Twitter Airline Sentiment Analysis

Cindy Herrera

DSC-680

January 26, 2020

Agenda

- Case Study Overview
- Definition of the Problem
- Data Understanding
- Data Preparation/Cleaning
- Modeling & Deployment
- Summary and Conclusion
- References

Case Study Overview

Social Network

Communication

SOCIAL NETWORK PLATFORMS

- Twitter
- SnapChat
- Instagram
- Facebook
- LinkedIn
- YouTube

- Provide powerful means of communication
- Join interesting groups and pages
- Integrated messenger
- Very large community
- Easy to use
- Get updates from major brands
- Integrates with third party services

Social Media Analytics

OPEN SOURCE SOFTWARE

- Python
- Socioboard
- Hootsuite
- Everypost
- Zoho Social

CLOSED SOURCE SOFTWARE

- Crimson Hexagon
- Talkwalker
- Google Analytics

Definition of the Problem

Social Network

Sentiment Analytics

PROBLEM WITH SENTIMENT ANALYTICS

- The problem in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/level
- Whether the expressed opinion is positive, negative, neutral

SENTIMENT ANALYSIS

- Involves classifying comments or opinions
- Classic “sentiment application” is tracking peoples thoughts
- “Opinion Mining” is AKA in marketing terminology "voice of the customer"

DATA UNDERSTANDING

DELTA

UNITED

Twitter Analysis on Airlines

- Data from Twitter on Airlines
- Social data exists in many forms
- Lot of meaningless accounts
- Data coming from streams is quite noisy and polluted
- Takes a lot of effort separate information
- Tweets are frequently used to express a tweeter's emotion on a particular subject
- The challenge is to gather all such relevant data, detect and summarize the overall sentiment on a topic

Southwest

The process of applying rigorous methods to make sense of the social data is called social data analytics.

Data Preprocessing

SOCIAL MEDIA

Contains different types of data:

- User profiles
- Statistics
- Verbatims
- Media

DATASET: TWEETS ON AIRLINES

Data Preprocessing

The first step should be to check the shape of the dataframe and then check the number of null values in each column.

In this way we can get an idea of the redundant columns in the data frame depending on which columns have the highest number of null values.

```
1 print("Shape of the dataframe is",df.shape)
2 print("The number of nulls in each column are \n", df.isna().sum())
```

```
Shape of the dataframe is (14640, 15)
The number of nulls in each column are
 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
```

```
1 print("Percentage null or na values in df")
2 ((df.isnull() | df.isna()).sum() * 100 / df.index.size).round(2)
```

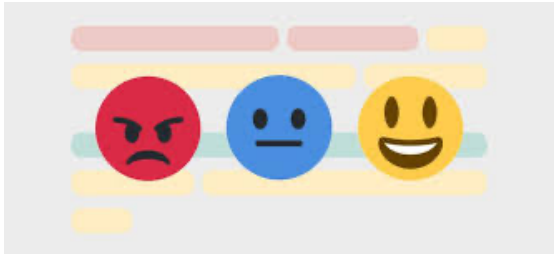
```
Percentage null or na values in df

tweet_id                          0.00
airline_sentiment                 0.00
airline_sentiment_confidence      0.00
negativereason                   37.31
negativereason_confidence        28.13
airline                           0.00
airline_sentiment_gold           99.73
name                              0.00
negativereason_gold              99.78
retweet_count                     0.00
text                              0.00
tweet_coord                      93.04
tweet_created                     0.00
tweet_location                   32.33
user_timezone                    32.92
dtype: float64
```

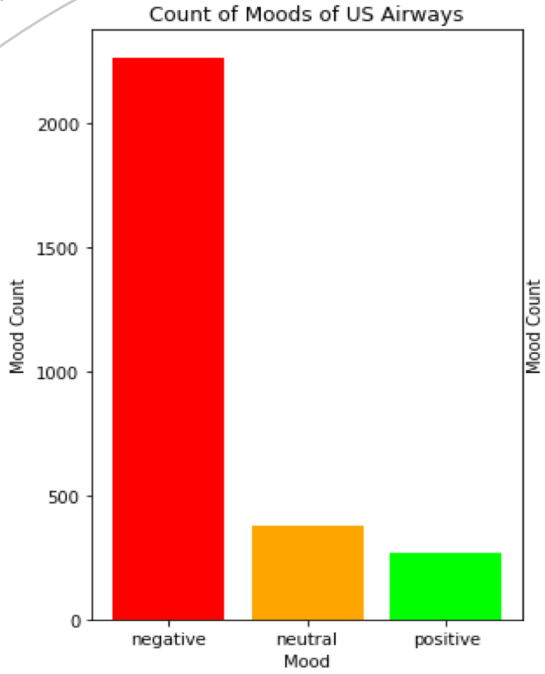
Visualizing Total Tweets

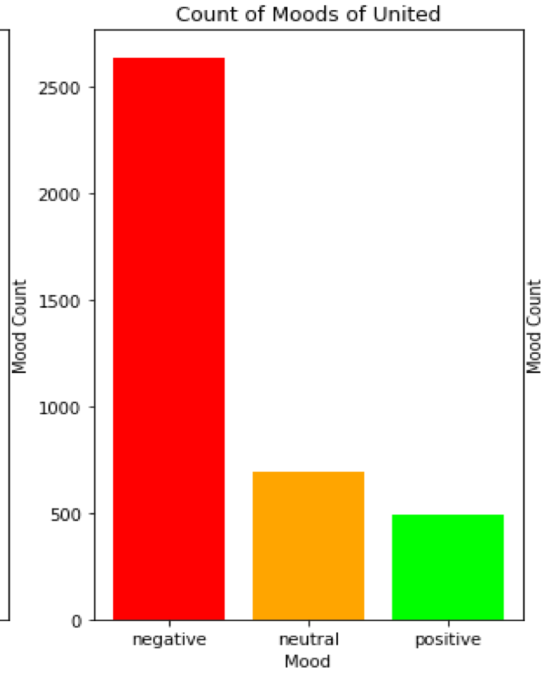
First I started by visualizing the total number of tweets for each airline

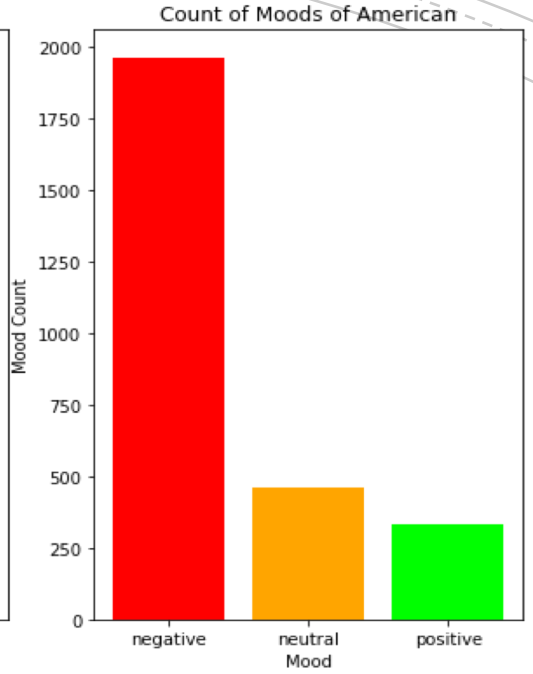
Airline	Total Tweets
United	3,822
US Airways	2,913
American	2,759
Southwest	2,420
Delta	2,222
Virgin America	504

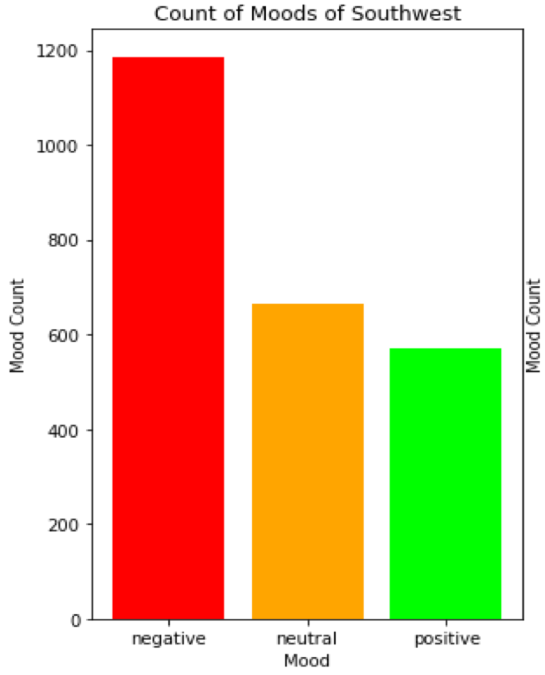


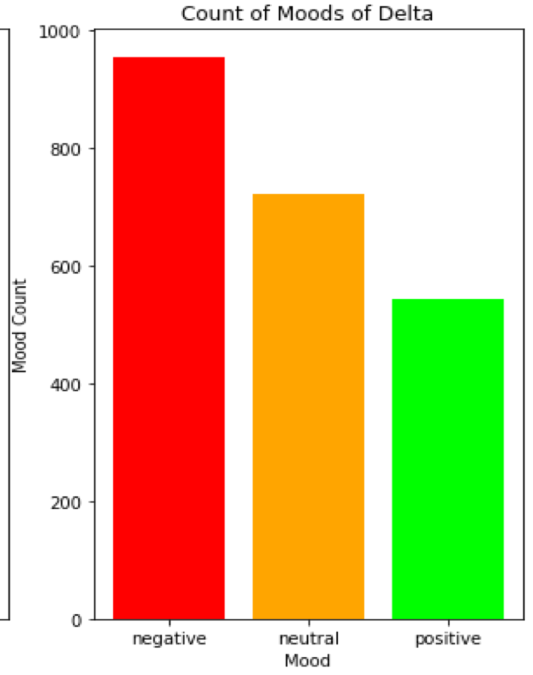
Visualizing Total Tweets by Airline

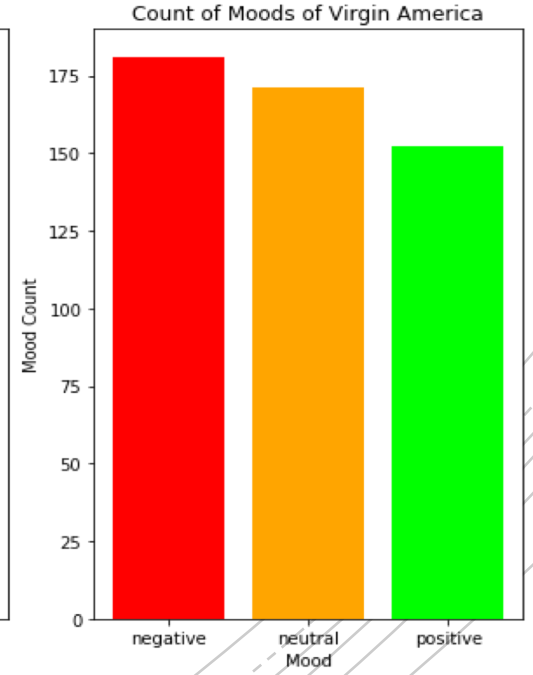


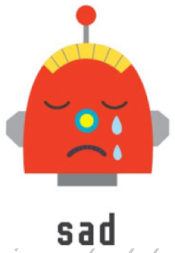




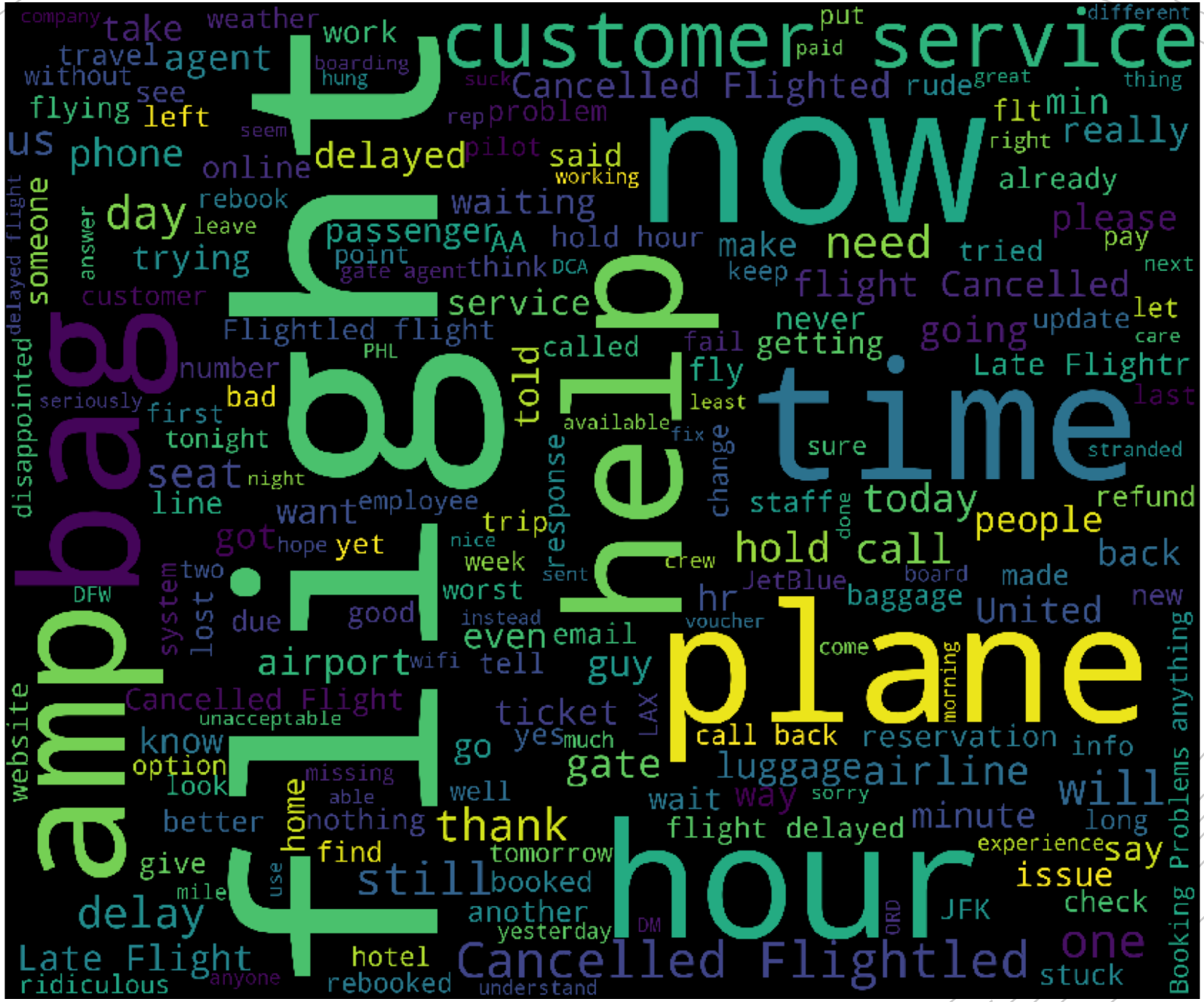


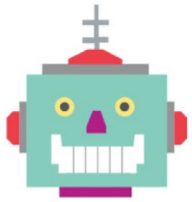






Wordcloud for Negative Sentiment Tweets

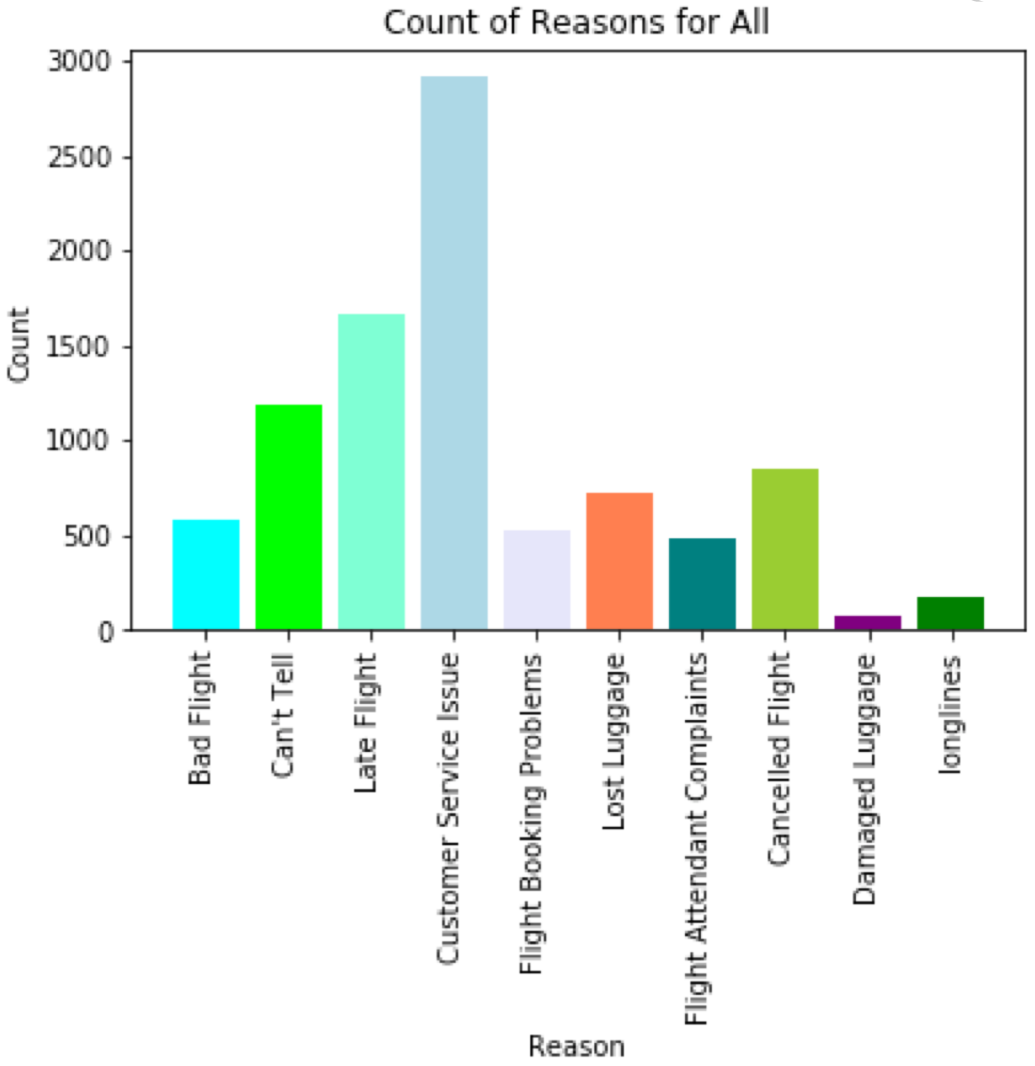




Wordcloud for Positive Sentiment Tweets

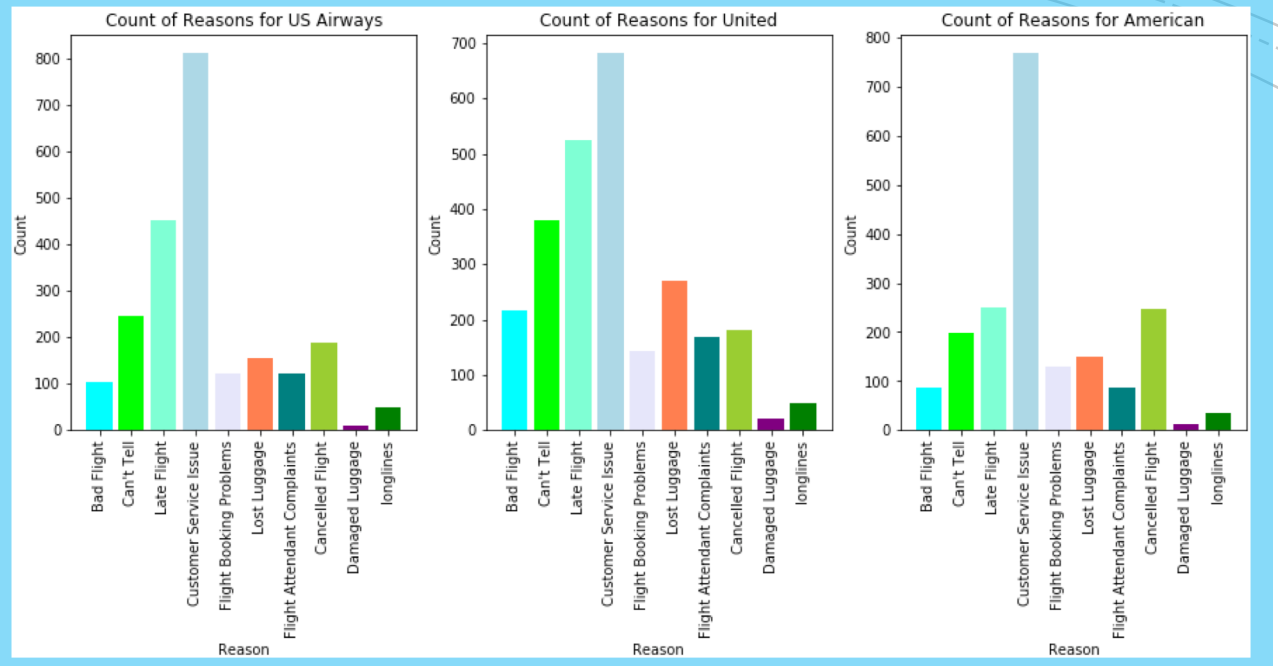
Negative Sentimental Tweets

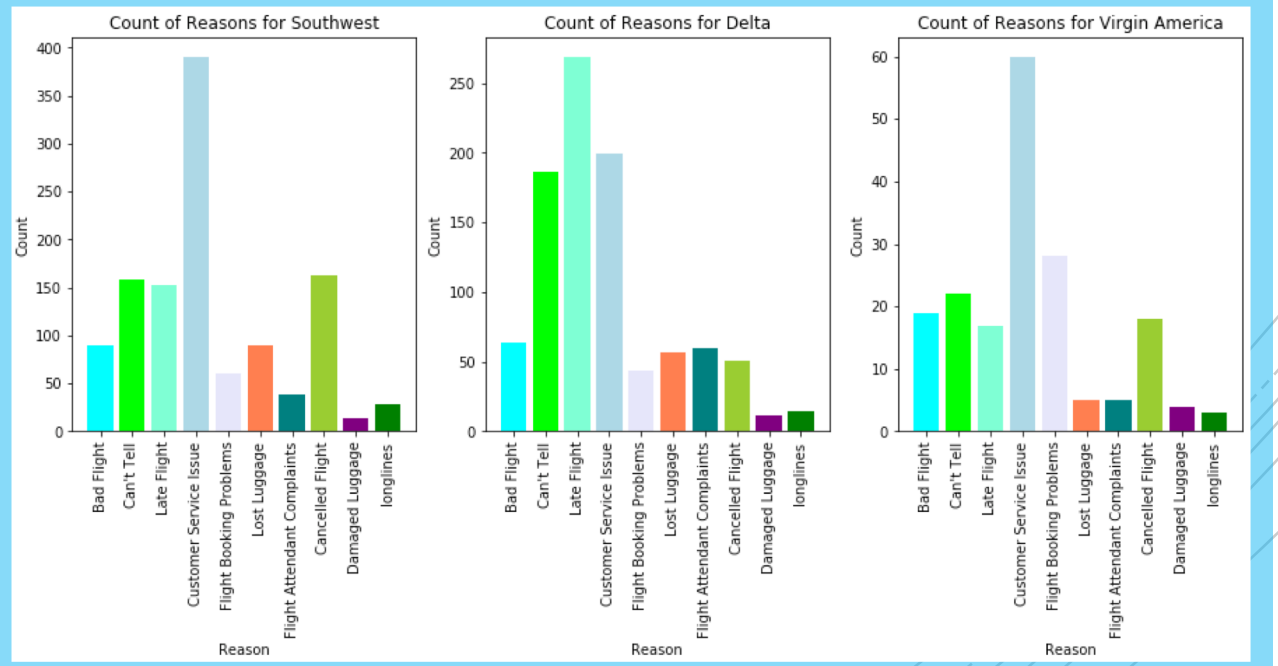
Turbulence on this Delta connection flight was so bad that it turned the drink cart upside down.



Negative Tweets by Airline

- Customer Service Issue is the main negative reason for US Airways, United, American, Southwest, Virgin America
- Late Flight is the main negative reason for Delta
- Airlines like US Airways, United, American have more than 500 negative reasons (Late flight, Customer Service Issue)





Model & Deployment

Preprocessing the tweet text data

Now, we will clean the tweet text data and apply classification algorithms on it

```
def tweet_to_words(tweet):
    letters_only = re.sub("[^a-zA-Z]", " ",tweet)
    words = letters_only.lower().split()
    stops = set(stopwords.words("english"))
    meaningful_words = [w for w in words if not w in stops]
    return( " ".join( meaningful_words ))
```

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

The data is split in the standard 80,20 ratio

```
train,test = train_test_split(df,test_size=0.2,random_state=42)
```

```
train_clean_tweet=[]
for tweet in train['clean_tweet']:
    train_clean_tweet.append(tweet)
test_clean_tweet=[]
for tweet in test['clean_tweet']:
    test_clean_tweet.append(tweet)
```

```
from sklearn.feature_extraction.text import CountVectorizer
v = CountVectorizer(analyzer = "word")
train_features= v.fit_transform(train_clean_tweet)
test_features=v.transform(test_clean_tweet)
```

Decision Tree

Predicating sentiments from tweet text data

- SVM(Support Vector Machine)
- Decision Tree Classifier
- Random Forest Classifier

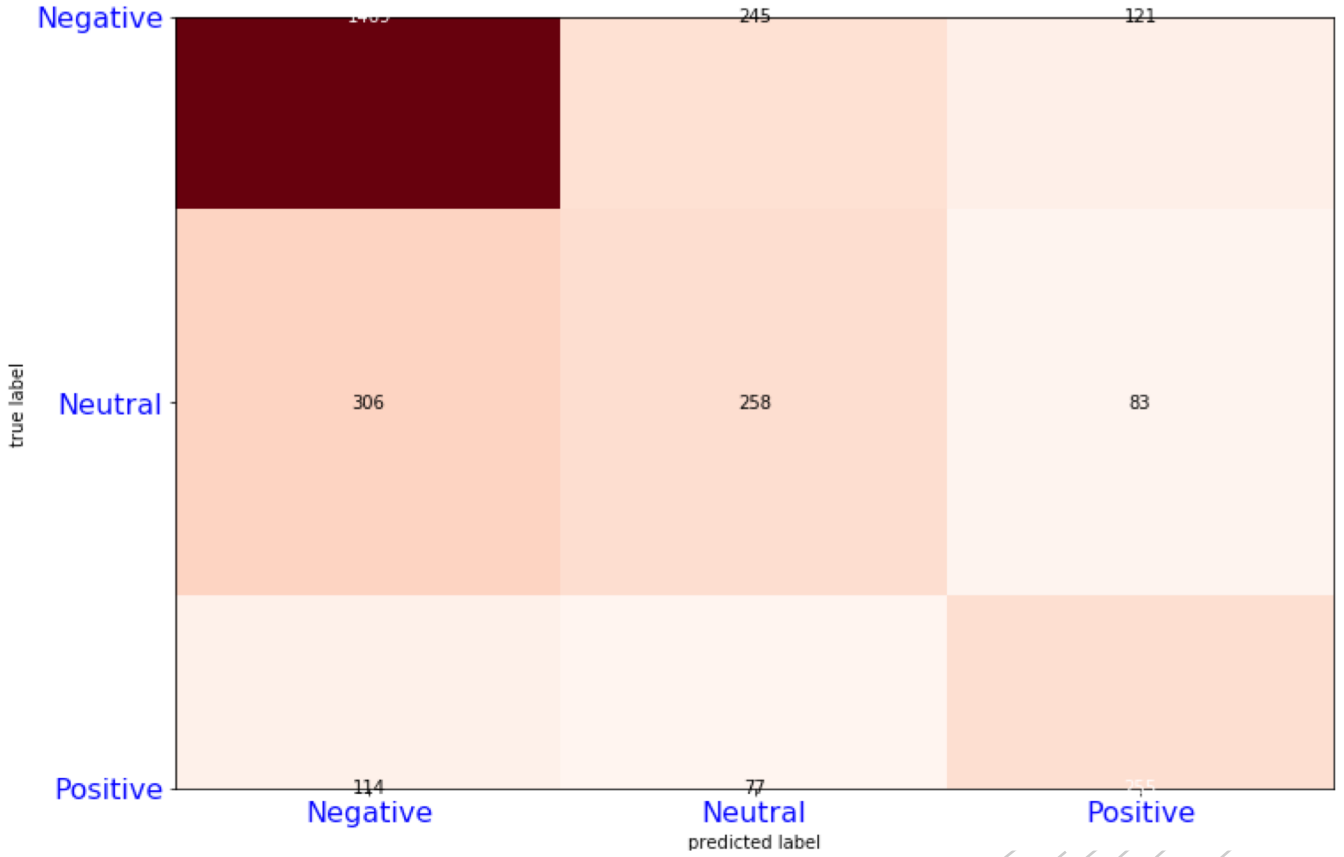
Model & Deployment

```
Accuracy of DecisionTreeClassifieris 0.6769125683060109
              precision    recall  f1-score   support

    negative       0.78      0.80      0.79      1835
     neutral       0.44      0.40      0.42       647
    positive       0.56      0.57      0.56       446

    accuracy                           0.68      2928
   macro avg       0.59      0.59      0.59      2928
weighted avg       0.67      0.68      0.67      2928
```

```
<Figure size 432x288 with 0 Axes>
```



Random Forest Tree

Predicating sentiments from tweet text data

- SVM(Support Vector Machine)
- Decision Tree Classifier
- Random Forest Classifier

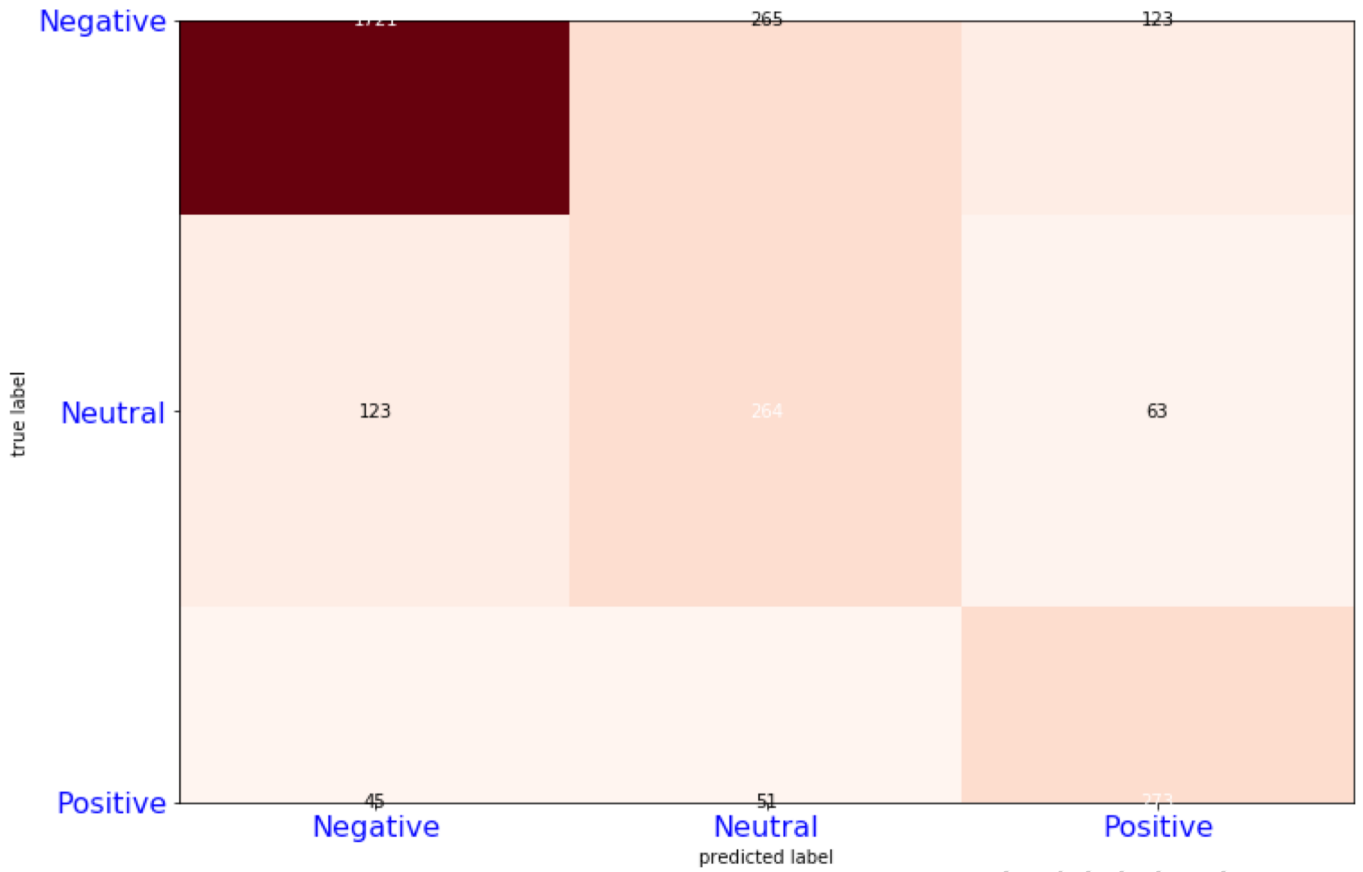
Model & Deployment

```
Accuracy of RandomForestClassifieris 0.7711748633879781
              precision    recall  f1-score   support

    negative       0.91      0.82      0.86      2109
     neutral       0.46      0.59      0.51       450
    positive       0.59      0.74      0.66       369

    accuracy                           0.77      2928
   macro avg       0.65      0.71      0.68      2928
weighted avg       0.80      0.77      0.78      2928
```

```
<Figure size 432x288 with 0 Axes>
```



Summary

- Financial loss can cripple a business due to customer experience (CE)
- Major driving negative sentiment is customer experience
- Business need to value their employees