Executive Summary

Overview

This is an analysis of the 1.4 million cell phone reviews. I chose to explore reviews as I am a techno-savvy and personally I often review the phones that I buy or have used before. In fact, I have been rated 'helpful' on BestBuy which is a credential I am very proud of. I was curious to explore how analytics can help phone companies make decision-based on customer reviews. For this, I made use of text analytics.

Text analytics extract meaning from human language and have the power to offer businesses insights into large amounts of data such as the opinions of customers; insights which can be used to drive business decisions.

The goals of this exercise are:

- To assess the public perception of cell phones via exploratory data analysis and visualization
- To build a machine learning model which accurately predicts the sentiment of reviews

Description of the dataset

The dataset I am using for this project contains a total of 1.4 million user ratings and cell phone reviews for different brands of cell phones from different countries. The main column that is essential for the analysis is the review text itself and the score for the review which is crucial for sentiment analysis. It contains other information like the date review was posted, the language of the review text, country of the purchased phone, the source where the phone was purchased, and reviewer name that may not necessarily help us draw any conclusion. The review score ranges from 1-10 and there seem to be more positive reviews than negative reviews in the dataset.

Data Cleaning

Firstly, after adding all the CSV files to the data frame, I have removed all the NaN values that are useless for our analysis. Secondly, for this project, I have decided to use reviews that are in the English language only. Since our score column values range from 1-10, I have added a separate column named ‘Sentiment’. The sentiment labels are:

1 - 3: Negative
4 - 6: Neutral
7-10: Positive

Secondly, I have extracted the cell phone brand names and grouped them as a separate column named Phone. For example, there are multiple products from Samsung that are reviewed separately. I extracted the first word from the product column that would be Samsung and applied that to all product having Samsung as its first word. This way all products manufactured by Samsung will have name Samsung under the column 'Phone'.
In the column named country we only have the country code instead of the country name. Using 'pycountry' library from python, I have replaced the country column containing country code with its name and language column containing a specific language code name with its full name. Similarly using Pandas DateTime function I have replaced the date column containing the dates in US format to international program readable date format.

Descriptive Analysis

Basic Statistics

There are 1.33 million data in the dataset after removing all the information containing null values. The highest number of reviews were written in English i.e 545424 while Russian is at second with the number of 184025 and Chinese lies at last with only 10 reviews written in Chinese. Since I am only working with reviews written in English for our project, I will have to deal with 545424 number of data. The mean value of score for English reviews is 7.68, while the median and standard deviation are 8.8 and 2.88 respectively. This shows how our data contains more positive reviews than negative. Similarly, the highest number of reviews were given in the year 2016 which is 108377 reviews and 672 reviews were given in the year 2000 which is the least in the dataset. The number of reviewers from the US is 312110, 127873 from India whereas only 10 reviews were from Singapore. The number of sentiment values are as follows:-

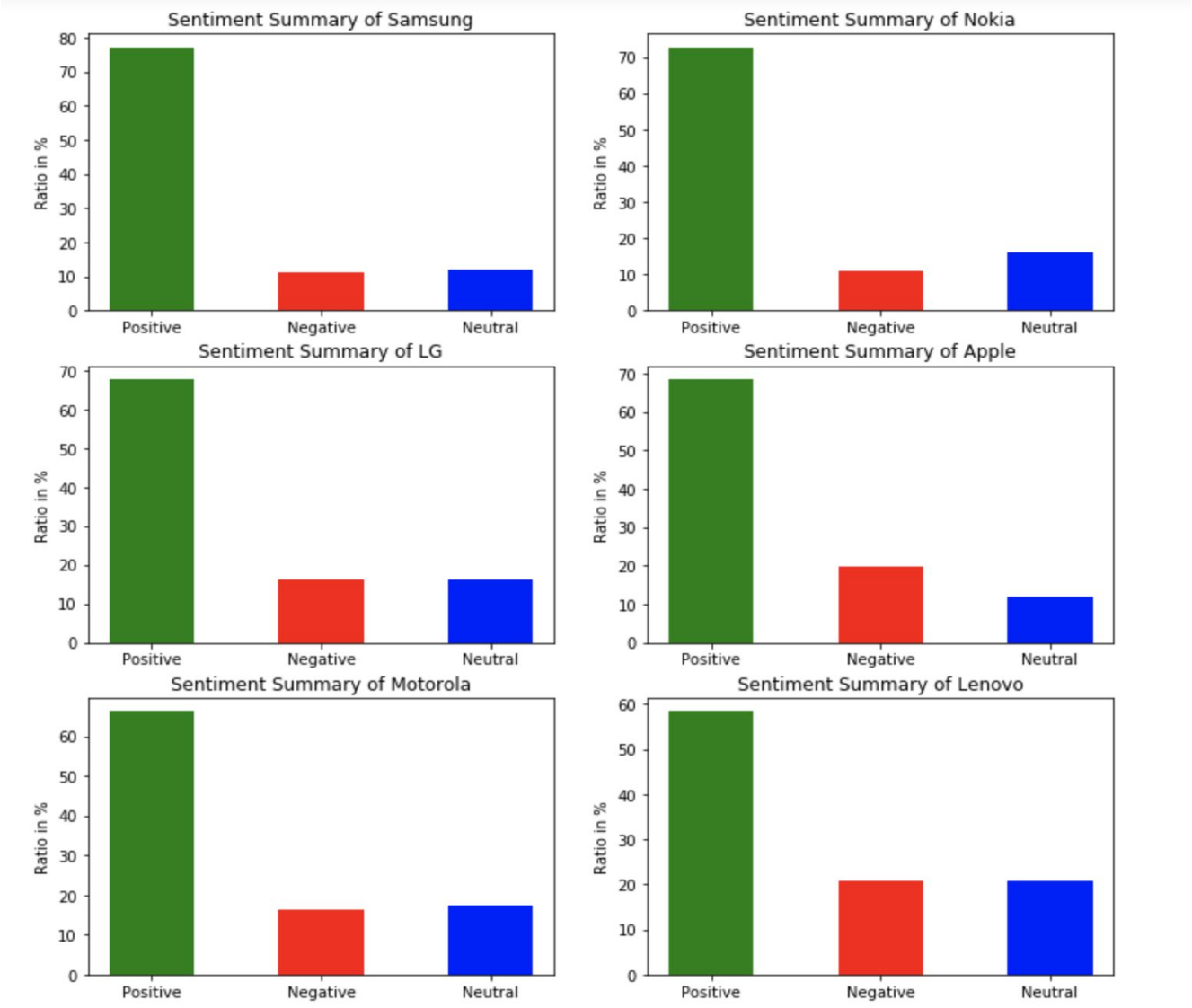
Positive 385578
Neutral 81635
Negative 78211

The number of unique phones reviewed is 1371 among which Samsung is at the top with 134116 reviews. Lastly, some of the popular sources where the reviews are written are Amazon, Samsung, and Phone Arena.

Graphs

For the purpose of this analysis, the data will be manipulated in such that plotting the data will allow for easier data visualization.

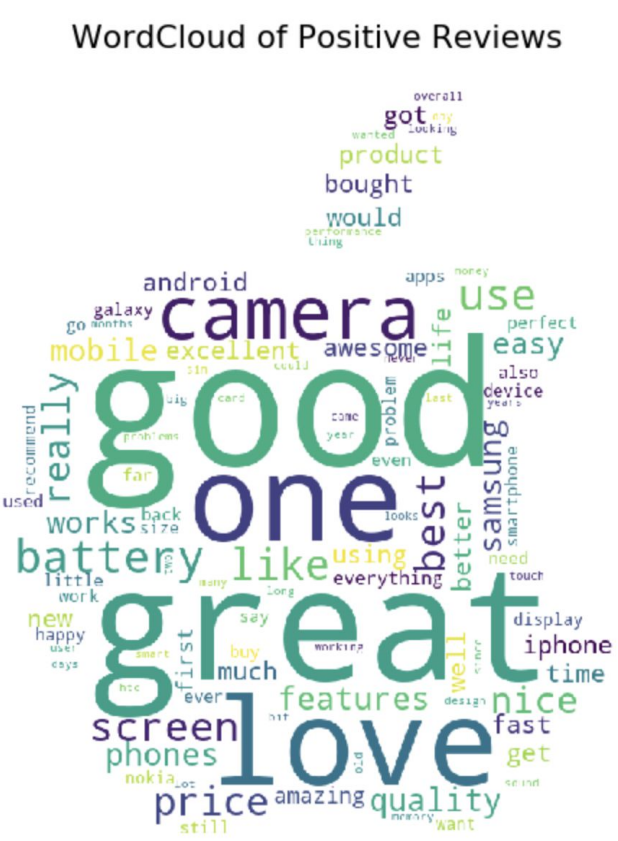
1. Bar Graph comparing the sentiment percentage of top 6 sold cell phones



This above bar graph compares the sentiment percentage of top 6 cell phones. I calculated the percentage by calculating the ratio of a particular sentiment over the total review for that specific phone. First of all, the bar graph shows that for all 6 cell phone brand the percentage of positive reviews are way higher than the negative or neutral reviews. This may be because of the fact that we have more positive reviews than other reviews in the first place. But this also correlates with the fact that most popular phone brands like Samsung and Apple are most likely preferable. Samsung has the highest positive review percentage that is around 78% while Lenovo has the lowest with around 57%. This may be because of the fact that Samsung actually produces both

midrange and higher-end devices targeting both rich and average income countries. Since the second highest number of reviewers are from India after the US, this shows how Samsung is able to conduct a successful business in both countries and hence has the highest positive review percentage. The fact that Nokia is the second most positive rated has to do with the fact that this data was actually taken from the year 2002-2017 when Nokia was at its prime stage and the company was doing a great job with their hardware performance worldwide. However, the brand like Apple and LG is more inclined towards US customers.

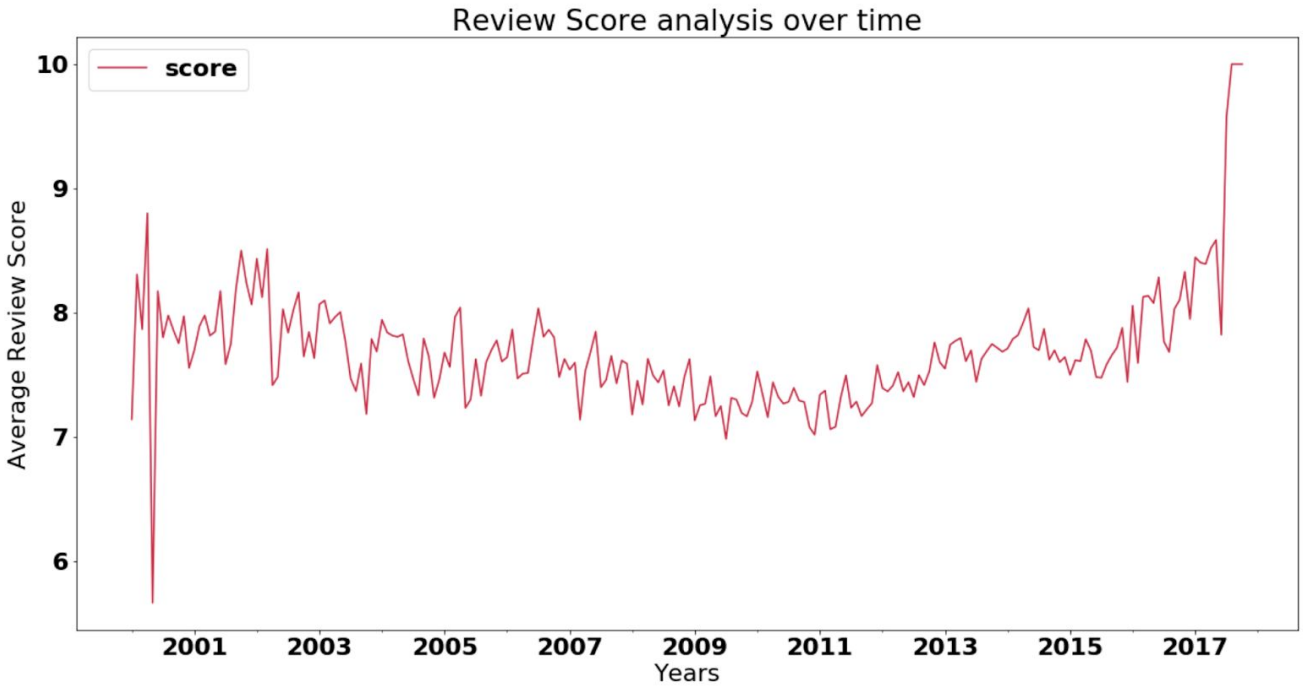
2. Word Cloud of both Positive and Negative reviews



I extracted the most used positive and negative words using countvectorizer that provides a simple way to both tokenize a collection of text documents and build a vocabulary of common words, but also to encode new documents using that vocabulary. I removed the stopwords adding my own custom stopwords that are relative in terms of cell phones to the English stopwords library imported from nltk corpus from Scikit learn. The reviews word cloud highlights words like good, great, love, battery, camera, Android, screen, quality, price and so on. This clearly

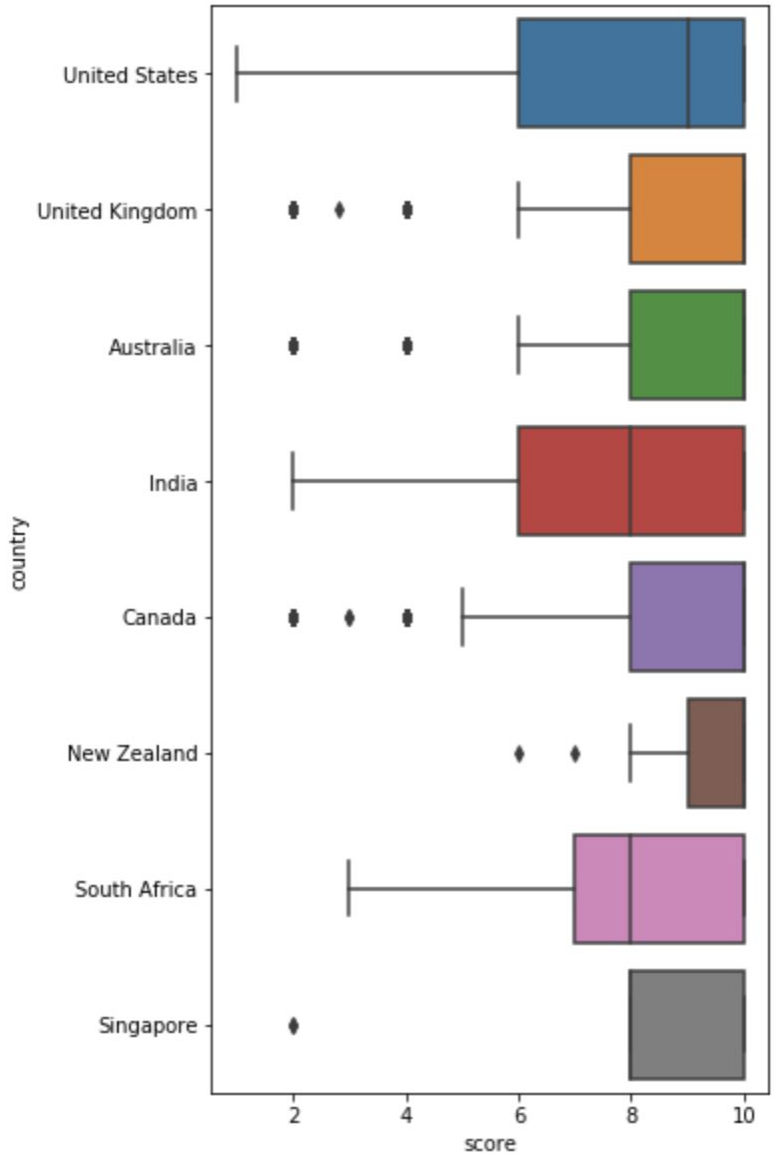
describes features that the reviewer liked about the phone like a good camera, amazing screen, good battery life, and a reasonable price.
Surprisingly the negative reviews word cloud also has words like charge, sim, touch,money, battery, price, camera and so on. This shows us that the features like camera, screen, battery, and price play a crucial role to determine a phone review. This will actually help phone companies improve their phone reviews by first improving the very features like camera, screen, and battery.

3. Line Graph of Time Vs Average Score on monthly basis



This line graph may seem biased because it's y-axis value ranges from 6-10 that is only positive reviews. It actually has to do with the fact that there were way more positive reviews in the dataset than the negative reviews. This is a time series line graph that highlights the changing trend of reviews along with the time. Surprisingly our dataset started with almost all negative reviews because in 2001 the line graph has dropped dramatically and it has remained constant towards the positive reviews ever since with the highest positive reviews given recently around the end of 2017.

4. BoxPlot comparing the review score pattern from different countries



This is a boxplot describing the median review score as well as the outlier for the review score. Almost all of the countries score range lies in the positive sentiment score range. Again this has to do with the fact of our dataset containing more positive reviews than negative or neutral. The United States has the highest median value that lies around the score of 9. The UK, Australia,

and Canada have multiple outliers lying outside the range of positive score. That means people in these countries either strongly prefer the phone or they don't like it at all.

Sentiment Analysis/Machine Learning

For the machine learning model, I analyzed the sentiment of reviews using TF-IDF Vectorizer and Linear SVM for the classifier. Finally, I calculated the accuracy score by testing my sentiment classifier against the test data. Overall, I came up with an accuracy score of **94.7%**.

Pre-Processing

First of all, I have created my label and stored it in a list called target. I haven't used any neutral reviews for this analysis. So my label would be 0 if the review is negative (score from 1-3) and 1 if the review is positive (score from 7-10). Furthermore, I have pre-processed the reviews to first change all text to lowercase and then removed spaces and punctuations since they don't have any sentiment at all.

Vectorizer

Vectorization simply is a process involving converting each review into a certain numeric value so that it will make sense for the machine to learn. For my case, I have found TF-IDF vectorizer from sci-kit learn more effective over Countvectorizer. The CountVectorizer provides a simple way to both tokenize a collection of text documents and build a vocabulary of known words. An encoded vector is returned with a length of the entire vocabulary and an integer count for the number of times each word appeared in the text. In simple words, it counts the occurrence of a word in a document and creates a bag of words model.

Why TF-IDF?

Word counts are a good starting point but are very basic. One issue I found with simple counts is that even after removing stop words, some words like "phone" will appear many times and their large counts will not be very meaningful in the encoded vectors. An alternative is to calculate word frequencies, and by far the most popular method is called TF-IDF. This is an acronym that stands for "Term Frequency-Inverse Document" Frequency which are the components of the resulting scores assigned to each word. It converts the text reviews to a matrix of "tfidf" features that are the method to convert the textual information into the vector space. They are a measure of how important a word in a text is. The TfidfVectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new text documents.

Significance of n-gram range

Changing the default value of n-gram range yielded more accuracy in my case. I set the n-gram range to (1,2) that will add both two consecutive words and single word as an input. It's about

treating ngrams additionally. Let me illustrate with a simple example. "very good" is a 2-gram that is considered as an extra feature separately from "very" and "good" when you have an n-gram range of (1,2).

Significance of Custom StopWords

The stop words that I import from the library 'english' were not enough or relevant in the case of cell phone reviews. So, I created my own list of stop words and appended it to the stopwords list imported from 'NLTK' library by using an algorithm to find most repeating words from the text reviews which didn't have any sentiments.

Classifier

First, using 'sklearn model_selection' library I have divided the train and test data as 75% and 25% respectively before doing any classification. Support Vector Machine can be very much effective in sentiment clarification as they are treated as classifiers with high accuracy. So, I have used Linear SVC as my classifier because I found out that its training time was also significantly better than other linear classifiers.

Significance of C value

The C parameter tells the SVM optimization how much you want to avoid misclassifying each training example. The value of c that gave me the highest accuracy was 0.79.

Future Potential Analysis

I think I did a pretty solid job coming up with an accuracy of **94.7%** for my sentiment classifier. If I get more time to work with this dataset, my future goal would be to use the trained sentiment model to predict the sentiment of a different dataset, such as tips or recommendation data (which are short reviews and advice that do not have a star rating). I can then go on to create a recommender system that will allow phone companies to target a user based on their reviews for the phone they previously bought. Additionally, I will drill down into a particular business (which will be a smartphone company) and analyze the sentiment over time. It would be extremely interesting and highly useful if the model could tell us what caused say a drop in sentiment (e.g. if in a given period the sentiment has dropped, and in that same period the reviews are complaining about the quality of the phone, this could give the phone company actionable insights).

References

- https://towardsdatascience.com/another-twitter-sentiment-analysis-with-python-part-5-50b4e87d9bdd
- https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html

- https://towardsdatascience.com/sentiment-analysis-with-python-part-1-5ce197074184