NATURAL LANGUAGE PROCESSING TWITTER SENTIMENT ANALYSIS

DATA SONICS

The Team

Emmanuel Kipkorir
Data Scientist

Medrine Waeni
Principal Data Lead

Antony Brian
DevOps Engineer

Samson Kamunyu
ML Engineer

Betty Bett
Data Administrator

John Mungai
Product Manager

Overview

In the recent past, Twitter has gained traction among consumers as a place where they can express themselves towards goods and services. With this in mind, brands such as Google and Apple have a keen interest in tweet engagements as they use it as a medium to address the challenges facing their clients.

Problem Statement

There is need for companies such as Google and Apple to filter through the millions of tweets streaming in to draw insights on their customer satisfaction.

Objectives

- To accurately classify the polarity of tweets
- To identify the relationship between tweet sentiments and brands.
- To identify how brands are associated with emotions.

Data Understanding

- The data was outsourced from data.world.
- The data contains 36574 data points and 3 columns.
- The dataset was published to the public on 30th August 2013.

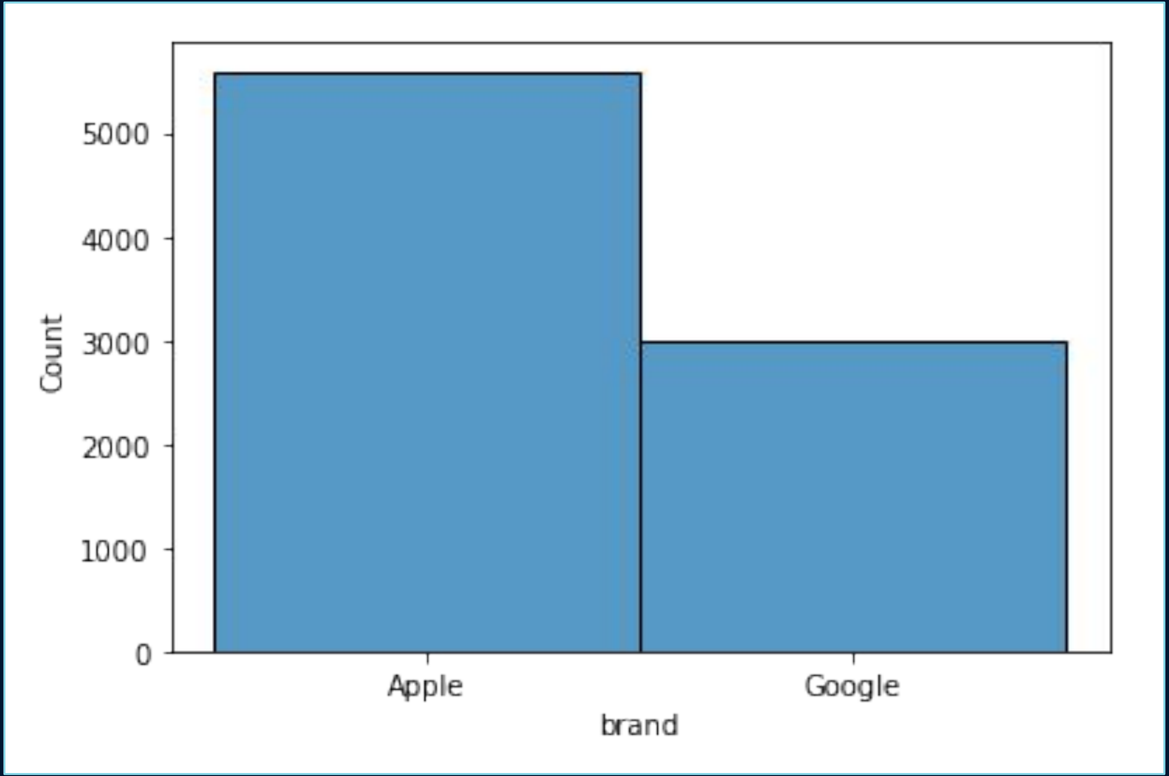
Data Preparation

- Renaming the columns
- Removing missing values from tweet text
- Handle the duplicated data by dropping them
- Remove capitalization, punctuation and stop words
- Fill the missing values of sentiments with their appropriate values

Exploratory Data Analysis

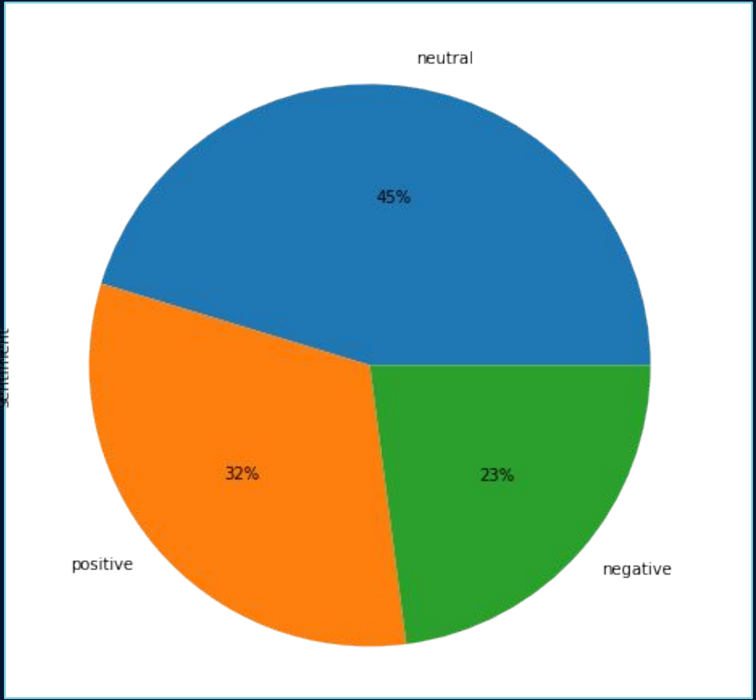
01 | Univariate Analysis

A Graph of Brands Distribution



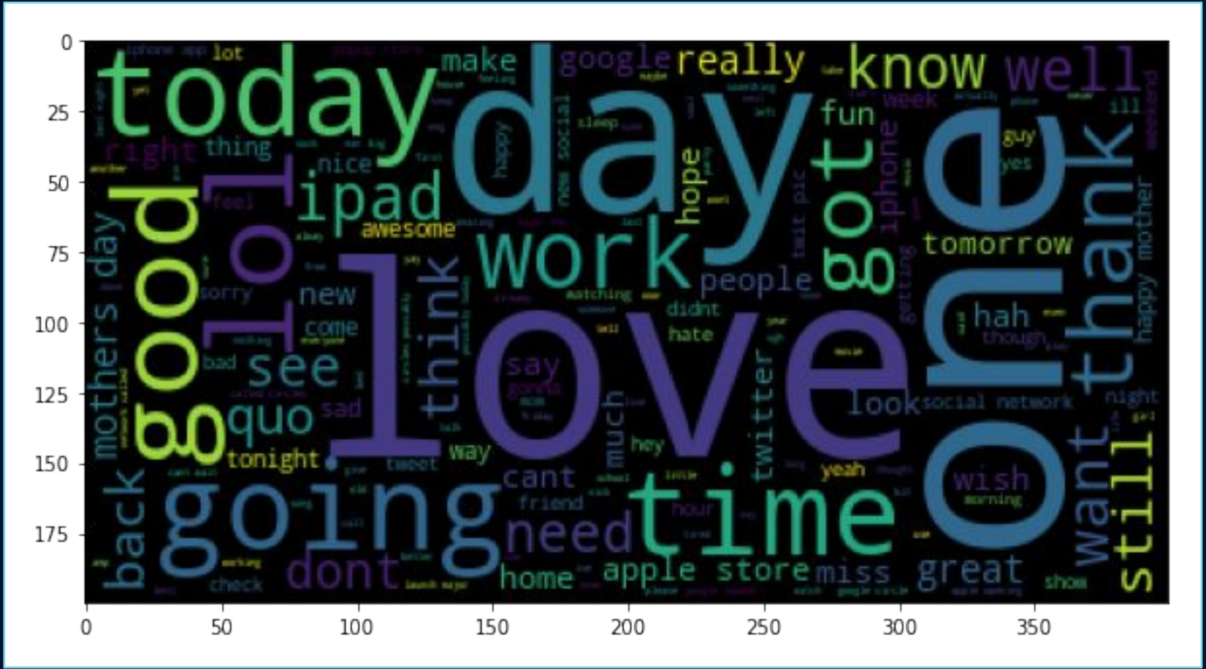
Apple has a higher product count than Google

Distribution of Tweet Sentiments



Neutral has the highest tweet sentiments at 45% followed by Positive at 32% and Negative at 23%

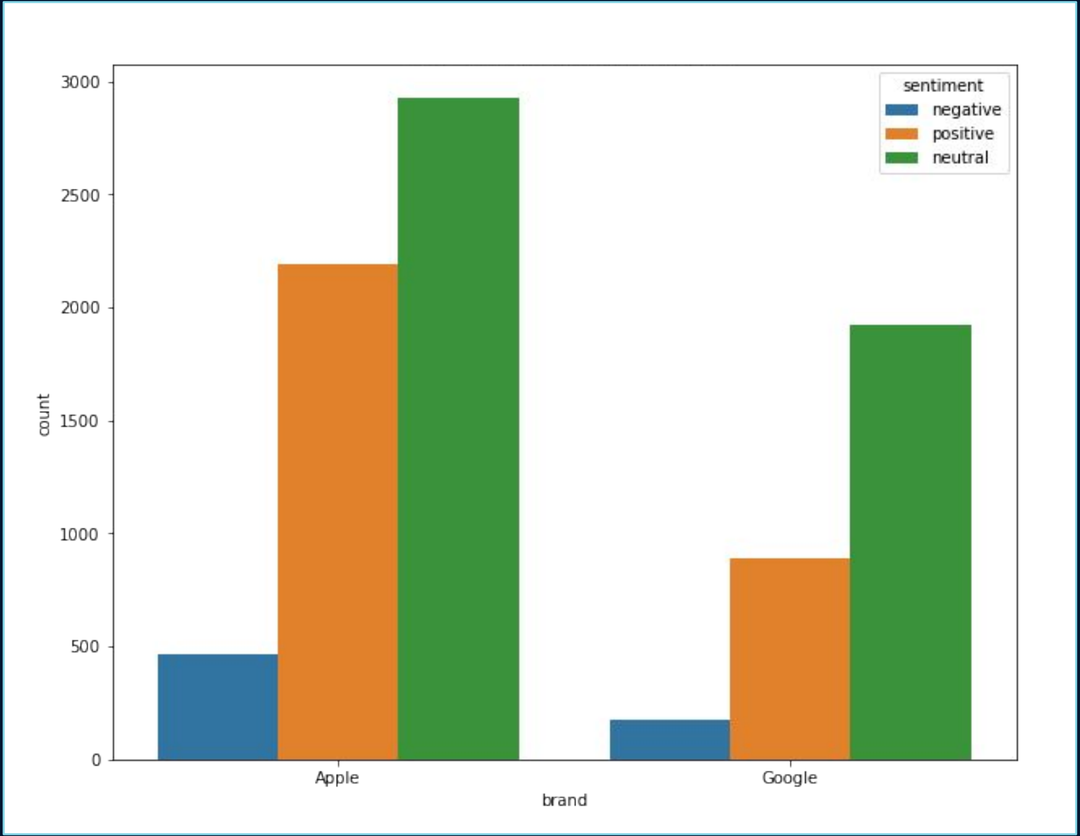
A Distribution of Word Count in Tweets



The words love, one and day are the most used words in tweet sentiments

02 Bivariate Analysis

Tweets Distribution Across Brands



In both Apple and Google neutral sentiments scored highest while negative sentiments the least

Models Performance

	Basemodel	XGBoost	GradientBoost	AdaBoost
Train F1 score	0.760215	0.922111	0.780253	0.663089
Test F1 score	0.622554	0.686923	0.681812	0.646121
Train accuracy	0.760406	0.922259	0.782920	0.670823
Test accuracy	0.623438	0.688646	0.686457	0.654264

Findings

- XGBoost and Gradient-Boosting models performed very well but were overfitting.
- AdaBoost model generalized well on both training and test data and chosen as final model for deployment.

Recommendations

- More balances data is needed to increase the accuracy of the model.
- The model accuracy can also be improved by using a neural network.

THANK YOU!

Do you have any questions/comments?

datasonics@dsbi.ac.ke

+254(0) 715 036 182

datasonics.com