Alert! Analytics

Aswini Muthukumar

smartphone Sentiment analysis

Contents

[Project Background 2](#_Toc103113554)

[Data Collection 2](#_Toc103113555)

[Findings 3](#_Toc103113556)

[Distribution 3](#_Toc103113557)

[Null Values 3](#_Toc103113558)

[Pre-processing and Sentiment Mapping 3](#_Toc103113559)

[Features Selection Methods 4](#_Toc103113560)

[Near Zero Variance 4](#_Toc103113561)

[Recursive Feature Elimination 4](#_Toc103113562)

[Machine Learning Models and Comparison of Methods on Original Dataset 5](#_Toc103113563)

[C5.0 Model on Features Selected Datasets 6](#_Toc103113564)

[Feature Engineering 7](#_Toc103113565)

[Confidence of Improved Predictive Power and Efficiency 8](#_Toc103113566)

[Predictions on Large Matrix From Web Crawl 9](#_Toc103113567)

[Lessons Learned 10](#_Toc103113568)

# Project Background

Helio is working with a government health agency to create a suite of smart phone medical apps for use by aid workers in developing countries. This suite of apps will enable the aid workers to manage local health conditions. The government agency requires that the app suite be bundled with one model of smart phone. One of the purposes of this project is to help Helio narrow their list down to one device, by examining the prevalence of positive and negative attitudes towards devices on the web. The goal of this project is to provide Helio with a report that contains an analysis of sentiments towards the target devices, as well as a description of the methods and processes we used to arrive at our conclusions.

The role of Alert! Analytics in this project is to **conduct a broad-based web sentiment analysis on the two top smartphone models - iPhone and Samsung Galaxy and produce findings that describes user attitudes towards these smartphones.**

# Data Collection

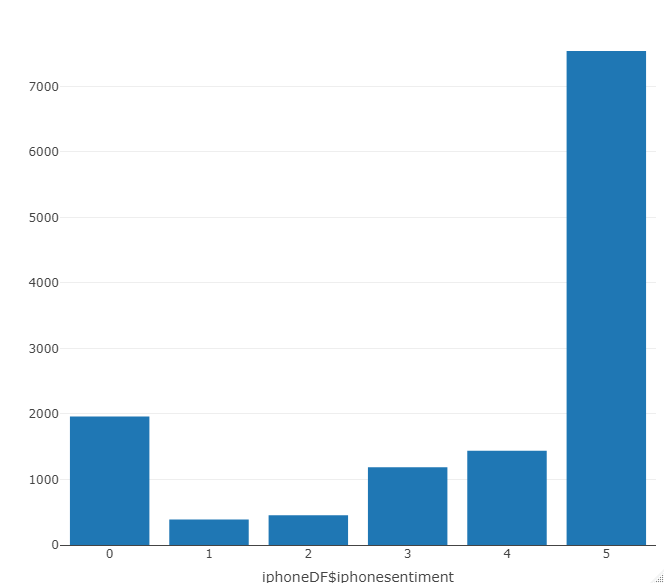
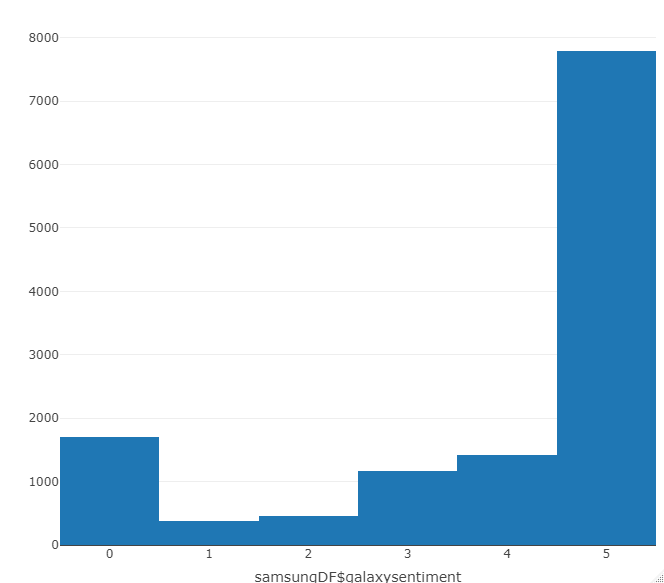
Although there are a number of ways to capture sentiment from text documents, our general approach to this project is to count words associated with sentiment towards the target devices within relevant documents on the web. We then leverage this data and machine learning methods to look for patterns in the documents that enable us to label each of these documents with a value that represents the level of positive or negative sentiment toward each of these devices. We then analyze and compare the frequency and distribution of the sentiments for each of these devices.

# Findings

## Distribution

On initial exploration of the small matrices, which are samples of sentiment captures from the web, it was found that there may have been good chances of error in data, data collection and/or manual labeling. The small matrices were created for initial analysis and experimentation with different machine learning models, out of which the optimal one was selected. This one model was then applied to the large matrices which are massive datasets of sentiment captures from web crawl.

The below plot shows that both iphonesentiment and galaxysentiment (target variables for analysis, prediction and outcome) from their respective similar small matrices are highly right skewed. Both the devices seem to have a highly positive sentiment, which may or may not be true. It would help to have equally distributed data for an optimal analysis.

Distribution graphs for the sample small matrices containing sentiment categories and levels

## Null Values

There were no missing or null values in both the small matrices.

## Pre-processing and Sentiment Mapping

The small matrices were analyzed and appropriate changes were made. Both iphonesentiment and galaxysentiment were set to factor level data types since this is a classification problem.

Also, mapping of the sentiment levels were inferred as below upon observation of distribution.

0: Sentiment Unclear  
1: very negative  
2: somewhat negative  
3: neutral  
4: somewhat positive  
5: very positive

# **Features Selection Methods**

# 

**The below methods were tried to create respective datasets for analysis, out of which the best dataset was selected for application with the optimal machine learning model to predict sentiment.**

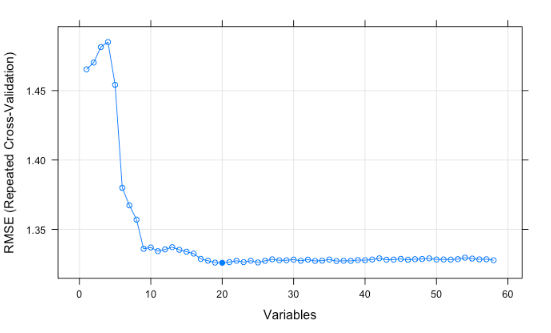
## **Near Zero Variance**

Zero or near zero variance variables indicate identical or close to identical predictor values across all the samples. These predictor variables are not really informative and they could also break some models. So this method was applied to remove all the near zero variance variables and the results were saved in a separate data frame for further analysis.

## **Recursive Feature Elimination**

Recursive Feature Elimination (RFE) builds models to implement backwards selection of predictors based on predictor importance ranking. The least important predictors will be removed and the model will be re-built. The process happens recursively until the optimal subset of predictors are found. This subset can be used to produce an accurate model.

The above method was applied and the most important features were noted. It was found that an efficient model can be built with 20 features.



Repeated Features Elimination (RFE) graph to find the most important features

The top 5 variables in the iPhone small matrix for prediction of iphonesentiment were found to be:

1. iPhone
2. Samsunggalaxy
3. iphonecampos
4. iphonedispos
5. iphoneperpos

# Machine **Learning Models and Comparison of Methods on Original Dataset**

For this classification problem, the below four models were tried on the original dataset and their performance metrics were compared.

1. Decision Tree Classifier C5.0
2. Random Forest Classifier (RF)
3. Support Vector Machine (SVM)
4. K-Nearest Neighbors (KKNN)

The below chart shows the two performance metrics, accuracy and kappa by the above models.

**Although C5.0 and RF models had similar metric results, RF model took a long time to build and crashed a few times. So, C5.0 model was found to be the best of the above models not only for its performance but also for its efficient computational time.**

# C5.0 Model on Features Selected Datasets

The C5.0 machine learning model to predict smartphone sentiment from the original data set was fit to the other datasets iphoneNZV and iphoneRFE prepared earlier using different features selection methods. Their performance metrics were evaluated. The ultimate goal of experimenting with the machine learning model on different datasets is to find the model which achieves the highest accuracy in predicting sentiment while being efficient at the same time. This model will further be used to apply on the massive sentiment captures dataset (features selection methods applied if needed) from the web crawl (largematrix), where actual predictions are to be done.

Accuracy and Kappa metrics were not found to have improved in the processed datasets and hence these methods and datasets were dropped and not considered further.

# Feature Engineering

Feature engineering is the art of working with the data so that it can more readily be consumed by machine learning algorithms. Feature engineering includes mutating existing attributes, combining attributes, deconstructing attributes and much more.

Previously, it was noticed that the dependent variable's (sentiment) factor levels had very poor sensitivity and balanced accuracy. It was experimented to see if combining some of the sentiment factor levels will help increase accuracy and kappa. The sentiment levels were remapped as below:

1. sentiment unclear
2. negative
3. neutral
4. positive

The earlier built C5.0 model was fit to the recoded sentiments dataset.

# Confidence of Improved Predictive Power and Efficiency

Accuracy of the C5.0 classifier model on the recoded dataset was found to have increased considerably from 77% to 85% and Kappa from 55% to 62%. Hence, the goal to achieve the model with top predictive power and efficiency was attained through this method.

# Predictions on Large Matrix From Web Crawl

The above found optimal C5.0 model from feature engineered dataset was fit to the large matrix file prepared by web scraping the Common Crawl site.

As seen from the above charts, there were huge counts of unclear sentiment predictions on both iPhone and Samsunggalaxy. The predictions on both cases are left skewed with most number of unclear sentiments. Perhaps, distribution of data in the large matrix may have not been even. And also, the model from unevenly distributed small matrices were used for prediction on the large matrix.

An evenly distributed training dataset would help in preparing a more accurate machine learning model for predictions.

# Conclusion

The distributions of predicted sentiments towards both the phones were quite similar, with the negative sentiment being overwhelming for both phones. Most people are known to write a strong negative review than for a positive sentiment. And, sentiments of common people towards smartphones are ever changing. Perhaps, data **collection on sentiment captures for analysis should be considered at organizational level from the employees.**

But, the predicted high negative sentiment kept aside, both iphone and SamsungGalaxy have around 5k positive reviews compared to negative and neutral reviews. So, **the implication is that though both phones have similar positive reviews, it would be a great idea to consider SamsungGalaxy for this project because of its ease of use and cost.**

Also, because of its sophisticated operating system and phone system, it would be difficult for Samsung users to adapt to iphone, whereas the opposite is easy.

# Lessons Learned

One of the most important lessons in this project was to set up compute clusters. Setting up compute clusters made the project work a little less hard.

The next lesson learned was to use CARET package to do the heavy lifting. Training models and making predictions are done seamlessly by itself without having to code line by line extensively.

Understanding and experimenting with feature selection and feature engineering was a vast and time consuming process. I figured that I need to learn and dig deeper on how to go about this in a smart way.

Check mapping of levels in classification problems and removed irrelevant and redundant levels to avoid confusion.

THANKYOU