Sentiment Analysis on U.S. Airline Dataset

Meena Rapaka
Siva Naga Lakshmi Karamsetty
Ying Ke

Table of Contents

Abstract:

Can anyone truly interpret what a human intends? I think no one can precisely. Yet, machines can. Natural language Processing, which we are acquainted with means – computers can understand what we mean. Here, we are applying Sentiment Analysis – which implies that computers can not only understand what we say but can comprehend what we mean. As artificial intelligence evolves in our daily life, it has become vital for the world to advance technologically and be able to communicate with the machines in a language we are accustomed to. Natural language processing helps us fills this complex gap. And sentimental analysis helps us identify, extract, quantify and study emotion from information provided. Human communication is nuanced and complex and it is difficult for us interpret the actual emotion. Sentiment is a combination of tone of voice, word choice, writing style and for computers to understand the way humans communicate, the definitions of the words and what we really mean, they need to understand our sentiments.

The key aspect of sentiment analysis is to analyze a body of text for understanding the opinion expressed by it. Basic task is to quantify this sentiment with a positive or negative value, called polarity. The overall sentiment is often inferred as positive, neutral or negative from the sign of the polarity score. Here we are using the supervised machine learning approaches to compute the sentiment of the sentences.

Introduction:

The project is regarding analysis about the problems of each major U.S airline such as American Airlines, Delta, Southwest, United, US Airways, Virgin America. Twitter data was scraped from the airlines and is categorized into positive, negative and neutral tweets, followed by categorizing negative reasons further such as “delay” or “rude service”. 14640 tweets from 7700 users were analyzed as a part of it. The dataset is processed, and modelling techniques are applied further to get desired results.

Natural language processing techniques such as word clouds, TF-IDF, Bag of words, ngrams, sentiment analysis etc., are used to process the data. Also, machine learning techniques such as logistic regression, random forest, support vector machine, K-Nearest Neighbor, Decision tree, Naïve Bayes are applied to predict the outcome variables. A baseline model, Support vector machine classifier is performed to check the accuracy and use it as a baseline for rest our analysis. Then, we compute the accuracies for various models to recognize the best performing model among the different models we applied. We got the best accuracy for sentiment analysis with Logistic regression with an accuracy of 77% for both TF-IDF and Bag of Words model compared to the baseline accuracy of 64.5%.

Dataset:

This dataset is about U.S Airlines dataset, it contains whether the sentiment of the tweets in this set is positive, neutral or negative for six US airlines such as American, Delta, Southwest, United etc., These tweets are extracted from Twitter API and was scraped from February of 2015. Dataset contains 14640 tweets from 7700 users which were analyzed. Variables in the dataset are tweet_id, airline_sentiment, negativereason_gold, text etc., Variables such as text, airline_sentiment are considered as base parameters.

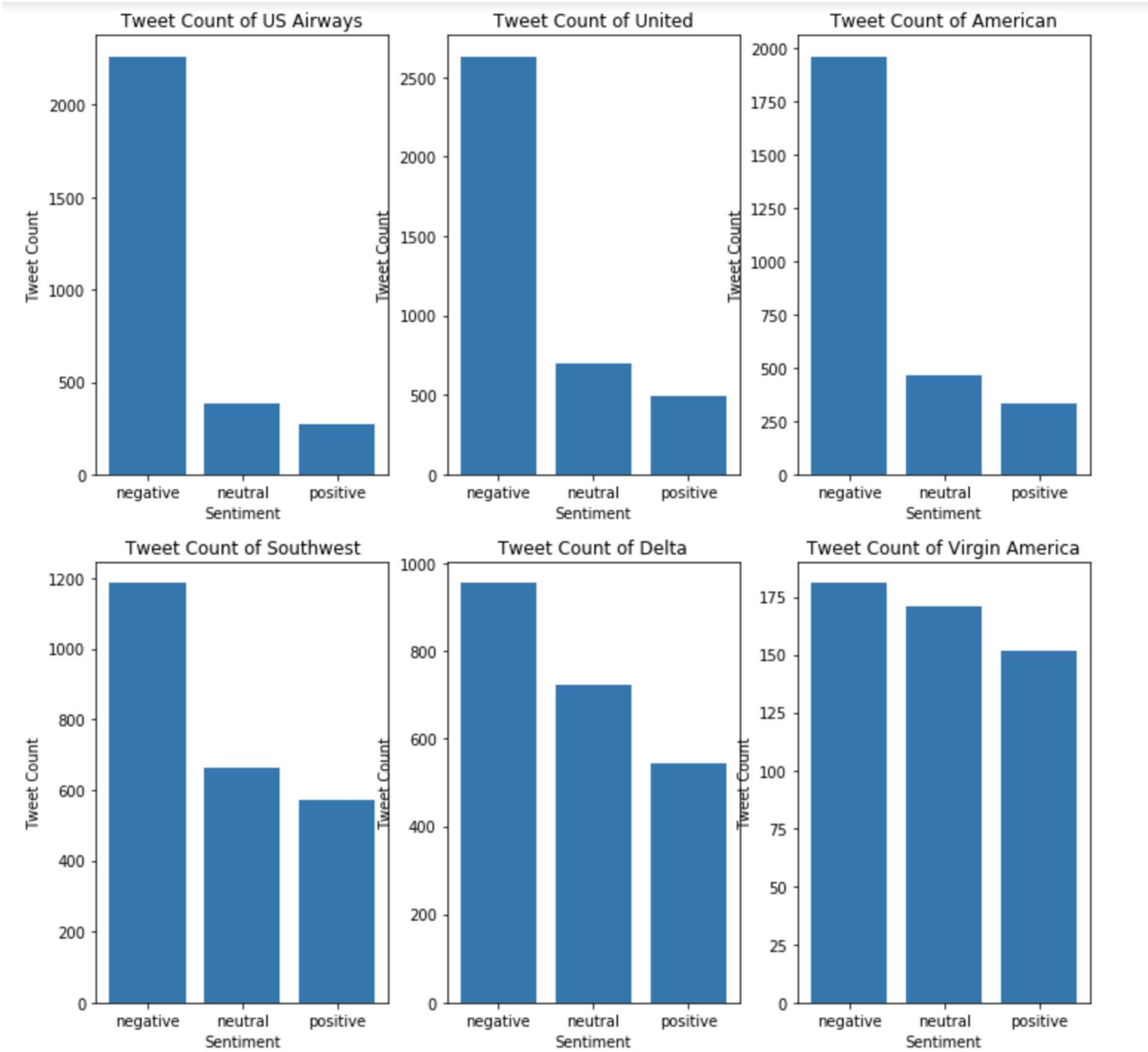
A1 | fx tweet_id

	A	B	C	D	E	F	G	H	I	J	
1	tweet_id	airline_senti	airline_sentime	negativereason	negativereason	airline	airline_senti	name	negativereason_	retweet_co	text
2	5.70E+17	neutral	1			Virgin America		cairdin		0	@VirginAmerica What @dhepburn said.
3	5.70E+17	positive	0.3486		0	Virgin America		jnardino		0	@VirginAmerica plus you've added commercials to the experience... tacky.
4	5.70E+17	neutral	0.6837			Virgin America		yvonnalynn		0	@VirginAmerica I didn't today... Must mean I need to take another trip!
5	5.70E+17	negative	1	Bad Flight	0.7033	Virgin America		jnardino		0	@VirginAmerica it's really aggressive to blast obnoxious "entertainment" in your guests' faces & they have litt
6	5.70E+17	negative	1	Can't Tell	1	Virgin America		jnardino		0	@VirginAmerica and it's a really big bad thing about it
7	5.70E+17	negative	1	Can't Tell	0.6842	Virgin America		jnardino		0	@VirginAmerica seriously would pay $30 a flight for seats that didn't have this playing.
8	5.70E+17	positive	0.6745		0	Virgin America		cjmcginnis		0	@VirginAmerica yes, nearly every time I fly VX this ,Äúear worm,Äù won,Äôt go away :)
9	5.70E+17	neutral	0.634			Virgin America		pilot		0	@VirginAmerica Really missed a prime opportunity for Men Without Hats parody, there. https://t.co/mWpG7grEZP
10	5.70E+17	positive	0.6559			Virgin America		dhepburn		0	@virginamerica Well, I didn't,Ä¶but NOW I DO! :-D
11	5.70E+17	positive	1			Virgin America		YupitsTate		0	@VirginAmerica it was amazing, and arrived an hour early. You're too good to me.
12	5.70E+17	neutral	0.6769		0	Virgin America		idk_but_youtube		0	@VirginAmerica did you know that suicide is the second leading cause of death among teens 10-24
13	5.70E+17	positive	1			Virgin America		HyperCamiLax		0	@VirginAmerica I <3 pretty graphics. so much better than minimal iconography. :D
14	5.70E+17	positive	1			Virgin America		HyperCamiLax		0	@VirginAmerica This is such a great deal! Already thinking about my 2nd trip to @Australia & I haven't even g
15	5.70E+17	positive	0.6451			Virgin America		mollanderson		0	@VirginAmerica @virginmedia I'm flying your #fabulous #Seductive skies again! U take all the #stress away from t
16	5.70E+17	positive	1			Virgin America		sjespers		0	@VirginAmerica Thanks!
17	5.70E+17	negative	0.6842	Late Flight	0.3684	Virgin America		smartwatermelon		0	@VirginAmerica SFO-PDX schedule is still MIA.
18	5.70E+17	positive	1			Virgin America		ItzBrianHunty		0	@VirginAmerica So excited for my first cross country flight LAX to MCO I've heard nothing but great things about Vi
19	5.70E+17	negative	1	Bad Flight	1	Virgin America		heatherovieda		0	@VirginAmerica I flew from NYC to SFO last week and couldn't fully sit in my seat due to two large gentleman on
20	5.70E+17	positive	1			Virgin America		thebrandiray		0	I ,ù§Ô∏è flying @VirginAmerica. ,ò∫Ô∏èüëç
21	5.70E+17	positive	1			Virgin America		JNLpierce		0	@VirginAmerica you know what would be amazingly awesome? BOS-FLL PLEASE!!!!!!! I want to fly with only you.
22	5.70E+17	negative	0.6705	Can't Tell	0.3614	Virgin America		MISSGJ		0	@VirginAmerica why are your first fares in May over three times more than other carriers when all seats are availa
23	5.70E+17	positive	1			Virgin America		DT_Les		0	@VirginAmerica I love this graphic. http://t.co/UT5GrRwAaA
24	5.70E+17	positive	1			Virgin America		ElvinaBeck		0	@VirginAmerica I love the hipster innovation. You are a feel good brand.
25	5.70E+17	neutral	1			Virgin America		rjlynch21086		0	@VirginAmerica will you be making BOS>LAS non stop permanently anytime soon?
26	5.70E+17	negative	1	Customer Service Issue	0.3557	Virgin America		ayeevickiee		0	@VirginAmerica you guys messed up my seating.. I reserved seating with my friends and you guys gave my seat aw
27	5.70E+17	negative	1	Customer Service Issue	1	Virgin America		Leora13		0	@VirginAmerica status match program. I applied and it's been three weeks. Called and emailed with no response
28	5.70E+17	negative	1	Can't Tell	0.6614	Virgin America		meredithjlynn		0	@VirginAmerica What happened 2 ur vegan food options?! At least say on ur site so i know I won't be able 2 eat ar
29	5.70E+17	neutral	0.6854			Virgin America		AdamSinger		0	@VirginAmerica do you miss me? Don't worry we'll be together very soon.
30	5.70E+17	negative	1	Bad Flight	1	Virgin America		blackjackpro911		0	@VirginAmerica amazing to me that we can't get any cold air from the vents. #VX358 #noair #worstflightever #roa
31	5.70E+17	neutral	0.615		0	Virgin America		TenantsUpstairs		0	@VirginAmerica LAX to EWR - Middle seat on a red eye. Such a noob maneuver. #sendambien #andchexmix
32	5.70E+17	negative	1	Flight Booking Problem	1	Virgin America		jordanpichler		0	@VirginAmerica hi! I just bked a cool birthday trip with you, but i can't add my elevate no. cause i entered my midd
33	5.70E+17	neutral	1			Virgin America		JCervantezzz		0	@VirginAmerica Are the hours of operation for the Club at SFO that are posted online current?
34	5.70E+17	negative	1	Customer Service Issue	1	Virgin America		Cuschoolie1		0	@VirginAmerica help, left expensive headphones on flight 89 IAD to LAX today. Seat 2A. No one answering L&
35	5.70E+17	negative	1	Customer Service Issue	1	Virgin America		amandubmccarty		0	@VirginAmerica awaiting my return phone call, just would prefer to use your online self-service option :(

Fig 1. Twitter dataset

Visualization of Tweets:

The six U.S airlines dataset is visualized as shown below for all the airlines based on their polarity – negative, positive or neutral. The plots are plotted for the polarity against the tweet count. Based on the visualizations made, you can observe that US Airways has highest number of negative tweets when compared with other airlines. Southwest airlines have highest number of positive tweets. The scale if different for all the visualizations.



Data Cleansing:

As a part of Data cleaning, we have removed the unnecessary columns such as negativereason_gold, airline_sentiment_gold etc., And we clean the data by removing stop words, special characters, URL and replacing words like haven't to have not, isn't is not etc., We then declare a "msg_list". The tweets are tokenized and then lemmatized, we check for word spell. Now the cleaned message is appended into "msg_list". To address these issues in the data, before applying models we pre-processed the data and used a function called CountVectorizer from scikit learn package of python.

Data Pre-processing:

Our Dataset is split into 80% training data and 20% testing data. We extract features for Cleaned tweets using:

TF-IDF
Bag of words

TF-IDF:

Term frequency – Inverse document frequency. It is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The tf–idf value increases proportionally to the number of times a word appears in the document. It returns the weight of a word.

$$tf(t,d) = \frac{number\ of\ occurrences\ of\ term\ in\ document}{total\ number\ of\ all\ words\ in\ document}$$

The Term Frequency (TF) of a term, t, and a document, d.

Bag of words:

The bag-of-words model is a way of representing text data when modeling text with machine learning algorithms. It is simple to understand and implement and has seen great success in problems such as language modeling and document classification. This model focuses completely on the words, or sometimes a string of words, but usually pays no attention to the "context" so-to-speak. The bag of words model usually has a large list, probably better thought of as a sort of "dictionary," which are words that carry sentiment. These words each have their own "value" when found in text. The values are typically all added up and the result is a sentiment valuation. The equation to add and derive a number can vary, but this model mainly focuses on the words, and makes no attempt to understand language fundamentals.

Baseline Model:

To compute the Baseline model, we used support vector machine as a baseline model with for both TF-IDF and Bag of words technique. Before this, we divided our dataset into training and test data as 80% and 20% respectively. We considered SVM as a baseline as it is not able to classify the dataset, it is returning the same output as the input.

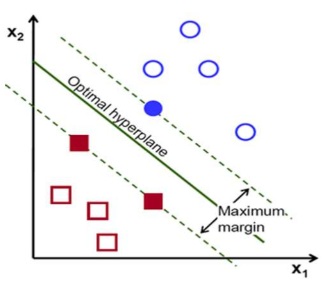
Model Building:

For this project, we build the predictive models on the dataset using the two feature sets – TF-IDF and Bag of Words. We used the below models to check which one is the best performing model. We calculate precision, recall, F1-score and support metrics and their weighted averages for all the models.

- K nearest neighbor (KNN)
- Support Vector Machine (SVM)
- Decision Trees
- Random Forest
- Logistic Regression
- Naïve Bayes

Support Vector Machine:

In machine learning, support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. A discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. In two-dimensional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side.



```
0.6451502732240437
[[1889    0    0]
 [ 580    0    0]
 [ 459    0    0]]
              precision    recall  f1-score   support

           0       0.65      1.00      0.78      1889
           1       0.00      0.00      0.00       580
           2       0.00      0.00      0.00       459

   micro avg       0.65      0.65      0.65      2928
   macro avg       0.22      0.33      0.26      2928
weighted avg       0.42      0.65      0.51      2928
```

SVM for TF-IDF

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
  kernel='rbf', max_iter=-1, probability=False, random_state=19,
  shrinking=True, tol=0.001, verbose=False) 0.6454918032786885
[[1888    0    1]
 [ 580    0    0]
 [ 457    0    2]]
              precision    recall  f1-score   support

           0       0.65      1.00      0.78      1889
           1       0.00      0.00      0.00       580
           2       0.67      0.00      0.01       459

   micro avg       0.65      0.65      0.65      2928
   macro avg       0.44      0.33      0.26      2928
weighted avg       0.52      0.65      0.51      2928
```

SVM for Bag of Words

Accuracy for support vector machine for TF-IDF is 64.5% and support vector machine for 64.5%.

K-Nearest Neighbor:

In pattern recognition, the k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression. Here, the data points are separated into several classes to predict the classification of a new data point.

```
0.41188524590163933
[[688 961 240]
 [152 353  75]
 [100 194 165]]
              precision    recall  f1-score   support

           0       0.73      0.36      0.49      1889
           1       0.23      0.61      0.34       580
           2       0.34      0.36      0.35       459

   micro avg       0.41      0.41      0.41      2928
   macro avg       0.44      0.44      0.39      2928
weighted avg       0.57      0.41      0.44      2928
```

KNN for TF-IDF

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=None, n_neighbors=5, p=2,
           weights='uniform') 0.5942622950819673
[[1165  561  163]
 [ 157  298  125]
 [  91   91  277]]
              precision    recall  f1-score   support

           0       0.82      0.62      0.71      1889
           1       0.31      0.51      0.39       580
           2       0.49      0.60      0.54       459

   micro avg       0.59      0.59      0.59      2928
   macro avg       0.54      0.58      0.55      2928
weighted avg       0.67      0.59      0.62      2928
```

KNN for Bag of Words

Accuracy for KNN using TF-IDF is 41.1% and Bag of Words is 59.4%.

Decision Trees:

A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements. Decision tree is a graph that uses a branching method to illustrate every possible outcome of a decision.

```
0.7004781420765027
[[1522  231  136]
 [ 226  286   68]
 [ 138   78  243]]
              precision    recall  f1-score   support

           0       0.81      0.81      0.81      1889
           1       0.48      0.49      0.49       580
           2       0.54      0.53      0.54       459

   micro avg       0.70      0.70      0.70      2928
   macro avg       0.61      0.61      0.61      2928
weighted avg       0.70      0.70      0.70      2928
```

Decision Tree for TF-IDF

```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=8,
            splitter='random') 0.6789617486338798
[[1444  302  143]
 [ 215  293   72]
 [ 122   86  251]]
              precision    recall  f1-score   support

           0       0.81      0.76      0.79      1889
           1       0.43      0.51      0.46       580
           2       0.54      0.55      0.54       459

   micro avg       0.68      0.68      0.68      2928
   macro avg       0.59      0.61      0.60      2928
weighted avg       0.69      0.68      0.68      2928
```

Decision Tree for Bag of Words

Decision trees has an accuracy of 70% for TF-IDF and 67.8%.

Random Forest:

Random forest is a supervised learning algorithm and are an ensemble learning method for classification and regression. It operates by constructing a multitude of decision trees and outputs the mean prediction of individual trees.

```
0.7418032786885246
[[1686  143   60]
 [ 265  254   61]
 [ 161   66  232]]
              precision    recall  f1-score   support

           0       0.80      0.89      0.84      1889
           1       0.55      0.44      0.49       580
           2       0.66      0.51      0.57       459

   micro avg       0.74      0.74      0.74      2928
   macro avg       0.67      0.61      0.63      2928
weighted avg       0.73      0.74      0.73      2928
```

Random Forest for TF-IDF

```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=None,
            oob_score=False, random_state=19, verbose=0, warm_start=False) 0.7120901639344263
[[1554  224  111]
 [ 223  272   85]
 [ 121   79  259]]
              precision    recall  f1-score   support

           0       0.82      0.82      0.82      1889
           1       0.47      0.47      0.47       580
           2       0.57      0.56      0.57       459

   micro avg       0.71      0.71      0.71      2928
   macro avg       0.62      0.62      0.62      2928
weighted avg       0.71      0.71      0.71      2928
```

Random Forest for Bag of Words

Random Forest has an accuracy of 74.1% of TF-IDF and 71.2%.

Logistic Regression:

Logistic regression is a common predictive analysis which is used to describe data and analyze the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables. It is a statistical method that analyzes the data, in which there are one or more independent variables determines the outcome where the outcome has only two possible values. For example, classifying a boy or girl, binary digits 0 or 1 etc.

Top ten and last ten words were predicted using both TF-IDF and Bag of Words model.

```
0.7704918032786885
[[1777   74   38]
 [ 328  219   33]
 [ 153   46  260]]
              precision    recall  f1-score   support

           0       0.79      0.94      0.86      1889
           1       0.65      0.38      0.48       580
           2       0.79      0.57      0.66       459

   micro avg       0.77      0.77      0.77      2928
   macro avg       0.74      0.63      0.66      2928
weighted avg       0.76      0.77      0.75      2928
```

Logistic Regression for TF-IDF

```
top ten          coeff        word
1598   8.057828        hours
211    7.775456          bad
724    6.451484      delayed
1080   6.440725          fix
2588   5.787779         rude
2561   5.610567   ridiculous
1584   5.404636         hour
2902   5.089241       system
1900   4.932235      luggage
721    4.842146        delay
```

Top ten words using TF-IDF

```
last ten          coeff          word
1884 -3.505671         love
966  -3.538154    excellent
418  -3.925201  cannot wait
90   -3.932570      amazing
1747 -4.066641        kudos
197  -4.286923      awesome
1447 -4.628584        great
3339 -4.742162      worries
2959 -5.085186       thanks
2951 -7.521570        thank
```

Last ten words using TF-IDF

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l1', random_state=19, solver='warn',
          tol=0.0001, verbose=0, warm_start=False) 0.7756147540983607
[[1684  152   53]
 [ 226  311   43]
 [ 107   76  276]]
              precision    recall  f1-score   support

           0       0.83      0.89      0.86      1889
           1       0.58      0.54      0.56       580
           2       0.74      0.60      0.66       459

   micro avg       0.78      0.78      0.78      2928
   macro avg       0.72      0.68      0.69      2928
weighted avg       0.77      0.78      0.77      2928
```

Logistic Regression for Bag of Words

```
top ten         coeff          word
759   3.370306       screwed
328   3.175137           fix
730   3.173638    ridiculous
350   3.058929        forced
934   2.958470       useless
832   2.907747      suitcase
968   2.689175        werent
738   2.656015          rude
423   2.523745       holding
916   2.420277  unacceptable
```

Top ten words using Bag of Words

```
last ten         coeff       word
32   -2.098237    amazing
220  -2.158966      deals
952  -2.210318       warm
631  -2.245442   passbook
72   -2.256186    awesome
978  -2.275365  wonderful
482  -2.358649      kudos
861  -2.443951      thank
287  -2.671162  excellent
987  -3.355398    worries
```

Last ten using Bag of Words

Logistic Regression has an accuracy of 77% for TF-IDF and 77.5% for Bag of Words. Top ten and last ten words are identified for both the features.

Naïve Bayes:

In machine learning, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

```
0.7506830601092896
[[1814   56   19]
 [ 379  171   30]
 [ 212   34  213]]
              precision    recall  f1-score   support

           0       0.75      0.96      0.84      1889
           1       0.66      0.29      0.41       580
           2       0.81      0.46      0.59       459

   micro avg       0.75      0.75      0.75      2928
   macro avg       0.74      0.57      0.61      2928
weighted avg       0.74      0.75      0.72      2928
```

Naïve Bayes for TF-IDF

```
MultinomialNB(alpha=0.5, class_prior=None, fit_prior=True) 0.7359972677595629
[[1662  154   73]
 [ 303  223   54]
 [ 144   45  270]]
              precision    recall  f1-score   support

           0       0.79      0.88      0.83      1889
           1       0.53      0.38      0.45       580
           2       0.68      0.59      0.63       459

   micro avg       0.74      0.74      0.74      2928
   macro avg       0.67      0.62      0.64      2928
weighted avg       0.72      0.74      0.72      2928
```

Naïve Bayes for Bag of Words

Naïve Bayes has an 75% accuracy for TF-IDF and 73.5% for Bag of Words.

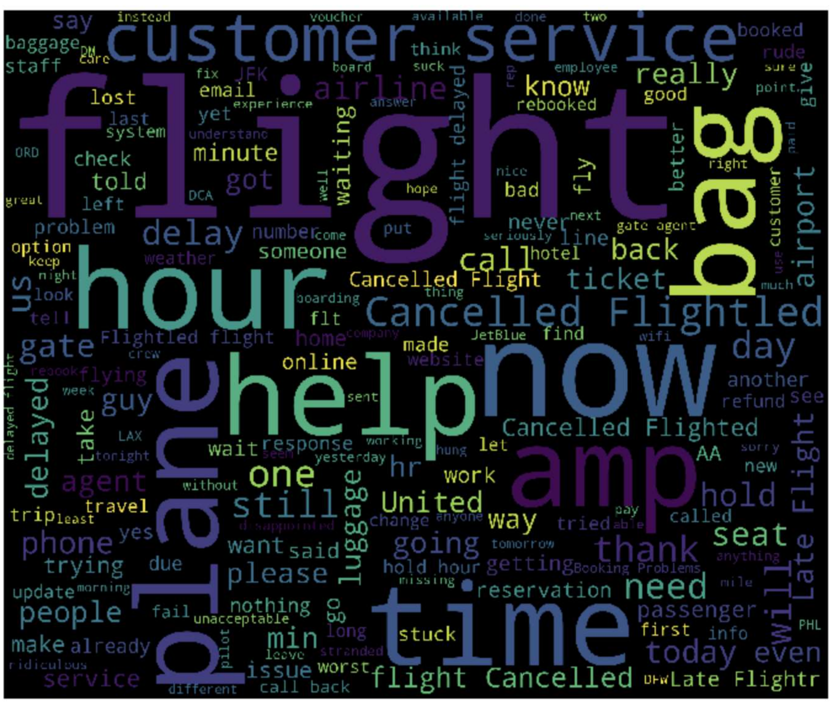
Model Comparison:

All the above models are compared using accuracy as the common metric. Below is the table showing the accuracies that are achieved using various models:

Model	Accuracy (TF-IDF)	Accuracy (Bag of Words)
Support Vector Machine	64.5%	64.5%
K Nearest Neighbor	41.1%	59.4%
Decision Trees	70%	67.8%
Random Forest	74.1%	71.2%
Logistic Regression	77%	77.5%
Naïve Bayes	75%	73.5%

Data Analysis:

World cloud for Negative tweets.



Words like help, delayed, cancelled, waiting, customer service, response define negative words.

Error analysis:

In this project, our dataset contains 14640 tweets from 7700 users which were analyzed. By using KNN algorithms, we cannot clearly identify which type of distance to use with the best results, and the training data is not that large to predict higher accuracy source as possible. The SVM and random forest can address the overfitting problem. However, there is no large amount of data in our dataset which includes the six different airline and classify them into three main categories – positive, negative and neutral. Based on the analysis, we can see the prediction of the accuracy score which have the same value in Bag of Words and TF-IDF for Support Vector Machines. Moreover, we have implemented the logistic algorithm which is helpful and flexible in our model. We can see there is more than 10000 records of data points per predictor which can provide more accuracy scores compared to the other models. And the main reason would be the binary data. For example, in bag of words model, the sentences will be tokenized and lemmatized into the new list, all the data will be label 0 by value 0 or even label 1 by value 1. By using the logistic regression, we can get the probability estimates which will be smoother and performance well than any other algorithms.

Conclusion:

The initial baseline accuracy was at 64.5%, able to improve the accuracy to a considerably high percent is achieved by applying logistic regression method and found that Logistic regression was a better performing model with an accuracy of 77% for TF-IDF and Bag of Words features.

References:

1. Zygmunt Z. June 8th, 2015. Classifying text with bag-of-words: a tutorial. Retrieved from web article: http://fastml.com/classifying-text-with-bag-of-words-a-tutorial/

2. Pramod Chandrayan. Aug 26, 2015. Machine Learning part 3: Logistic Regression. Retrieved from web article: https://towardsdatascience.com/machine-learning-part-3-logistics-regression-9d890928680f

3. Cambridge University Press. 2008. Retrieved from web article: https://towardsdatascience.com/machine-learning-part-3-logistics-regression-9d890928680f

4. Sklearn.feature_extraction.text.HashingVectorizer¶. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.HashingVectorizer.html