**Problem Statement: Customer Sentiment Analysis.**

This is a Natural language Processing based problem.

**Data Description:**

Data file named sentiment\_analysis.csv contains the data for this project. This dataset contains three columns named airline\_sentiment ,airline and text which are all object types. airline\_sentiment and airline are categorical columns where text data is a text which needs to be preprocessed before building a predicting model on the data. It contains 14846 rows and 3 columns.

airline\_sentiment is a dependent(target) variable while others are independent variables.

**Data Preprocessing:**

‘text column contains lots of noise in data such as stop words, punctuation marks, special characters, inconsistent cases, numbers,@ twitter handles and inflectional endings. These keywords do not convey useful information and add noise to data. Words such as ‘process’, ‘processed’ or ‘processing’ can be reduced to base(root) form of ‘process’ by removing inflectional endings such as ‘ed’ or ‘ing’.Therefore, it is recommended to remove these keywords before building the predictive model. These keywords have been removed using different functions in python notebook and after preprocessing, resultant data is completely cleaned which is further ready for exploratory data analysis.

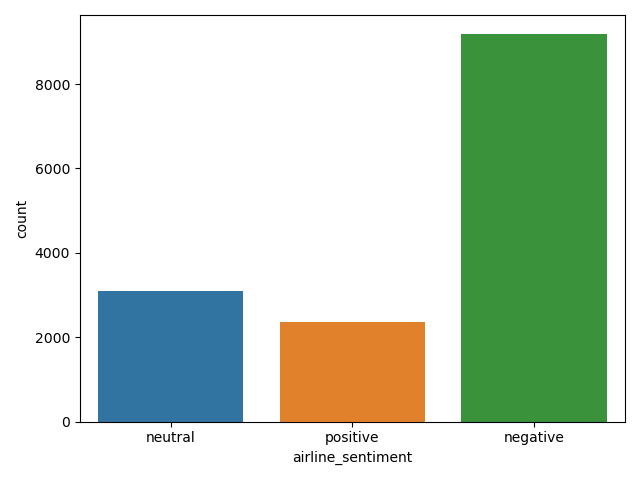
**Exploratory Data Analysis:**

I have performed data analysis on the cleaned data. Firstly, I have checked for missing values in data, there were no missing values in three columns. Then I have performed distribution across various categories of sentiments :positive, negative, neutral. I concluded that distribution of classes is imbalanced. Number of Negative Comments were around 4 times the number of positive comments. Number of Negative comments were around 3 times the number of neural comments.

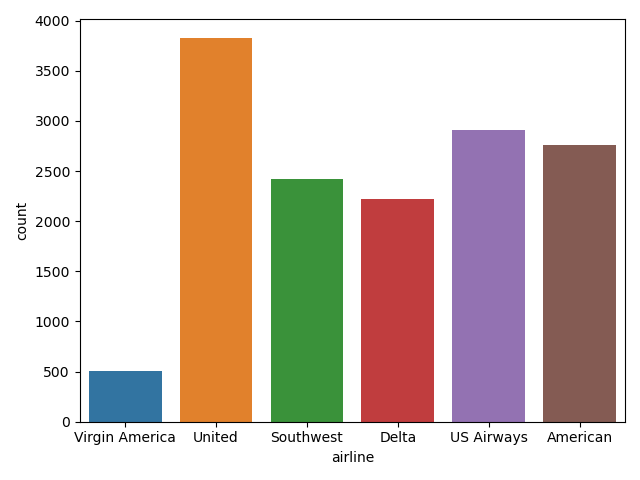
Then I have performed distribution across various categories of airlines: 'Virgin America' ,'United', 'Southwest', 'Delta', 'US Airways' ,'American'. I found that distribution is almost equal for all airlines except Virgin America, which is almost 4 times less than other airlines.

**Visualization:**

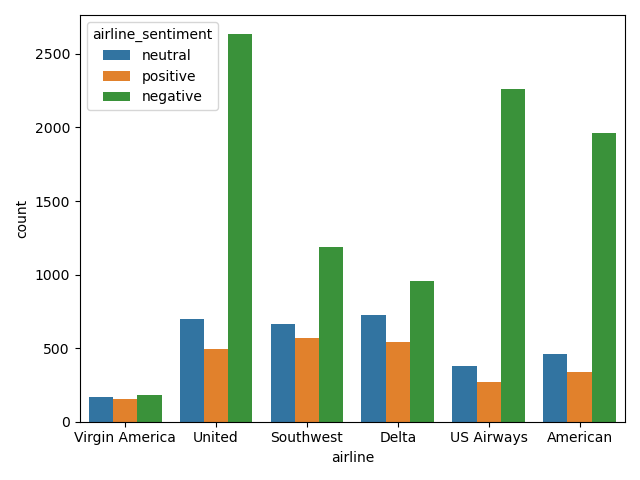
**Counts for different sentiments:**



**Counts for different airlines:**



**Visualizing relationship between airline sentiment and type of airline:**

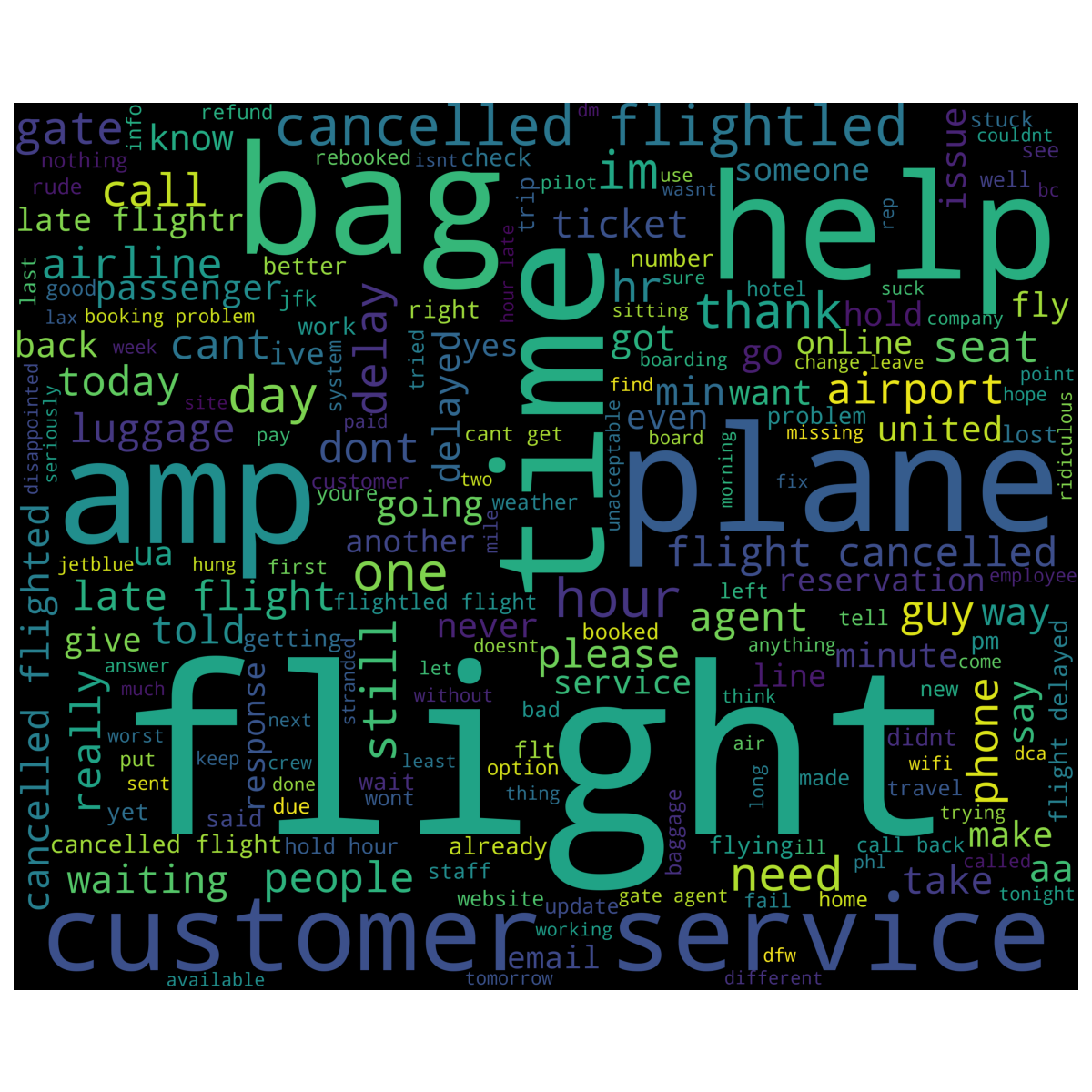


**Word Cloud for positive words:**



Word clouds are generated to visualize the most frequent words. I subsetted the dataset to visualize most frequent words for positive sentiment. Words that are bigger in size have appeared more frequently than words which are smaller in size. Here, most frequent positive words are awesome ,good, thank, flight, good, great, best etc.

**Word Cloud for negative words:**



Here, I subsetted the dataset to visualize most frequent words for negative sentiment. Words that are bigger in size have appeared more frequently than words which are smaller in size. Here, most frequent positive words are flight, bag, help, time, plane, customer service etc.

**Modelling:**

For modelling purpose, I have chosen mainly three algorithms namely: Linear SVM, Logistic regression and Random Forest.

**Linear SVM:**SVM has always performed better for text classification. In addition, text classification outperforms when using SVM with Linear kernel. Therefore, I have used SVM with Linear kernel for this dataset. Using this model has resulted in F1 score of 0.7872

**Logistic Regression:** Logistic regression also performs well on text data. Therefore, I decided to use this model. It has also resulted in F1 score of 0.7814.

**Random Forest:** Random forest model is an ensemble model which is known to improve the predictive performance of the model because it creates the ensemble of multiple trees and aggregate their resulted based on average or majority vote. This performs best for most of the classifiers and it has resulted in F1 score of 0.7616.

**Ridge Classification**: I used Ridge regression because text classification problems tend to be high dimensional and high dimensional problems are likely to be linearly seperable. Therefore, linear classifiers such as Ridge regression perform well on this data. Also, Ridge classification avoids overfitting by regularizing the weights of the model and keeping them small. Also, if the task is linearly seperable, then it outperforms other classifiers. Using this model resulted in f1 score of 0.7790.

**Bernoulli Naïve Bayes:** It performs very well on text classification problems. In addition, it performs extremely well in the presence of large number of features. In text data, each word is treated as a feature and there are thousands of different words. Also, it performs well in the presence of irrelevant features and is relatively unaffected by them.

**Performance Metric**:

I have chosen F1 score as a metric for this dataset because F1 performs better for imbalanced datasets. In this dataset, as class distributions are unequal, therefore F1 metric is well suited metric. F1 Score is an weighted average of precision and recall. Higher F1 score is preferred. Higher F1 score means model has high precision and high recall. Accuracy Score cannot be used in this case as it requires equal class distribution for target variable.

**Comparison:** Among all the classifiers, Linear SVC outperformed other classifiers. This is because Linear SVC performed well on high dimensional data. As this text data after performing vectorization is high dimensional, and high dimensional data is likely to be linear ,therefore Linear SVC performs well on this text data.

**Topic Modelling:**

Topic Modelling is a technique to understand the different topics associated with the customer feedback. These topics can reveal important information about the customer’s intention about the tweet. We can define number of topics to present to us by tuning the model. Topic modelling basically models text keywords into different groups which tells us main topics of discussion in dataset.

Here, I have performed topic modelling using Latent Dirichlet Allocation because LDA results in more logical and consistent topics as compared to other techniques. I have provided number of components as 5 for topics modelling which will result in 5 different topics and identified the top 10 words for each topic.

**Libraries used:**

Python : 3.6.6

Sklearn: 0.23.1

Numpy: 1.18.5

Pandas:1.0.5

Jupyter:1.0.0

WordCloud: 1.7.0

Seaborn:0.10.1

Matplotlib:3.3.0

NLTK:3.5