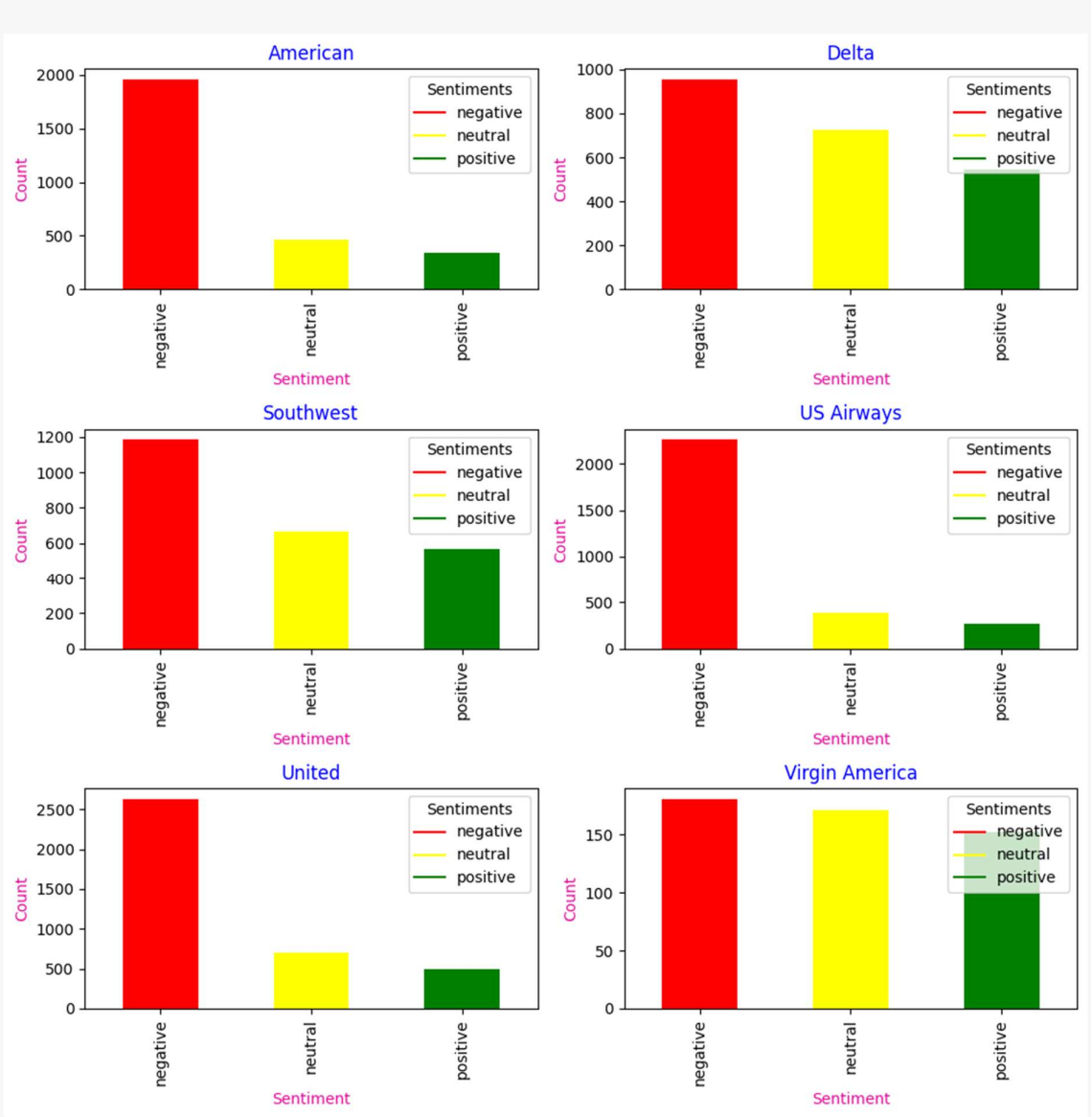
```
airline_counts=df['airline'].value_counts()
x=airline_counts.index
y=airline_counts.values
plt.figure(figsize=(10, 4))
plt.bar(x,y,color='#de122a')
plt.xlabel('Airlines',color='green')
plt.ylabel('Count',color='green')
plt.title('No of tweets for the airlines',color='blue')
plt.xticks(rotation=90)
plt.show()
```



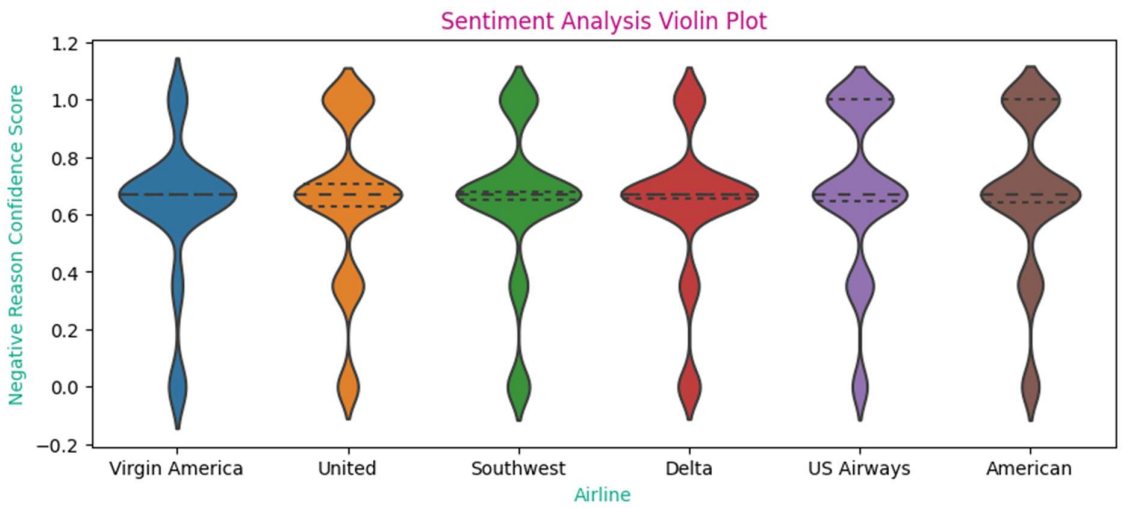
```
sentiment_counts = df.groupby(['airline_sentiment',
'airline']).size().unstack(fill_value=0)
colors = ['red', 'yellow', 'green','blue','orange','grey']
sentiment_counts.plot(kind='bar', stacked=True, color=colors)
plt.xlabel('Sentiment',color='#e310ab')
plt.ylabel('Count',color='#e310ab')
plt.title('Sentiment Distribution for the Airlines',color='#a114de')
plt.legend(title='Sentiment', loc='upper right')
plt.show()
```



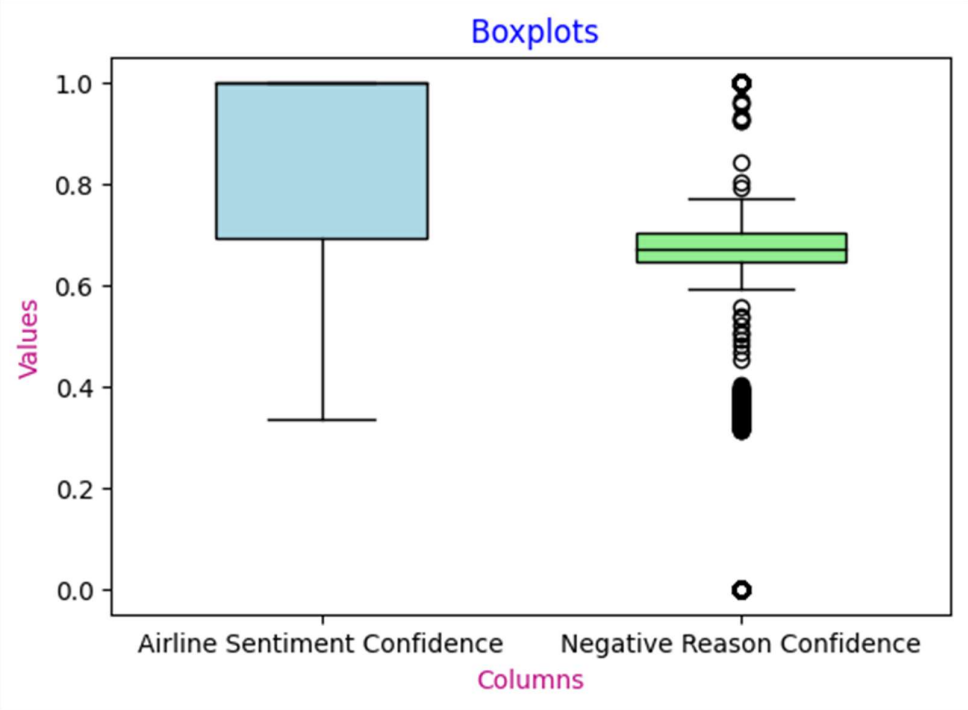
```
sentiment_counts = df.groupby(['airline',
'airline_sentiment']).size().unstack(fill_value=0)
unique_airlines = sentiment_counts.index
fig, axes = plt.subplots(3, 2, figsize=(10, 10))
axes = axes.flatten()
colors = ['red', 'yellow', 'green']
legend_dict = {
    'negative': 'red',
    'neutral': 'yellow',
    'positive': 'green'
}

for i, j in enumerate(unique_airlines):
    sentiment_counts.loc[j].plot(kind='bar', stacked=True, ax=axes[i],
color=[legend_dict[c] for c in sentiment_counts.columns])
    axes[i].set_title(j,color='blue')
    axes[i].set_xlabel('Sentiment',color='#e6079b')
    axes[i].set_ylabel('Count',color='#e6079b')
    legend_handles = [plt.Line2D([0], [0],
color=legend_dict[sentiment], label=sentiment) for sentiment in
sentiment_counts.columns]
    axes[i].legend(handles=legend_handles, title='Sentiments',
loc='upper right')

plt.tight_layout()
plt.show()
```


American
Delta
Southwest
US Airways
United
Virgin America
Sentiments
negative
neutral
positive
Count
Sentiment

```
plt.figure(figsize=(10, 4))
sns.violinplot(x='airline', y='negativereason_confidence', data=df,
inner='quartile')
plt.xlabel('Airline',color='#0ca889')
plt.ylabel('Negative Reason Confidence Score',color='#0ca889')
plt.title('Sentiment Analysis Violin Plot',color='#bd0981')
plt.show()
```



```
data1 = df['airline_sentiment_confidence']
data2 = df['negativereason_confidence']
plt.figure(figsize=(6, 4))
bp1 = plt.boxplot(data1, positions=[1], patch_artist=True, widths=0.5)
bp2 = plt.boxplot(data2, positions=[2], patch_artist=True, widths=0.5)
box_colors = ['lightblue', 'lightgreen']
whisker_color = 'black'

for bplot, color in zip([bp1, bp2], box_colors):
    for element in ['boxes', 'whiskers', 'medians', 'fliers']:
        plt.setp(bplot[element], color=whisker_color)
        if element == 'boxes':
            plt.setp(bplot[element], facecolor=color)
plt.xticks([1, 2], ['Airline Sentiment Confidence', 'Negative Reason
Confidence'])
plt.xlabel('Columns',color='#bd0981')
plt.ylabel('Values',color='#bd0981')
plt.title(' Boxplots',color='blue')
plt.show()
```



```
from collections import Counter
word_counts = Counter(df['negativereason'])
from wordcloud import WordCloud
wordcloud = WordCloud(width=800, height=400,
background_color='white').generate_from_frequencies(word_counts)
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

architecture model: