Sentiment Analysis of Marketing

1. Abstract

Sentiment Analysis also known as Opinion Mining refers to the use of natural language processing, text analysis to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine.

In this project, we aim to perform Sentiment Analysis of product based reviews. Data used in this project are online product reviews collected from “amazon.com”. We expect to do review-level categorization of review data with promising outcomes.

1. Introduction

Sentiment is an attitude, thought, or judgment prompted by feeling. Sentiment analysis, which is also known as opinion mining, studies people's sentiments towards certain entities. From a user's perspective, people are able to post their own content through various social media, such as forums, micro-blogs, or online social networking sites. From a researcher's perspective, many social media sites release their application programming interfaces (APIs), prompting data collection and analysis by researchers and developers. However, those types of online data have several flaws that potentially hinder the process of sentiment analysis. The first flaw is that since people can freely post their own content, the quality of their opinions cannot be guaranteed. he second flaw is that ground truth of such online data is not always available. A ground truth is more like a tag of a certain opinion, indicating whether the opinion is positive, negative, or neutral.

"It is a quite boring movie…….. but the scenes were good enough. "

The given line is a movie review that states that "it" (the movie) is quite boring but the scenes were good. Understanding such sentiments require multiple tasks.

Hence, SENTIMENTAL ANALYSIS is a kind of text classification based on *Sentimental Orientation* (SO) of opinion they contain.

Sentiment analysis of product reviews has recently become very popular in text mining and computational linguistics research.

- Firstly, evaluative terms expressing opinions must be extracted from the review.
- Secondly, the SO, or the polarity, of the opinions must be determined.
- Thirdly, the opinion strength, or the intensity, of an opinion should also be determined.
- Finally, the review is classified with respect to sentiment classes, such as Positive and Negative, based on the SO of the opinions it contains.

2. Review of Literature

The most fundamental problem in sentiment analysis is the sentiment polarity categorization, by considering a dataset containing over 5.1 million product reviews from Amazon.com with the products belonging to four categories.

A max-entropy POS tagger is used in order to classify the words of the sentence, an additional python program to speed up the process. The negation words like no, not, and more are included in the adverbs whereas Negation of Adjective and Negation of Verb are specially used to identify the phrases.

The following are the various classification models which are selected for categorization: Naïve Bayesian, Random Forest, Logistic Regression and Support Vector Machine.

For feature selection, Pang and Lee suggested to remove objective sentences by extracting subjective ones. They proposed a text-categorization technique that is able to identify subjective content using minimum cut. Gann et al. selected 6,799 tokens based on Twitter data, where each token is assigned a sentiment score, namely TSI (Total Sentiment Index), featuring itself as a positive token or a negative token. Specifically, a TSI for a certain token is computed as:

$$TSI = \frac{p - \frac{tp}{tn} \times n}{p + \frac{tp}{tn} * n}$$

where p is the number of times a token appears in positive tweets and *n* is the number of times a token appears in negative tweets is $\frac{tp}{tn}$ the ratio of total number of positive tweets over total number of negative tweets.

3. Objective of the Project

- Scrapping product reviews on various websites featuring various products specifically amazon.com.
- Analyze and categorize review data.

- Analyze sentiment on dataset from document level (review level).

- Categorization or classification of opinion sentiment into-
  - Positive
  - Negative

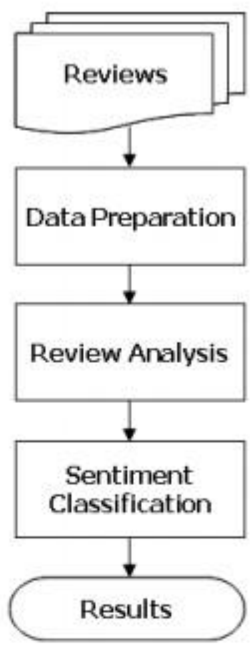


Figure 1: A typical sentiment analysis model.

4. System Design

Hardware Requirements:

- Core i5/i7 processor
- At least 8 GB RAM
- At least 60 GB of Usable Hard Disk Space

Software Requirements:

- Python 3.x
- Anaconda Distribution
- NLTK Toolkit
- UNIX/LINUX Operating System.

Data Information:

- The Amazon reviews dataset consists of reviews from amazon. The data span a period of 18 years, including ~35 million reviews up to March 2013. Reviews include product and user information, ratings, and a plaintext review. For more information, please refer to the following paper: J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. RecSys, 2013.

- The Amazon reviews full score dataset is constructed by Xiang Zhang (xiang.zhang@nyu.edu) from the above dataset. It is used as a text classification benchmark in the following paper: Xiang Zhang, Junbo Zhao, Yann LeCun. Character-level Convolutional Networks for Text Classification. Advances in Neural Information Processing Systems 28 (NIPS 2015).

- The Amazon reviews full score dataset is constructed by randomly taking 200,000 samples for each review score from 1 to 5. In total there are 1,000,000 samples.

Star Level	General Meaning
☆	I hate it.
☆☆	I don't like it.
☆☆☆	It's okay.
☆☆☆☆	I like it.
☆☆☆☆☆	I love it.

Books	reviews (22,507,155 reviews)	metadata (2,370,585 products)	image features
Electronics	reviews (7,824,482 reviews)	metadata (498,196 products)	image features
Movies and TV	reviews (4,607,047 reviews)	metadata (208,321 products)	image features
CDs and Vinyl	reviews (3,749,004 reviews)	metadata (492,799 products)	image features
Clothing, Shoes and Jewelry	reviews (5,748,920 reviews)	metadata (1,503,384 products)	image features
Home and Kitchen	reviews (4,253,926 reviews)	metadata (436,988 products)	image features
Kindle Store	reviews (3,205,467 reviews)	metadata (434,702 products)	image features
Sports and Outdoors	reviews (3,268,695 reviews)	metadata (532,197 products)	image features
Cell Phones and Accessories	reviews (3,447,249 reviews)	metadata (346,793 products)	image features
Health and Personal Care	reviews (2,982,326 reviews)	metadata (263,032 products)	image features
Toys and Games	reviews (2,252,771 reviews)	metadata (336,072 products)	image features
Video Games	reviews (1,324,753 reviews)	metadata (50,953 products)	image features
Tools and Home Improvement	reviews (1,926,047 reviews)	metadata (269,120 products)	image features
Beauty	reviews (2,023,070 reviews)	metadata (259,204 products)	image features
Apps for Android	reviews (2,638,173 reviews)	metadata (61,551 products)	image features
Office Products	reviews (1,243,186 reviews)	metadata (134,838 products)	image features
Pet Supplies	reviews (1,235,316 reviews)	metadata (110,707 products)	image features
Automotive	reviews (1,373,768 reviews)	metadata (331,090 products)	image features
Grocery and Gourmet Food	reviews (1,297,156 reviews)	metadata (171,760 products)	image features
Patio, Lawn and Garden	reviews (993,490 reviews)	metadata (109,094 products)	image features
Baby	reviews (915,446 reviews)	metadata (71,317 products)	image features
Digital Music	reviews (836,006 reviews)	metadata (279,899 products)	image features
Musical Instruments	reviews (500,176 reviews)	metadata (84,901 products)	image features
Amazon Instant Video	reviews (583,933 reviews)	metadata (30,648 products)	image features

Data Format:

The dataset we will use is .json file. The sample of the dataset is given below.

{
"reviewSummary": "Surprisingly delightful",
"reviewText": " This is a first read filled with unexpected humor and profound insights into the art of politics and policy. In brief, it is sly, wry, and wise. ",
"reviewRating": "4",

}

5. Methodology for Implementation (Formulation/Algorithm)

DATA COLLECTION:

Data which means product reviews collected from amazon.com from May 1996 to July 2014. Each review includes the following information: 1) reviewer ID; 2) product ID; 3) rating; 4) time of the review; 5) helpfulness; 6) review text. Every rating is based on a 5-star scale, resulting all the ratings to be ranged from 1-star to 5-star with no existence of a half-star or a quarter-star.

SENTIMENT SENTENCE EXTRACTION & POS TAGGING:

Tokenization of reviews after removal of STOP words which mean nothing related to sentiment is the basic requirement for POS tagging. After proper removal of STOP words like “am, is, are, the, but” and so on the remaining sentences are converted in tokens. These tokens take part in POS tagging

In natural language processing, part-of-speech (POS) taggers have been developed to classify words based on their parts of speech. For sentiment analysis, a POS tagger is very useful because of the following two reasons: 1) Words like nouns and pronouns usually do not contain any sentiment. It is able to filter out such words with the help of a POS tagger; 2) A POS tagger can also be used to distinguish words that can be used in different parts of speech.

NEGETIVE PHRASE IDENTIFICATION:

Words such as adjectives and verbs are able to convey opposite sentiment with the help of negative prefixes. For instance, consider the following sentence that was found in an electronic device’s review: “The built in speaker also has its uses but so far nothing revolutionary." The word, “revolutionary" is a positive word according to the list in. However, the phrase “nothing revolutionary" gives more or less negative feelings. Therefore, it is crucial to identify such phrases. In this work, there are two types of phrases have been identified, namely negation-of-adjective (NOA) and negation-of-verb (NOV).

SENTIMENT CLASSIFICATION ALGORITHMS:

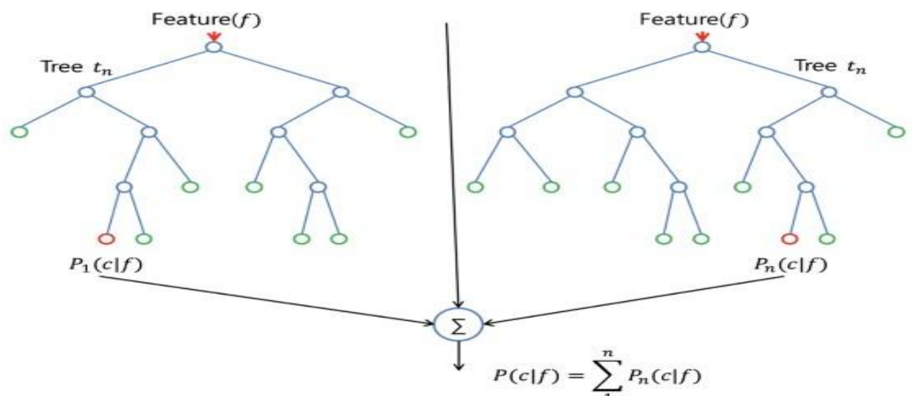
Naïve Bayesian classifier:

The Naïve Bayesian classifier works as follows: Suppose that there exist a set of training data, *D*, in which each tuple is represented by an *n*-dimensional feature vector, $X=x_1,x_2,..,x_n$, indicating *n* measurements made on the tuple from *n* attributes or features. Assume that there are *m* classes, $C_1,C_2,...,C_m$. Given a tuple *X*, the classifier will predict that *X* belongs to C_i if and only if: $P(C_i|X)>P(C_j|X)$, where $i,j\in[1,m]$*a n d* $i\neq j$. $P(C_i|X)$ is computed as:

$$P(C_i|X) = \prod_{k=1}^{n} P(x_k|C_i)$$

Random forest

The random forest classifier was chosen due to its superior performance over a single decision tree with respect to accuracy. It is essentially an ensemble method based on bagging. The classifier works as follows: Given *D*, the classifier firstly creates *k* bootstrap samples of *D*, with each of the samples denoting as D_i. A D_i has the same number of tuples as *D* that are sampled with replacement from *D*. By sampling with replacement, it means that some of the original tuples of *D* may not be included in D_i, whereas others may occur more than once. The classifier then constructs a decision tree based on each D_i. As a result,



a “forest" that consists of *k* decision trees is formed.

To classify an unknown tuple, *X*, each tree returns its class prediction counting as one vote.

The final decision of X's class is assigned to the one that has the most votes.

The decision tree algorithm implemented in scikit-learn is CART (Classification and Regression Trees). CART uses Gini index for its tree induction. For D, the Gini index is computed as:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

Where p_i is the probability that a tuple in D belongs to class C_i. The Gini index measures the impurity of D. The lower the index value is, the better D was partitioned.

Support vector machine

Support vector machine (SVM) is a method for the classification of both linear and nonlinear data. If the data is linearly separable, the SVM searches for the linear optimal separating hyperplane (the linear kernel), which is a decision boundary that separates data of one class from another. Mathematically, a separating hyper plane can be written as: $W \cdot X + b = 0$, where W is a weight vector and $W = w1, w2, ..., w\ n$. X is a training tuple. b is a scalar. In order to optimize the hyperplane, the problem essentially transforms to the minimization of $\|W\|$, which is eventually computed as:

$$\sum_{i=1}^{n} \alpha_i y_i x_i,$$

where α_i are numeric parameters, and y_i are labels based on support vectors, X_i .

That is: if $y_i = 1$ then

$$\sum_{i=1}^{n} w_i x_i \geq 1;$$

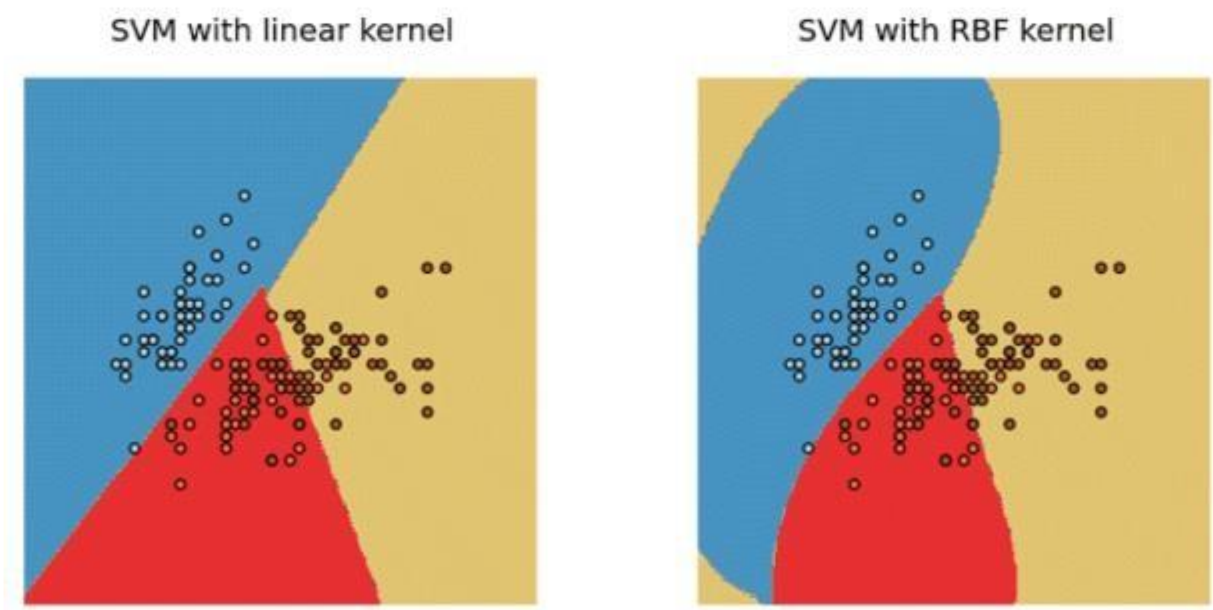
if $y_i = -1$ then

$$\sum_{i=1}^{n} w_i x_i \geq -1.$$

If the data is linearly inseparable, the SVM uses nonlinear mapping to transform the data into a higher dimension. It then solve the problem by finding a linear hyperplane. Functions to perform such transformations are called kernel functions. The kernel function selected for our experiment is the Gaussian Radial Basis Function (RBF):

$$K(X_i, X_j) = e^{-\gamma \|X_i - X_j\|^2/2}$$

where X_i are support vectors, X_j are testing tuples, and γ is a free parameter that uses the default value from scikit-learn in our experiment. Figure shows a classification example of SVM based on the linear kernel and the RBF kernel on the next page-



Logistic Regression

Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). The prediction is based on the use of one or several predictors (numerical and categorical). A linear regression is not appropriate for predicting the value of a binary variable for two reasons:

- A linear regression will predict values outside the acceptable range (e.g. predicting probabilities outside the range 0 to 1)
- Since the dichotomous experiments can only have one of two possible values for each experiment, the residuals will not be normally distributed about the predicted line.

On the other hand, a logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression is similar to a linear regression, but the curve is constructed using the natural logarithm of the “odds” of the target variable, rather than the probability. Moreover, the predictors do not have to be normally distributed or have equal variance in each group.

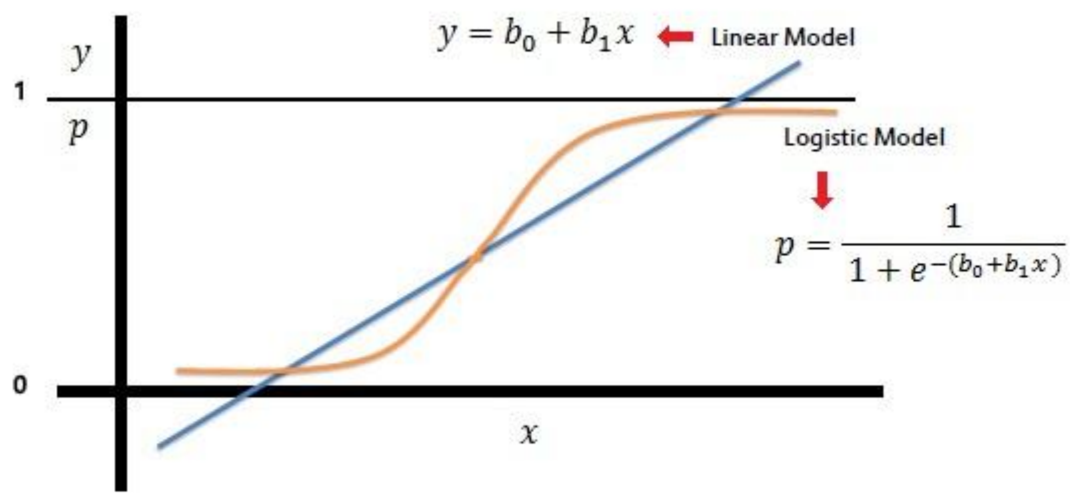
Logistic regression uses maximum likelihood estimation (MLE) to obtain the model coefficients that relate predictors to the target. After this initial function is estimated, the process is repeated until LL (Log Likelihood) does not change significantly.

$$\boldsymbol{\beta}^1 = \boldsymbol{\beta}^0 + [\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}]^{-1}.\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{\mu})$$

$\boldsymbol{\beta}$ is a vector of the logistic regression coefficients.

$\boldsymbol{W}$ is a square matrix of order N with elements $n_i\pi_i(1-\pi_i)$ on the diagonal and zeros everywhere else.

$\boldsymbol{\mu}$ is a vector of length N with elements $\mu_i = n_i\pi_i$.



5. Implementation Details

The training of dataset consists of the following steps:

- **Unpacking of data:** The huge dataset of reviews obtained from amazon.com comes in a .json file format. A small python code has been implemented in order to read the dataset from those files and dump them in to a pickle file for easier and fastaccess and object serialization.

```
with open(data_file, 'r') as file_handler:
    for review in file_handler.readlines():
        df[i] = ast.literal_eval(review)
        i += 1

reviews_df = pd.DataFrame.from_dict(df, orient = 'index')
reviews_df.to_pickle('reviews_digital_music.pickle')
```

Hence initial fetching of data is done in this section using Python File Handlers.

- **Preparing Data for Sentiment Analysis:**

i) The pickle file is hence loaded in this step and the data besides the one used for sentiment analysis is removed. As shown in our sample dataset in Page 11, there are a lot of columns in the data out of which only rating and text review is what we require. So, the column, "reviewSummary" is dropped from the data file.

ii) After that, the review ratings which are 3 out of 5 are removed as they signify neutral review, and all we are concerned of is positive and negative reviews.

iii) The entire task of preprocessing the review data is handled by this

```
reviews_df.drop(columns = ['reviewSummary'], inplace = True)
reviews_df['reviewRating'] = reviews_df.reviewRating.astype('int')
```

```
reviews_df = reviews_df[reviews_df.reviewRating != 3] # Ignoring 3-star reviews -> neutral
reviews_df = reviews_df.assign(sentiment = np.where(reviews_df['reviewRating'] >= 4, 1, 0)) # 1 -> Positive, 0 -> Negati
```

utility class- "NltkPreprocessor".

```python
class NltkPreprocessor:

    def __init__(self, stopwords = None, punct = None, lower = True, strip = True):
        self.lower = lower
        self.strip = strip
        self.stopwords = stopwords or set(sw.words('english'))
        self.punct = punct or set(string.punctuation)
        self.lemmatizer = WordNetLemmatizer()

    def tokenize(self, document):
        tokenized_doc = []

        for sent in sent_tokenize(document):
            for token, tag in pos_tag(wordpunct_tokenize(sent)):
                token = token.lower() if self.lower else token
                token = token.strip() if self.strip else token
                token = token.strip('_0123456789') if self.strip else token
                # token = re.sub(r'\d+', '', token)

                if token in self.stopwords:
                    continue

                if all(char in self.punct for char in token):
                    continue

                lemma = self.lemmatize(token, tag)
                tokenized_doc.append(lemma)

        return tokenized_doc

    def lemmatize(self, token, tag):
        tag = {
            'N': wn.NOUN,
            'V': wn.VERB,
            'R': wn.ADV,
            'J': wn.ADJ
        }.get(tag[0], wn.NOUN)

        return self.lemmatizer.lemmatize(token, tag)
```

iv) The time required to prepare the following data is hence displayed.

```
administrator@administrator-OptiPlex-3040:~/Desktop/sentiment_analysis$
Preprocessing data...
Preprocessing data completed!
Preprocessing time:  0.163 s
```

The time taken to preprocess the data is calculated and displayed

- **Preprocessing Data:** This is a vital part of training the dataset. Here Words present in the file are accessed both as a solo word and also as pair of words. Because, for example the word "bad" means negative but when someone writes "not bad" it refers to as positive. In such cases considering single word for training data will work otherwise. So words in pairs are checked to find the occurrence to modifiers before any adjective which if present which might provide a different meaning to the outlook.

```
X = reviews_df_preprocessed.iloc[:, -1].values
y = reviews_df_preprocessed.iloc[:, -2].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 42)
```

- **Training Data/ Evaluation:** The main chunk of code that does the whole evaluation of sentimental analysis based on the preprocessed data is a part of this. The following are the steps followed:

```
pipeline = Pipeline([
            ('vect', TfidfVectorizer(ngram_range = (1,2), stop_words = 'english', sublinear_tf = True)),
            ('chi', SelectKBest(score_func = chi2, k = 50000)),
            ('clf', LinearSVC(C = 1.0, penalty = 'l1', max_iter = 3000, dual = False, class_weight='balanced'))
        ])

model = pipeline.fit(X_train, y_train)
```

i) The Accuracy, Precision, Recall, and Evaluation time is calculated and displayed.

ii) Navie Bayes, Logistic Regression, Linear SVM and Random forest classifiers are applied on the dataset for evaluation of sentiments.

iii) Prediction of test data is done and Confusion Matrix of prediction isdisplayed. **iv)** Total positive and negative reviews are counted.

v) A review like sentence is taken as input on the console and if positive the console gives 1 as output and 0 for negative input.

6. Results and Sample Output

The ultimate outcome of this Training of Public reviews dataset is that, the machine is capable of judging whether an entered sentence bears positive response or negative response.

Precision (also called positive predictive value) is the fraction of relevant instances among the retrieved instances, while **Recall** (also known as sensitivity) is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Both precision and recall are therefore based on an understanding and measure of relevance.

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

F_1 score (also **F-score** or **F-measure**) is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score: p is the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive). The F_1 score is the harmonic average of the precision and recall, where an F_1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.

$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

In statistics, a **receiver operating characteristic curve**, i.e. **ROC curve**, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The Total Operating Characteristic (TOC) expands on the idea of ROC by showing the total information in the two-by-two contingency table for each threshold. ROC gives only two bits of relative information for each threshold, thus the TOC gives strictly more information than the ROC.

When using normalized units, the area under the curve (often referred to as simply the AUC) is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one (assuming 'positive' ranks higher than 'negative'). This can be seen as follows: the area under the curve is given by (the integral boundaries are reversed as large T has a lower value on the x-axis).

$$A = \int_{\infty}^{-\infty} \mathrm{TPR}(T)\mathrm{FPR}'(T)\,dT = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(T' > T) f_1(T') f_0(T)\,dT'\,dT = P(X_1 > X_0)$$

The machine evaluates the accuracy of training the data along with precision Recall and F_1

The Confusion matrix of evaluation is calculated.

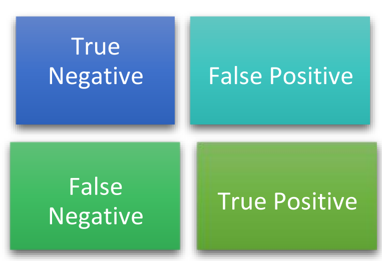
It is thus capable of judging an externally written review as positive or negative.

A positive review will be marked as [1], and a negative review will be hence marked as [0].

Results obtained using Hold-out Strategy(Train-Test split) [values rounded upto 2 decimal places]**.**

Name of classifier	F_1	Accuracy	Precision	Recall	ROC AUC
Multinomial NB	**85.25%**	**85.31%**	**85.56%**	**84.95%**	**85.31%**
Logistic Regression	**88.12%**	**88.05%**	**87.54%**	**88.72%**	**88.05%**
Linear SVC	**88.12%**	**88.11%**	**87.59%**	**88.80%**	**88.11%**
Random Forest	**82.43%**	**81.82%**	**79.74%**	**85.30%**	**81.83%**

The Confusion Matrix Format is as follows:



The Confusion Matrix of Each Classifier are as follows:

68556	11470
12032	67942

69928	10098
9023	70951

Classifier 1: Multinomial NB

Classifier 3: Liner SVC

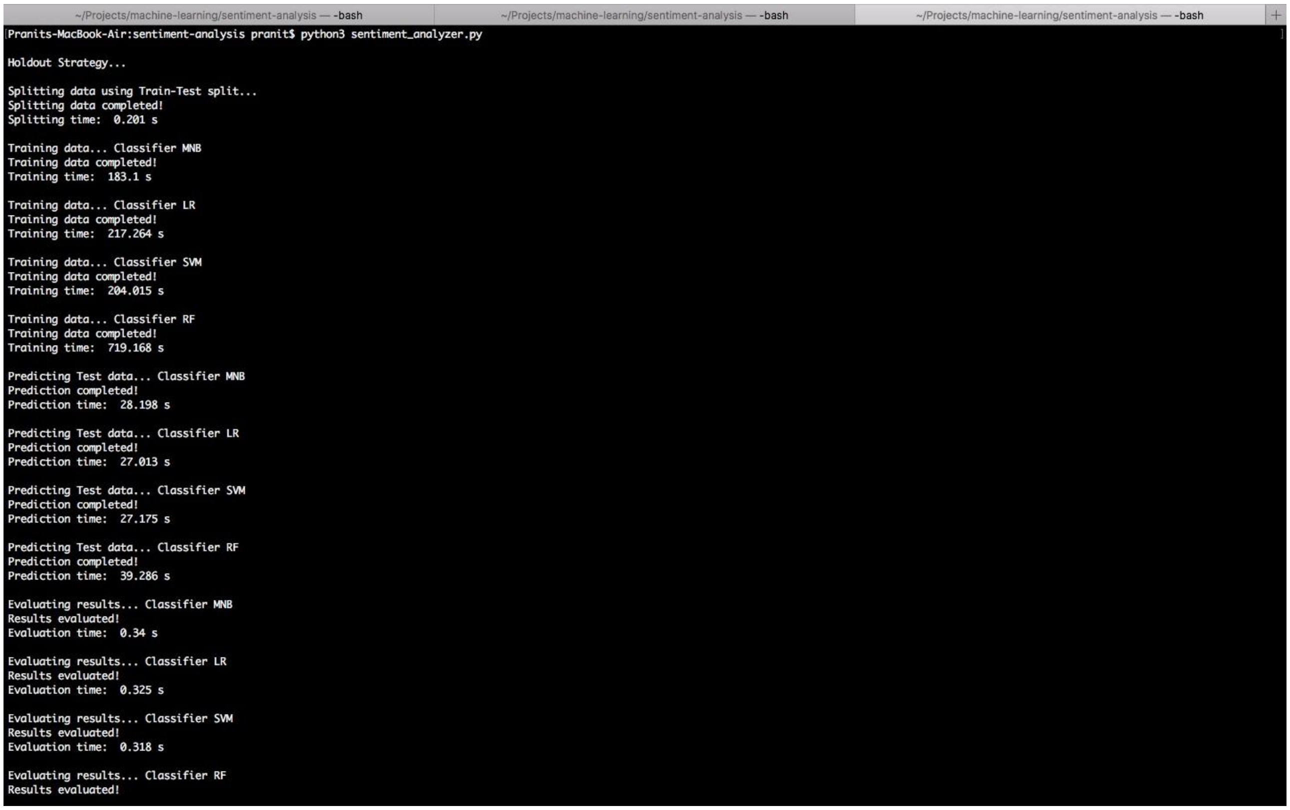
Classifier 2: Logistic Regression

69963	10063
8955	17019

62695	17331
11749	68225

Classifier 4: Random Forest

The following are the images of such sample output after successful dataset training using the classifiers:



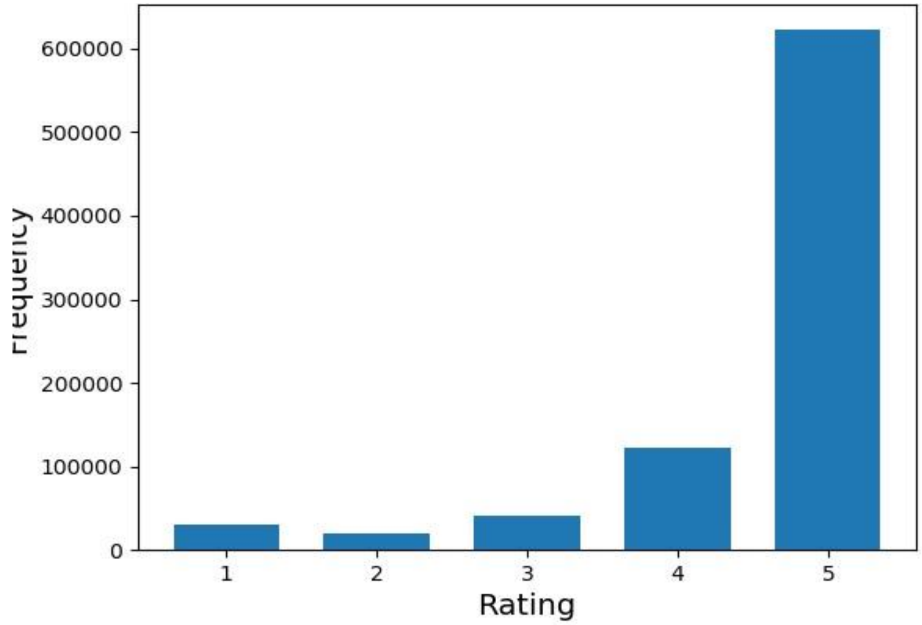
```
~/Projects/machine-learning/sentiment-analysis — -bash    ~/Projects/machine-learning/sentiment-analysis — -bash    ~/Projects/machine-learning/sentiment-analysis — -bash
Pranits-MacBook-Air:sentiment-analysis pranit$ python3 sentiment_analyzer.py

Holdout Strategy...

Splitting data using Train-Test split...
Splitting data completed!
Splitting time:  0.201 s

Training data... Classifier MNB
Training data completed!
Training time:  183.1 s

Training data... Classifier LR
Training data completed!
Training time:  217.264 s

Training data... Classifier SVM
Training data completed!
Training time:  204.015 s

Training data... Classifier RF
Training data completed!
Training time:  719.168 s

Predicting Test data... Classifier MNB
Prediction completed!
Prediction time:  28.198 s

Predicting Test data... Classifier LR
Prediction completed!
Prediction time:  27.013 s

Predicting Test data... Classifier SVM
Prediction completed!
Prediction time:  27.175 s

Predicting Test data... Classifier RF
Prediction completed!
Prediction time:  39.286 s

Evaluating results... Classifier MNB
Results evaluated!
Evaluation time:  0.34 s

Evaluating results... Classifier LR
Results evaluated!
Evaluation time:  0.325 s

Evaluating results... Classifier SVM
Results evaluated!
Evaluation time:  0.318 s

Evaluating results... Classifier RF
Results evaluated!
```

```
~/Projects/machine-learning/sentiment-analysis — -bash    ~/Projects/machine-learning/sentiment-analysis — -bash    ~/Projects/machine-learning/sentiment-analysis — -bash
Evaluating results... Classifier MNB
Results evaluated!
Evaluation time:  0.34 s

Evaluating results... Classifier LR
Results evaluated!
Evaluation time:  0.325 s

Evaluating results... Classifier SVM
Results evaluated!
Evaluation time:  0.318 s

Evaluating results... Classifier RF
Results evaluated!
Evaluation time:  0.315 s

Evaluation metrics of classifier MNB
Accuracy: 0.8531125
Precision: 0.8555633909232861
Recall: 0.8495511041088354
f1: 0.852546647760782
ROC AUC: 0.8531113429223856
Confusion Matrix: [[68556 11470]
 [12032 67942]]
Evaluation metrics of classifier LR
Accuracy: 0.88049375
Precision: 0.8754087033769695
Recall: 0.8871758321454473
f1: 0.8812529887034772
ROC AUC: 0.8804959209711317
Confusion Matrix: [[69928 10098]
 [ 9023 70951]]
Evaluation metrics of classifier SVM
Accuracy: 0.8811375
Precision: 0.8758910732345033
Recall: 0.8880261084852578
f1: 0.8819168487979336
ROC AUC: 0.8811397380703848
Confusion Matrix: [[69963 10063]
 [ 8955 71019]]
Evaluation metrics of classifier RF
Accuracy: 0.81825
Precision: 0.7974309224367666
Recall: 0.8530897541701052
f1: 0.8243218751887875
ROC AUC: 0.8182613192413517
Confusion Matrix: [[62695 17331]
 [11749 68225]]

Total number of observations: 160000
Positives in observation: 79974
Negatives in observation: 80026
Majority class is: 50.01624999999999%
Pranits-MacBook-Air:sentiment-analysis pranit$
```

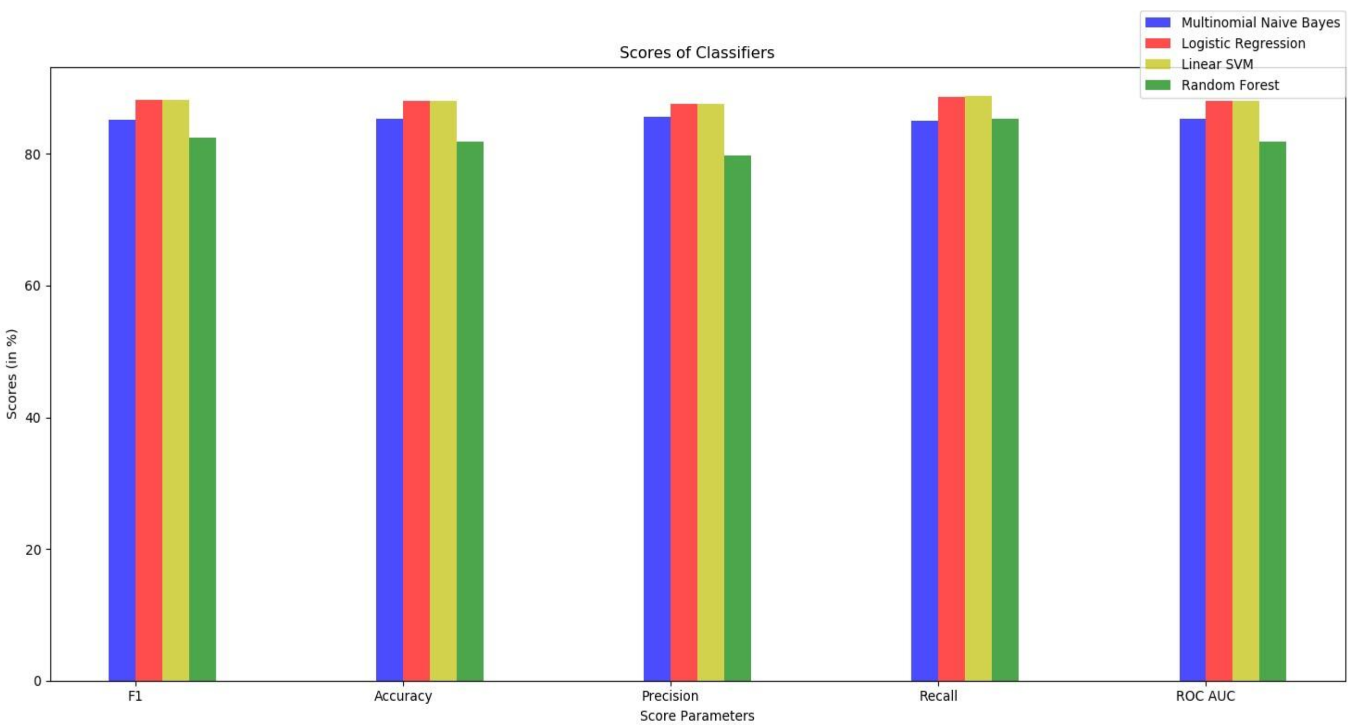
```
administrator@administrator-OptiPlex-3040:~/Desktop/sentiment_analysis$ python3 sentiment_analyzer.py
Preprocessing data...
Preprocessing data completed!
Preprocessing time:  0.131 s

Training data...
Training data completed!
Training time:  244.431 s

Predicting Test data...
Prediction completed!
Prediction time:  11.46 s

Evaluating results...
Accuracy: 0.94855693908754
Precision: 0.983433383243815
Recall: 0.9613014112497147
f1: 0.9722414612616284
Results evaluated!
Evaluation time:  0.084 s

Confusion matrix: [[  7575   2412]
 [  5764 143182]]

Total number of observations: 158933
Positives in observation: 148946
Negatives in observation: 9987
Majority class is: 93.7162200424078%
Worst product ever
[0]
```

```
administrator@administrator-OptiPlex-3040:~/Desktop/sentiment_analysis$ python3 sentiment_analyzer.py
Preprocessing data...
Preprocessing data completed!
Preprocessing time:  0.163 s

Training data...
Training data completed!
Training time:  239.406 s

Predicting Test data...
Prediction completed!
Prediction time:  11.402 s

Evaluating results...
Accuracy: 0.9486261506420944
Precision: 0.983467838868093
Recall: 0.9613416943053187
f1: 0.9722789017488227
Results evaluated!
Evaluation time:  0.086 s

Confusion matrix: [[  7580   2407]
 [  5758 143188]]

Total number of observations: 158933
Positives in observation: 148946
Negatives in observation: 9987
Majority class is: 93.7162200424078%
not a good product
[1]
```

The Bar Graph showing the Frequency of Ratings in the dataset



This Bar graph shows the score of each classifier after successful training. The parameters be: F_1 Score, Accuracy, Precision, Recall and Roc-Auc.



7. Conclusion

Sentiment analysis deals with the classification of texts based on the sentiments they contain. This article focuses on a typical sentiment analysis model consisting of three core steps, namely data preparation, review analysis and sentiment classification, and describes representative techniques involved in those steps.

Sentiment analysis is an emerging research area in text mining and computational linguistics, and has attracted considerable research attention in the past few years. Future research shall explore sophisticated methods for opinion and product feature extraction, as well as new classification models that can address the ordered labels property in rating inference. Applications that utilize results from sentiment analysis is also expected to emerge in the near future.

Appendix

Code:

<u>Loading the dataset:</u>

```
import json import pickle import
numpy as np from matplotlib
import pyplot as plt from textblob
import TextBlob

# fileHandler = open('datasets/reviews_digital_music.json', 'r')
# reviewDatas = fileHandler.read().split('\n')
```

```
# reviewText = []
# reviewRating = []

# for review in reviewDatas:
#       if review == "":
#             continue
#       r = json.loads(review)
#       reviewText.append(r['reviewText'])
#       reviewRating.append(r['overall'])

# fileHandler.close()
# saveReviewText = open('review_text.pkl', 'wb')
# saveReviewRating = open('review_rating.pkl','wb')
# pickle.dump(reviewText, saveReviewText) #
pickle.dump(reviewRating, saveReviewRating)
reviewTextFile = open('review_text.pkl', 'rb')
reviewRatingFile = open('review_rating.pkl', 'rb')
reviewText = pickle.load(reviewTextFile)
reviewRating = pickle.load(reviewRatingFile)
# print(len(reviewText))
# print(reviewText[0])
# print(reviewRating[0]) # ratings
= np.array(reviewRating)
 plt.hist(ratings, bins=np.arange(ratings.min(), ratings.max()+2)-0.5, rwidth=0.7)
plt.xlabel('Rating', fontsize=14)  plt.ylabel('Frequency', fontsize=14)
plt.title('Histogram of Ratings', fontsize=18) plt.show() lang = {} i = 0 for
review in reviewText:
      tb = TextBlob(review)
l = tb.detect_language()
if l != 'en':
```

```
                    lang.setdefault(l, [])
          lang[l].append(i)
print(i, l)          i += 1 print(lang)
```

Scrapping data:

```
from selenium import webdriver from
selenium.webdriver.chrome.options import Options from
bs4 import BeautifulSoup import openpyxl class
Review():          def __init__(self):
                    self.rating=""
                    self.info=""
                    self.review=""
def scrape():
          options = Options()       options.add_argument("--headless") # Runs Chrome in
headless mode.              options.add_argument('--no-sandbox') # # Bypass OS security
model   options.add_argument('start-maximized')           options.add_argument('disable-
infobars')          options.add_argument("--disable-extensions")
driver=webdriver.Chrome(executable_path=r'C:\chromedriver\chromedriver.exe')
          url='https://www.amazon.com/Moto-PLUS-5th-Generation-Exclusive/product-
reviews/B0785NN142/ref=cm_cr_arp_d_paging_btm_2?ie=UTF8&reviewerType=all_reviews&pageNumb
er=5'
          driver.get(url)


          soup=BeautifulSoup(driver.page_source,'lxml')
ul=soup.find_all('div',class_='a-section review')
review_list=[]   for d in ul:
                    a=d.find('div',class_='a-row')
sib=a.findNextSibling()
                    b=d.find('div',class_='a-row a-spacing-medium review-data')
                    '''print sib.text'''
```

```
            new_r=Review()
new_r.rating=a.text          new_r.info=sib.text
      new_r.review=b.text

            review_list.append(new_r)
driver.quit()      return review_list def
main():
      m = scrape()
      i=1 for r in
      m:

            book = openpyxl.load_workbook('Sample.xlsx')          sheet =
book.get_sheet_by_name('Sample Sheet')          sheet.cell(row=i, column=1).value = r.rating
      sheet.cell(row=i, column=1).alignment = openpyxl.styles.Alignment(horizontal='center',
vertical='center', wrap_text=True)
            sheet.cell(row=i, column=3).value = r.info
            sheet.cell(row=i, column=3).alignment =
openpyxl.styles.Alignment(horizontal='center', vertical='center', wrap_text=True)
sheet.cell(row=i, column=5).value = r.review.encode('utf-8')          sheet.cell(row=i,
column=5).alignment = openpyxl.styles.Alignment(horizontal='center', vertical='center',
wrap_text=True)
            book.save('Sample.xlsx')
            i=i+1          if
__name__ == '__main__':
  main()
```

Preprocessing Data:

```
import string from nltk.corpus import stopwords as sw from nltk.corpus import wordnet
as wn from nltk import wordpunct_tokenize from nltk import sent_tokenize from nltk
import WordNetLemmatizer from nltk import pos_tag class NltkPreprocessor:     def
__init__(self, stopwords = None, punct = None, lower = True, strip = True):
self.lower = lower          self.strip = strip
```

```
        self.stopwords = stopwords or set(sw.words('english'))
        self.punct = punct or set(string.punctuation)
        self.lemmatizer = WordNetLemmatizer()

    def tokenize(self, document):
        tokenized_doc = []

        for sent in sent_tokenize(document):                for token, tag in
pos_tag(wordpunct_tokenize(sent)):                token = token.lower() if
self.lower else token                token = token.strip() if self.strip else
token                token = token.strip('_0123456789') if self.strip else token
                # token = re.sub(r'\d+', '', token)
                if token in self.stopwords:
                    continue
                if all(char in self.punct for char in token):
                    continue

                lemma = self.lemmatize(token, tag)
tokenized_doc.append(lemma)

        return tokenized_doc
    def lemmatize(self, token,
tag):
        tag = {
            'N': wn.NOUN,
            'V': wn.VERB,
            'R': wn.ADV,
            'J': wn.ADJ
        }.get(tag[0], wn.NOUN)
        return self.lemmatizer.lemmatize(token, tag)
```

Sentiment Analysis:

```
import ast import numpy as np import pandas as pd
import re from nltk.corpus import stopwords from
nltk.stem import SnowballStemmer from
sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, chi2, SelectPercentile, f_classif from
sklearn.feature_extraction.text import TfidfVectorizer from sklearn.pipeline import Pipeline from
sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score,
confusion_matrix from sklearn.svm import LinearSVC # from textblob import TextBlob from time
import time

def getInitialData(data_file):
        print('Fetching initial data...')
        t = time()

        i = 0    df = {}        with
open(data_file, 'r') as file_handler:
                for review in file_handler.readlines():
        df[i] = ast.literal_eval(review)
                        i += 1

        reviews_df = pd.DataFrame.from_dict(df, orient = 'index')
reviews_df.to_pickle('reviews_digital_music.pickle')  print('Fetching data completed!')  print('Fetching time:
', round(time()-t, 3), 's\n')



# def filterLanguage(text):
#       text_blob = TextBlob(text)
#       return text_blob.detect_language()

def    prepareData(reviews_df):
print('Preparing   data...')    t   =
time()
```

```
reviews_df.rename(columns = {"overall" : "reviewRating"}, inplace=True)
reviews_df.drop(columns = ['reviewerID', 'asin', 'reviewerName', 'helpful', 'summary', 'unixReviewTime',
'reviewTime'], inplace = True)


reviews_df = reviews_df[reviews_df.reviewRating != 3.0] # Ignoring 3-star reviews -> neutral
reviews_df = reviews_df.assign(sentiment = np.where(reviews_df['reviewRating'] >= 4.0, 1, 0)) # 1 ->
Positive, 0 -> Negative

stemmer = SnowballStemmer('english')
stop_words = stopwords.words('english')

# print(len(reviews_df.reviewText))
# filterLanguage = lambda text: TextBlob(text).detect_language()
# reviews_df = reviews_df[reviews_df['reviewText'].apply(filterLanguage) == 'en']
# print(len(reviews_df.reviewText))

reviews_df = reviews_df.assign(cleaned = reviews_df['reviewText'].apply(lambda text: '
'.join([stemmer.stem(w) for w in re.sub('[^a-z]+|(quot)+', ' ', text.lower()).split() if w not in stop_words])))
reviews_df.to_pickle('reviews_digital_music_preprocessed.pickle')

print('Preparing data completed!')
print('Preparing time: ', round(time()-t, 3), 's\n')

def preprocessData(reviews_df_preprocessed):
 print('Preprocessing data...')  t =
time()

X = reviews_df_preprocessed.iloc[:, -1].values
y = reviews_df_preprocessed.iloc[:, -2].values
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 42)

    print('Preprocessing data completed!')
print('Preprocessing time: ', round(time()-t, 3), 's\n')

    return X_train, X_test, y_train, y_test

def evaluate(y_test, prediction):
print('Evaluating results...')
    t = time()

    print('Accuracy: {}'.format(accuracy_score(y_test, prediction)))
print('Precision: {}'.format(precision_score(y_test, prediction)))
print('Recall: {}'.format(recall_score(y_test, prediction)))          print('f1:
{}'.format(f1_score(y_test, prediction)))

    print('Results evaluated!')
print('Evaluation time: ', round(time()-t, 3), 's\n')

# getInitialData('datasets/reviews_digital_music.json')
# reviews_df = pd.read_pickle('reviews_digital_music.pickle')

# prepareData(reviews_df) reviews_df_preprocessed =
pd.read_pickle('reviews_digital_music_preprocessed.pickle')
# print(reviews_df_preprocessed.isnull().values.sum()) # Check for any null values

X_train, X_test, y_train, y_test = preprocessData(reviews_df_preprocessed)

print('Training data...') t
= time()
```

```
pipeline = Pipeline([
                                        ('vect', TfidfVectorizer(ngram_range = (1,2), stop_words = 'english',
sublinear_tf = True)),
                                        ('chi', SelectKBest(score_func = chi2, k = 50000)),
                                        ('clf', LinearSVC(C = 1.0, penalty = 'l1', max_iter = 3000, dual = False,
class_weight = 'balanced'))
                              ])

model = pipeline.fit(X_train, y_train)

print('Training data completed!') print('Training
time: ', round(time()-t, 3), 's\n')

print('Predicting Test data...') t
= time()

prediction = model.predict(X_test)

print('Prediction completed!')
print('Prediction time: ', round(time()-t, 3), 's\n')

evaluate(y_test, prediction)

print('Confusion matrix: {}'.format(confusion_matrix(y_test, prediction)))
print() l = (y_test == 0).sum() + (y_test ==
1).sum() s = y_test.sum()
print('Total number of observations: ' + str(l))
print('Positives in observation: ' + str(s)) print('Negatives
in observation: ' + str(l - s))
print('Majority class is: ' + str(s / l * 100) + '%')
```

Graph Plotting Code: import numpy as np import matplotlib.pyplot as plt from matplotlib.ticker import MaxNLocator from collections import namedtuple n_groups = 5

score_MNB = (85.25, 85.31, 85.56, 84.95, 85.31)

score_LR = (88.12, 88.05, 87.54, 88.72, 88.05)

score_LSVC=(88.12, 88.11, 87.59, 88.80, 88.11)

score_RF=(82.43, 81.82, 79.74, 85.30, 81.83)

#n1=(score_MNB[0], score_LR[0], score_LSVC[0], score_RF[0])

#n2=(score_MNB[1], score_LR[1], score_LSVC[1], score_RF[1])

#n3=(score_MNB[2], score_LR[2], score_LSVC[2], score_RF[2])

#n4=(score_MNB[3], score_LR[3], score_LSVC[3], score_RF[3])

#n5=(score_MNB[4], score_LR[4], score_LSVC[4], score_RF[4])

fig, ax = plt.subplots() index = np.arange(n_groups) bar_width =

0.1 opacity = 0.7 error_config = {'ecolor': '0.3'} rects1 =

ax.bar(index,score_MNB, bar_width, alpha=opacity,

color='b',

error_kw=error_config,

label='Multinomial Naive Bayes') z=index

+ bar_width rects2 = ax.bar(z, score_LR,

bar_width, alpha=opacity,

color='r', error_kw=error_config,

label='Logistic Regression') z=z+

bar_width

rects3 = ax.bar(z, score_LSVC, bar_width,

alpha=opacity, color='y',

error_kw=error_config,

label='Linear SVM') z=z+ bar_width

rects4 = ax.bar(z, score_RF, bar_width,

alpha=opacity, color='g',

error_kw=error_config,

```
label='Random Forest') ax.set_xlabel('Score
Parameters') ax.set_ylabel('Scores (in %)')
ax.set_title('Scores of Classifiers')
ax.set_xticks(index + bar_width / 2)
ax.set_xticklabels(('F1', 'Accuracy', 'Precision', 'Recall', 'ROC AUC'))
ax.legend(bbox_to_anchor=(1, 1.02), loc=5, borderaxespad=0)
fig.tight_layout() plt.show()
```

	tweet_id	airline_sentimen	airline_sentimen	negativereason	negativereason_	airline	airline_sentimen	name	negativereason_	retweet_count	text	tweet_coord	tweet_created	tweet_location	user_timezone
2	5703061336777	neutral	1			Virgin America		cairdin		0	@VirginAmerica What @dhepbur		2015-02-24 11:35:52 -0800		Eastern Time (US & Canada)
3	5703011308881	positive	0.3486		0	Virgin America		jnardino		0	@VirginAmerica plus you've adde		2015-02-24 11:15:59 -0800		Pacific Time (US & Canada)
4	5703010836728	neutral	0.6837			Virgin America		yvonnalynn		0	@VirginAmerica I didn't today... M		2015-02-24 11:1	Lets Play	Central Time (US & Canada)
5	5703010314076	negative	1	Bad Flight	0.7033	Virgin America		jnardino		0	@VirginAmerica it's really aggres		2015-02-24 11:15:36 -0800		Pacific Time (US & Canada)
6	5703008170744	negative	1	Can't Tell	1	Virgin America		jnardino		0	@VirginAmerica and it's a really b		2015-02-24 11:14:45 -0800		Pacific Time (US & Canada)
7	5703007670741	negative	1	Can't Tell	0.6842	Virgin America		jnardino		0	@VirginAmerica seriously would p it's really the only bad thing about		2015-02-24 11:14:33 -0800		Pacific Time (US & Canada)
8	5703006169013	positive	0.6745		0	Virgin America		cjmcginnis		0	@VirginAmerica yes, nearly every		2015-02-24 11:1	San Francisco C	Pacific Time (US & Canada)
9	5703002485533	neutral	0.634			Virgin America		pilot		0	@VirginAmerica Really missed a		2015-02-24 11:1	Los Angeles	Pacific Time (US & Canada)
10	5702999532869	positive	0.6559			Virgin America		dhepburn		0	@virginamerica Well, I didn't…but		2015-02-24 11:1	San Diego	Pacific Time (US & Canada)
11	5702954596312	positive	1			Virgin America		YupitsTate		0	@VirginAmerica it was amazing, a		2015-02-24 10:5	Los Angeles	Eastern Time (US & Canada)
12	5702941891430	neutral	0.6769		0	Virgin America		idk_but_youtube		0	@VirginAmerica did you know tha		2015-02-24 10:4	1/1 loner squad	Eastern Time (US & Canada)
13	5702897244532	positive	1			Virgin America		HyperCamiLax		0	@VirginAmerica I <3 pretty grap		2015-02-24 10:3	NYC	America/New_York
14	5702895840614	positive	1			Virgin America		HyperCamiLax		0	@VirginAmerica This is such a gr		2015-02-24 10:3	NYC	America/New_York
15	5702874084381	positive	0.6451			Virgin America		mollanderson		0	@VirginAmerica @virginmedia I'm		2015-02-24 10:21:28 -0800		Eastern Time (US & Canada)
16	5702859048095	positive	1			Virgin America		sjespers		0	@VirginAmerica Thanks!		2015-02-24 10:1	San Francisco, (	Pacific Time (US & Canada)
17	5702824691210	negative	0.6842	Late Flight	0.3684	Virgin America		smartwatermelon		0	@VirginAmerica SFO-PDX sched		2015-02-24 10:0	palo alto, ca	Pacific Time (US & Canada)
18	5702777243857	positive	1			Virgin America		ItzBrianHunty		0	@VirginAmerica So excited for m		2015-02-24 09:4	west covina	Pacific Time (US & Canada)
19	5702769173011	negative	1	Bad Flight	1	Virgin America		heatherovieda		0	@VirginAmerica I flew from NYC		2015-02-24 09:3	this place called	Eastern Time (US & Canada)
20	5702706846199	positive	1			Virgin America		thebrandiray		0	I ❤ flying @VirginAmerica. 😊👍		2015-02-24 09:1	Somewhere cele	Atlantic Time (Canada)
21	5702679566487	positive	1			Virgin America		JNLpierce		0	@VirginAmerica you know what w		2015-02-24 09:0	Boston | Walthar	Quito
22	5702658835133	negative	0.6705	Can't Tell	0.3614	Virgin America		MISSGJ		0	@VirginAmerica why are your firs		2015-02-24 08:55:56 -0800		
23	5702641451168	positive	1			Virgin America		DT_Les		0	@VirginAmerica	[40.74804263, -7	2015-02-24 08:49:01 -0800		
24	5702594202878	positive	1			Virgin America		ElvinaBeck		0	@VirginAmerica I love the hipster		2015-02-24 08:3	Los Angeles	Pacific Time (US & Canada)
25	5702588222975	neutral	1			Virgin America		rjlynch21086		0	@VirginAmerica will you be makir		2015-02-24 08:2	Boston, MA	Eastern Time (US & Canada)
26	5702565535020	negative	1	Customer Servic	0.3557	Virgin America		ayeevickiee		0	@VirginAmerica you guys messe		2015-02-24 08:1	714	Mountain Time (US & Canada)
27	5702491024049	negative	1	Customer Servic	1	Virgin America		Leora13		0	@VirginAmerica status match pro		2015-02-24 07:49:15 -0800		
28	5702396328073	negative	1	Can't Tell	0.6614	Virgin America		meredithjlynn		0	@VirginAmerica What happened		2015-02-24 07:11:37 -0800		
29	5702178315576	neutral	0.6854			Virgin America		AdamSinger		0	@VirginAmerica do you miss me?		2015-02-24 05:4	San Francisco, (	Central Time (US & Canada)
30	5702078864937	negative	1	Bad Flight	1	Virgin America		blackjackpro911		0	@VirginAmerica	[42.361016, -71.	2015-02-24 05:0	San Mateo, CA & Las Vegas, NV	
31	5701245961809	neutral	0.615		0	Virgin America		TenantsUpstairs		0	@VirginAmerica	[33.94540417, -	2015-02-23 23:3	Brooklyn	Atlantic Time (Canada)
32	5701140218542	negative	1	Flight Booking P	1	Virgin America		jordanpichler		0	@VirginAmerica hi! I just bked a c		2015-02-23 22:52:29 -0800		Vienna
33	5700947013714	neutral	1			Virgin America		JCervantezzz		0	@VirginAmerica Are the hours of		2015-02-23 21:3	California, San F	Pacific Time (US & Canada)
34	5700884041566	negative	1	Customer Servic	1	Virgin America		Cuschoolie1		0	@VirginAmerica	[33.94209449, -	2015-02-23 21:1	Washington DC	Quito
35	5700845827808	negative	1	Customer Servic	1	Virgin America		amanduhmccarty		0	@VirginAmerica awaiting my retu		2015-02-23 20:55:30 -0800		Pacific Time (US & Canada)
36	5700767929936	positive	1			Virgin America		NorthTxHomeTeam		0	@VirginAmerica	[33.2145038, -96	2015-02-23 20:2	Texas	Central Time (US & Canada)
37	5700519912773	neutral	0.6207			Virgin America		miaerolinea		0	Nice RT @VirginAmerica: Vibe wi		2015-02-23 18:4	Worldwide	Caracas
38	5700513815343	positive	1			Virgin America		Nicsplace		0	@VirginAmerica Moodlighting is t		2015-02-23 18:4	Central Texas	
39	5700453935656	positive	1			Virgin America		Nicsplace		0	@VirginAmerica @freddieawards		2015-02-23 18:1	Central Texas	
40	5700389414971	neutral	0.6791		0	Virgin America		elisha_malulani		0	@VirginAmerica when can I book		2015-02-23 17:5	i'm creating a m	Pacific Time (US & Canada)
41	5700358768450	negative	1	Customer Servic	1	Virgin America		DannyDouglass		0	@VirginAmerica Your chat suppo		2015-02-23 17:4	San Francisco, (	Pacific Time (US & Canada)
42	5700335933946	positive	0.6639			Virgin America		jamesferrandini		0	@VirginAmerica View of downtow		2015-02-23 17:32:54 -0800		
43	5700254823448	negative	0.6688	Flight Booking P	0.6688	Virgin America		will_lenzenjr		0	@VirginAmerica Hey, first time fly		2015-02-23 17:0	Iowa City	Central Time (US & Canada)
44	5700163042849	neutral	1			Virgin America		GottAmanda		0	@VirginAmerica	[34.0219817, -11	2015-02-23 16:2	Los Angeles	
45	5700154087884	neutral	0.6578		0	Virgin America		KGervaise		0	@VirginAmerica I have an unuse		2015-02-23 16:2	Georgia	Pacific Time (US & Canada)
46	5700135236500	neutral	1			Virgin America		papamurat		0	@VirginAmerica are flights leavin		2015-02-23 16:13:09 -0800		
47	5700122575490	positive	1			Virgin America		arieldaie		0	@VirginAmerica I'm #elevategold		2015-02-23 16:0	Los Angeles	
48	5700113414838	neutral	0.6799			Virgin America		vacations7		0	@VirginAmerica DREAM http://t.		2015-02-23 16:0	Turks and caicos	
49	5700105717072	positive	1			Virgin America		ChelseaPoe666		0	@VirginAmerica wow this just ble		2015-02-23 16:0	Oakland via Mid	Atlantic Time (Canada)
50	5700105394993	neutral	1			Virgin America		BobGlavinVO		0	@VirginAmerica @ladygaga @ca		2015-02-23 16:0	New York, NY	Eastern Time (US & Canada)
51	5700097134478	neutral	0.6436			Virgin America		lisaaiko		0	@VirginAmerica @ladygaga @ca		2015-02-23 15:58:00 -0800		
52	5700090354553	neutral	0.6764		0	Virgin America		grantbrowne		0	@VirginAmerica Is flight 769 on it		2015-02-23 15:5	Worldwide	Central Time (US & Canada)
53	5700068860129	positive	0.657			Virgin America		joyabsalon		0	@VirginAmerica @ladygaga @ca		2015-02-23 15:4	Northern Virginia	Eastern Time (US & Canada)
54	5700043917318	neutral	1			Virgin America		2v		0	@VirginAmerica wish you flew ou		2015-02-23 15:3	Los Angeles / At	Eastern Time (US & Canada)
55	5700011949004	neutral	0.7118		0	Virgin America		KSmithFoundHere		0	@VirginAmerica @ladygaga @ca		2015-02-23 15:24:09 -0800		Atlantic Time (Canada)
56	5700000716448	neutral	1			Virgin America		papamurat		0	@VirginAmerica Will flights be lea		2015-02-23 15:19:41 -0800		
57	5699964122865	negative	0.6939	Flight Booking P	0.6939	Virgin America		murphicus		0	@VirginAmerica hi! i'm so excited		2015-02-23 15:0	new york, new y	Eastern Time (US & Canada)
58	5699962454621	positive	1			Virgin America		VinnieFerra		0	@VirginAmerica you know it. Nee		2015-02-23 15:0	brooklyn, Ny	Pacific Time (US & Canada)
59	5699902226094	positive	0.635			Virgin America		KevinDemsi		0	@VirginAmerica @ladygaga @ca		2015-02-23 14:4	Bali, Republic of	Kuala Lumpur
60	5699901632096	neutral	0.7007			Virgin America		giffgaffman		0	@VirginAmerica @ladygaga @ca		2015-02-23 14:4	UK, USA.	
61	5699895044313	neutral	1			Virgin America		HanlonBrothers		0	@VirginAmerica New marketing s		2015-02-23 14:3	Gold Coast, Aus	Brisbane
62	5699893216980	neutral	1			Virgin America		emilybg78		0	@VirginAmerica @ladygaga @ca		2015-02-23 14:3	Stockton, CA	Arizona
63	5699890345015	negative	1	Customer Servic	1	Virgin America		rachie1126		0	@VirginAmerica I called a 3-4 we		2015-02-23 14:3	New York, NY	Eastern Time (US & Canada)
64	5699876224848	neutral	0.6858			Virgin America		adawson66		0	@VirginAmerica	[33.57963333, -	2015-02-23 14:30:13 -0800		
65	5699867825670	neutral	1			Virgin America		SocialPLC		0	@VirginAmerica @LadyGaga @C		2015-02-23 14:2	Twin Cities, Minr	Eastern Time (US & Canada)
66	5699863480415	positive	1			Virgin America		jeffreymace01		0	@VirginAmerica @ladygaga @ca		2015-02-23 14:25:09 -0800		
67	5699823076347	neutral	0.6814		0	Virgin America		1stcrown		0	@VirginAmerica Flight 0736 DAL		2015-02-23 14:0	USA	Central Time (US & Canada)
68	5699766201585	negative	1	Customer Servic	1	Virgin America		onerockgypsy		0	@VirginAmerica heyyyy guyyyys.		2015-02-23 13:4	next city	Pacific Time (US & Canada)
69	5699738213961	negative	1	Late Flight	0.6789	Virgin America		noelduan		0	@VirginAmerica Hi, Virgin! I'm on		2015-02-23 13:3	SF ↔ NY	Eastern Time (US & Canada)
70	5699725084992	positive	0.6922			Virgin America		Travelzoo		0	@VirginAmerica Congrats on win		2015-02-23 13:3	New York, NY	Pacific Time (US & Canada)
71	5699670199587	negative	1	Lost Luggage	1	Virgin America		gianagon		0	@VirginAmerica	[40.6413712, -73	2015-02-23 13:0	New York + Pan	Eastern Time (US & Canada)
72	5699618662246	neutral	1			Virgin America		bxchen		0	@virginamerica Need to change r		2015-02-23 12:4	San Francisco, (	Eastern Time (US & Canada)
73	5699498911636	neutral	0.6492		0	Virgin America		seimatrun		0	@VirginAmerica I emailed your cu		2015-02-23 12:0	Los Angeles	
74	5699489668733	neutral	1			Virgin America		jamied7		0	@VirginAmerica hi I just booked a		2015-02-23 11:5	London, England	London
75	5699463621266	negative	1	Flight Attendant	0.3516	Virgin America		seimatrun		0	@VirginAmerica your airline is aw		2015-02-23 11:4	Los Angeles	
76	5699429036838	positive	1			Virgin America		mrmichaellay		0	@VirginAmerica	[36.08457854, -	2015-02-23 11:3	Floridian from Cl	Eastern Time (US & Canada)
77	5699419574907	positive	1			Virgin America		TaylorLumsden		0	@VirginAmerica awesome. I flew		2015-02-23 11:2	Dallas, Texas	Mountain Time (US & Canada)
78	5699408349944	neutral	1			Virgin America		campusmoviefest		0	@VirginAmerica Or watch some c		2015-02-23 11:2	USA	Eastern Time (US & Canada)
79	5699403237465	neutral	1			Virgin America		TaylorLumsden		0	@VirginAmerica first time flying y		2015-02-23 11:2	Dallas, Texas	Mountain Time (US & Canada)
80	5699352320333	negative	1	Customer Servic	1	Virgin America		meme_meng		0	@VirginAmerica what is going on		2015-02-23 11:02:02 -0800		
81	5699343958654	neutral	1			Virgin America		kyle_romanoff		0	@VirginAmerica what happened t		2015-02-23 10:58:43 -0800		
82	5699338169633	negative	1	Customer Servic	1	Virgin America		GunsNDip		0	@VirginAmerica why can't you su		2015-02-23 10:56:25 -0800		Pacific Time (US & Canada)
83	5699337779311	positive	1			Virgin America		artisticwritr87		0	@VirginAmerica I've applied more		2015-02-23 10:5	Seattle, WA	Pacific Time (US & Canada)
84	5699334055063	negative	0.6792	Late Flight	0.3477	Virgin America		arieldaie		0	@VirginAmerica you're the best!!		2015-02-23 10:5	Los Angeles	
85	5699333605643	negative	1	Can't Tell	1	Virgin America		GunsNDip		0	@VirginAmerica I have no interes		2015-02-23 10:54:36 -0800		Pacific Time (US & Canada)
86	5699292431460	negative	1	Can't Tell	1	Virgin America		GunsNDip		0	@VirginAmerica it was a disappoi		2015-02-23 10:38:14 -0800		Pacific Time (US & Canada)
87	5699269988243	negative	1	Flight Booking P	1	Virgin America		jsatk		0	@VirginAmerica	[0.0, 0.0]	2015-02-23 10:2	Lower Pacific He	Pacific Time (US & Canada)
88	5699233949904	neutral	0.6705		0	Virgin America		serenaklal		0	@VirginAmerica Can't bring up m		2015-02-23 10:1	Chicago	Eastern Time (US & Canada)
89	5699220085882	neutral	1			Virgin America		openambit1		0	@VirginAmerica Random Q: what		2015-02-23 10:09:30 -0800		
90	5699208249053	neutral	0.6545		0	Virgin America		cabowine		0	@VirginAmerica I <3 Flying VA		2015-02-23 10:0	Los Cabos,Mexi	Arizona
91	5699190412441	negative	1	Can't Tell	0.6513	Virgin America		MaryAnnTaylorT		0	@VirginAmerica Why is the site d		2015-02-23 09:5	New York, NY	Arizona
92	5699159411920	neutral	1			Virgin America		RamotControl		0	@VirginAmerica "You down with F		2015-02-23 09:45:23 -0800		Pacific Time (US & Canada)
93	5699133394274	neutral	0.6639		0	Virgin America		losermelon		0	@VirginAmerica hi, i did not get p		2015-02-23 09:35:03 -0800		
94	5699118169370	negative	1	Cancelled Flight	1	Virgin America		AlisonK33774854		0	@VirginAmerica I like the TV and		2015-02-23 09:29:00 -0800		
95	5699116741587	negative	1	Late Flight	1	Virgin America		GunsNDip		0	@VirginAmerica just landed in LA		2015-02-23 09:28:26 -0800		Pacific Time (US & Canada)
96	5699112189425	neutral	0.6765		0	Virgin America		yazdanagh		0	@VirginAmerica why is flight 345		2015-02-23 09:26:37 -0800		
97	5699109818680	negative	1	Customer Servic	0.6863	Virgin America		MerchEngines		0	@VirginAmerica Is it me, or is you		2015-02-23 09:2	Los Angeles, CA	Arizona
98	5699092245216	negative	1	Customer Servic	0.6771	Virgin America		ColorCartel		0	@VirginAmerica I can't check in o		2015-02-23 09:1	Austin, TX	Mountain Time (US & Canada)
99	5699073364850	negative	1	Can't Tell	0.659	Virgin America		MustBeSpoken		0	@VirginAmerica - Let 2 scanned i		2015-02-23 09:11:12 -0800		
100	5698968056110	negative	1	Flight Booking P	0.6714	Virgin America		mattbunk		0	@virginamerica What is your pho		2015-02-23 08:2	Sterling Heights,	Eastern Time (US & Canada)
101	5698944496203	negative	1	Customer Servic	1	Virgin America		louisjenny		0	@VirginAmerica is anyone doing a		2015-02-23 08:1	Washington DC	Quito
102	5698944070019	neutral	1			Virgin America		STravelsW		0	@VirginAmerica trying to add my		2015-02-23 08:1	Manhattan Beach, CA	
103	5698921996906	negative	1	Late Flight	0.6882	Virgin America		GunsNDip		0	@VirginAmerica why must a trave		2015-02-23 08:11:03 -0800		Pacific Time (US & Canada)
104	5698914692107	neutral	1			Virgin America		joeyrenagade		0	@VirginAmerica check out new m		2015-02-23 08:0	Greater Los Angeles	
105	5698914361008	negative	0.6925	Late Flight	0.3521	Virgin America		mrmichaellay		0	@virginamerica	[0.0, 0.0]	2015-02-23 08:0	Floridian from Cl	Eastern Time (US & Canada)
106	5698873107134	negative	1	Late Flight	0.3486	Virgin America		GunsNDip		0	@VirginAmerica your no Late Flig		2015-02-23 07:51:37 -0800		Pacific Time (US & Canada)
107	5698870494460	positive	1			Virgin America		TheDuchessSF		0	@VirginAmerica - amazing custor		2015-02-23 07:5	Online	Pacific Time (US & Canada)
108	5698845517128	negative	1	Customer Servic	1	Virgin America		BeLeather		0	@VirginAmerica	[0.0, 0.0]	2015-02-23 07:40:39 -0800		Pacific Time (US & Canada)
109	5698844078524	negative	1	Flight Booking P	0.6366	Virgin America		BeLeather		0	@VirginAmerica	[0.0, 0.0]	2015-02-23 07:40:05 -0800		Pacific Time (US & Canada)
110	5698815485157	neutral	0.6593		0	Virgin America		drcaseydrake		0	@VirginAmerica	[37.79374402, -	2015-02-23 07:2	Dallas, TX	
111	5698736697003	positive	0.6823		0	Virgin America		flyfromWAS		0	@VirginAmerica has getaway dea		2015-02-23 06:5	Washington, DC	Eastern Time (US & Canada)
112	5698736681317	neutral	1			Virgin America		flyfromSEA		0	@VirginAmerica has getaway dea		2015-02-23 06:5	Seattle	Central Time (US & Canada)
113	5698736656402	positive	0.6806			Virgin America		flyfromNYC		0	@VirginAmerica has getaway dea		2015-02-23 06:5	New York City, N	Central Time (US & Canada)
114	5698736646127	neutral	0.6529			Virgin America		flyfromLAX		0	@VirginAmerica has getaway dea		2015-02-23 06:5	Los Angeles, CA	Central Time (US & Canada)
115	5698720586136	positive	0.678			Virgin America		Silvanabfer		0	@VirginAmerica Have a great we		2015-02-23 06:51:01 -0800		
116	5698612097819	positive	0.3482		0	Virgin America		AdamJdubs		0	@VirginAmerica come back to #P		2015-02-23 06:0	Earth	Eastern Time (US & Canada)
117	5698479201926	negative	1	Late Flight	1	Virgin America		nicholas_v		0	@VirginAmerica	[26.074379, -80.	2015-02-23 05:15:06 -0800		Eastern Time (US & Canada)
118	5698143397853	positive	1			Virgin America		tstashajones		0	@VirginAmerica is the best airline		2015-02-23 03:0	Halifax, Nova Sc	Eastern Time (US & Canada)
119	5697776073711	positive	1			Virgin America		SkateMamas		0	@VirginAmerica and again! Anoth		2015-02-23 00:3	Los Angeles, CA	
120	5697740782334	positive	1			Virgin America		dngoo		0	@VirginAmerica your beautiful fro		2015-02-23 00:2	Near a park, wal	Pacific Time (US & Canada)
121	5697703636235	positive	1			Virgin America		SamBrittenham		0	@VirginAmerica Love the team ru		2015-02-23 00:0	USA	Eastern Time (US & Canada)
122	5697483167763	negative	0.6832	Customer Servic	0.3773	Virgin America		usagibrian		0	@VirginAmerica Use another bro		2015-02-22 22:3	San Francisco C	Pacific Time (US & Canada)
123	5697412217839	negative	1	Flight Booking P	0.6767	Virgin America		usagibrian		0	@VirginAmerica And now the fligl		2015-02-22 22:1	San Francisco C	Pacific Time (US & Canada)
124	5697376036179	negative	1	Customer Servic	0.6527	Virgin America		KindofLuke		0	@VirginAmerica I like the custom		2015-02-22 21:56:44 -0800		
125	5697141277922	positive	1			Virgin America		ptbrodie		0	@VirginAmerica thanks to your o		2015-02-22 20:2	San Francisco	
126	5696751443538	positive	1			Virgin America		cheryleng		0	@VirginAmerica	[33.9469039, -11	2015-02-22 17:48:33 -0800		Pacific Time (US & Canada)
127	5696743581359	neutral	1			Virgin America		HishamSharaby		0	@VirginAmerica Do you provide c		2015-02-22 17:4	New York	Eastern Time (US & Canada)
128	5696664772650	negative	0.6703	Customer Servic	0.6703	Virgin America		ChrisFordisHere		0	@VirginAmerica	[51.04345575, -	2015-02-22 17:1	NYC	Tehran
129	5696524979477	positive	1			Virgin America		JKF1897		0	@VirginAmerica completely awes		2015-02-22 16:18:33 -0800		
130	5696491164872	neutral	1			Virgin America		F6x		0	@VirginAmerica	[40.64662464, -7	2015-02-22 16:0	San Francisco, (	Pacific Time (US & Canada)
131	5696432624592	neutral	0.6535		0	Virgin America		lawyang588		0	@VirginAmerica is flight 882 Cano		2015-02-22 15:41:51 -0800		
132	5696428455162	negative	1	Customer Servic	0.6448	Virgin America		melokudo		0	@VirginAmerica you are failing yo		2015-02-22 15:40:12 -0800		Eastern Time (US & Canada)
133	5696343183492	negative	1	Can't Tell	1	Virgin America		ChrysiChrysic		0	@VirginAmerica @FiDiFamilies u		2015-02-22 15:06:19 -0800		
134	5696336309783	neutral	1			Virgin America		nikkisixxfan93		0	@VirginAmerica has flight numbe		2015-02-22 15:0	Sacramento,Cali	Pacific Time (US & Canada)
135	5696332795460	negative	1	Cancelled Flight	0.6875	Virgin America		ChrysiChrysic		0	@VirginAmerica @ChrysiChrysic		2015-02-22 15:02:11 -0800		
136	5696300922734	negative	1	Late Flight	1	Virgin America		MOCBlogger		0	@VirginAmerica Another delayed		2015-02-22 14:4	San Diego	Alaska
137	5696274804257	negative	1	Customer Servic	1	Virgin America		tfaz		0	@VirginAmerica I need to registe		2015-02-22 14:3	Oakland, Califor	Pacific Time (US & Canada)
138	5696257392319	positive	1			Virgin America		lisaptv		1	@virginamerica you ROCK for ma		2015-02-22 14:32:14 -0800		Mountain Time (US & Canada)
139	5696256097997	neutral	1			Virgin America		dropapp		0	@VirginAmerica, @reallytallchris		2015-02-22 14:31:43 -0800		
140	5696201023893	positive	0.7011			Virgin America		HollywoodHotMom		0	@VirginAmerica always!!! Xoxo		2015-02-22 14:0	All Over!	Pacific Time (US & Canada)
141	5696195693728	negative	1	Flight Booking P	1	Virgin America		GoShar2012		0	@VirginAmerica why can't we boo		2015-02-22 14:0	Providence, RI	Eastern Time (US & Canada)