Extraction, Transformation, and Load Technical Report

Amazon Cell Phone Reviews

TEAM WONDERFUL

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Summary & Scope

We gathered our wonderful data from Kaggle. Our dataset focuses on both unlocked and locked carriers, and scoped on ten brands: ASUS, Apple, Google, HUAWEI, Motorola, Nokia, OnePlus, Samsung, Sony, and Xiaomi.

Reviews play a big role when recieving client feedback on a particular product or service one offers to the public. In this project, we want to determine whether Apple phones or Androids have better ratings. Not only including the numerical value of the rating but also the textual value. In order to find the textual value of a review, we had to delve deep into a Statistical Sentiment Analysis.

Our data was a pair of CSV files that contained information on phone reviews. One set contained the list of phone reviews and had a length of about 7500 records. The other set of data contained a list of phone models on which the individual reviews were captured.

Sentiment Analysis consists of two parts: Subjectivity and Polarity. Polarity simply means emotions expressed in a sentence across a range of negative to positive. Subjectivity means that a subjective sentence expresses some personal feelings, views, or beliefs. This will help us to find the textual value. We also want to find the most common words from the surveys to convey a pattern in the data.

Our initial bar graph shows that most ratings are five-star reviews, while the one-star reviews are a far second behind five-star reviews. Based on the fact that more people like what they are buying versus disliking what they are buying, we expect that our sentiment analysis will show that the results are fairly positive.

1.2 Technologies and resource contributions

This section lists out the team members and their contributions towards the ETL initiative. Use this section to also outline (or list) the tech stack used to obtain the final outcome.

Kevin Pate was responsible for creating the code for our statistical sentiment analysis. Bowen Ya was responsible for cleaning the data. Carl Burks helped to create the code for our statistical sentiment analysis. Keri Broughton was responsible for writing our report and pulling all of the information together. Terrance Atkinson was responsible for creating our PowerPoint presentation.

We used numpy, pandas, time, re, nltk, nltk.corpus, nltk.stem, wordcloud, textblob, seaborn, numpy.random, scipy.stats, collections, and matplotlib.pyplot.

1.3 Definitions, Acronyms and Abbreviations

Below is a list of new softwares, techniques, and modules that we used including their definitions and explanations:

Time is a defined module in Python which allows us to handle various operations regarding time, its conversions and representations, which its use in various applications in life.

Re stands for regular expression which specifies a set of strings that matches it; the functions in this module let you check if a particular string matches a given regular expression.

NLTK stands for natural language processing. It is one of the leading platforms for working with human language data and Python. We used it to help with applying in statistical

NLTK.corpus is a massive dump of all kinds of natural language data sets. We used this to dump our data into.

NLTK.stem basically means that NLTK can find the stems of words. This is useful for scanning our ratings to find relevant information. A word stem is part of a word. It is spot of a normalization idea, but linguistic. For example, the stem of the word waiting is wait.

Wordcloud is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. Again, we used this technique to help with reading the reviews. It was very useful as you can see later in the rest of the report.

Textblob is a library for processing textual data. It provides simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more. We used this to help process through the written reviews by customers. This worked alongside wordcloud and NLTK.stem to be specific.

Seaborn is a Python data visualization library based on matplotlib. Matplotlib is another data visualization tool that we used. Seaborn provides a high-level interface for drawing attractive and informative statistical graphics.

Numpy.random creates an array of specified shape and fills it with random values. It basically stands for random sampling. This helped with our statistical analysis becuase it helps to return an array of specified shapes and fills it with random integers from low (inclusive) to high (exclusive), i.e in the interval [low, high].

Scipy stats is a statistical function. This module contains a large number of probability distribution as well as a growing library of statistical functions. We used this to help with our statistical sentiment analysis.

Collections in Python are containers that are used to store collections of data, for example, list, dict, set, tuple, etc. These are built-in collections. A counter is a dict subclass for counting hashable objects.

2. ETL DETAILS

2.1 Data Import/Extract Sources and Method

We got our data set from Kaggle. We first imported the CSV files into SQL. After running into some issues, we cleaned the data through Python. We used a Python data frame. We dropped rows that were not needed. We also deleted all of the data that was not used before 2016. The transformed csv file was then loaded into Jupyter for the Sentiment Analysis.

2.2 Data Acquisition

We used two CSV files. One was an items file and the other was the reviews. We used asin of the product as the primary key to merge with the reviews, and then dropped irrelevant columns such as urls and image urls. We wanted to have a smaller and more focused dataframe so we only kept ratings and reviews since 2016.

2.3 Data Transform & Data Integrity

We dropped rows that were not needed. We also deleted all of the data that was not used before 2016. The transformed csv file was then loaded into Jupyter for the Sentiment Analysis. During our data transformation we had to change the "body" column to a string. We lowercased all of our reviews. This helps in the process of normalization which is an important step to keep the words in a uniform manner. We also removed punctuation and special characters. We also removed stop-words . These are words such as I, me, they, etc. which have no predictive power. Lastly, we removed all stem suffixes like -ing, ses, etc.

2.4 Data Refresh Frequency

The data will be refreshed monthly.

2.5 Data Security

We did not confront any data security issues throughout our sentiment analysis process.

2.6 Data Loading and Availability

We uploaded the cleaned data to Postgresql and it will be available to anyone who would like to do further analysis in the future.

3. DATA QUALITY & CONCLUSION

The data met our hypothesis, which was that reviews would be better than nuetral. This was proven by the polarity and subjectivity scores. Polarity shows most reviews are between 0 and .5 (.24 is the mean) while subjectivity falls between .234 and .6 (.49 is the mean). In our box plot, this basically showed that the reviews are fairly positive.

We then calculated the covariance and correlation. Covariance between the two variables is 0.13170629. This positive numbers suggests the variable change in the same direction as we expect. The correlation is 0.8759 which suggests a high level of correlation (any value above 0.5 and close to 1.0)

We also found that the most frequently used words are good, great phone , love, excel, great, love phone, perfect, good phone, excellent, and work great. Good was the most frequently used word. Our data was highly correlated (0.8759), which means our findings and the data that provided our findings are strong.