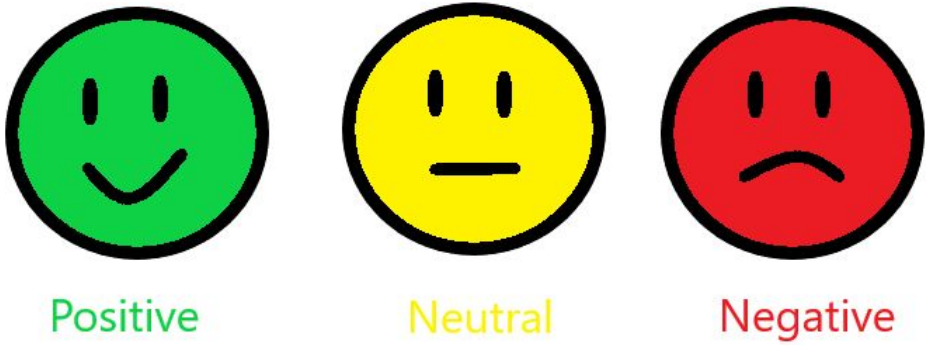
Classifying Positive and Negative Tweets

Flatiron Phase 4 Project
By: Rachael Bryant
Presented February 13th

Business Problem

- Analyses the text of tweets in order to predict what tweets are positive or negative for better marketing purposes for Twitter.

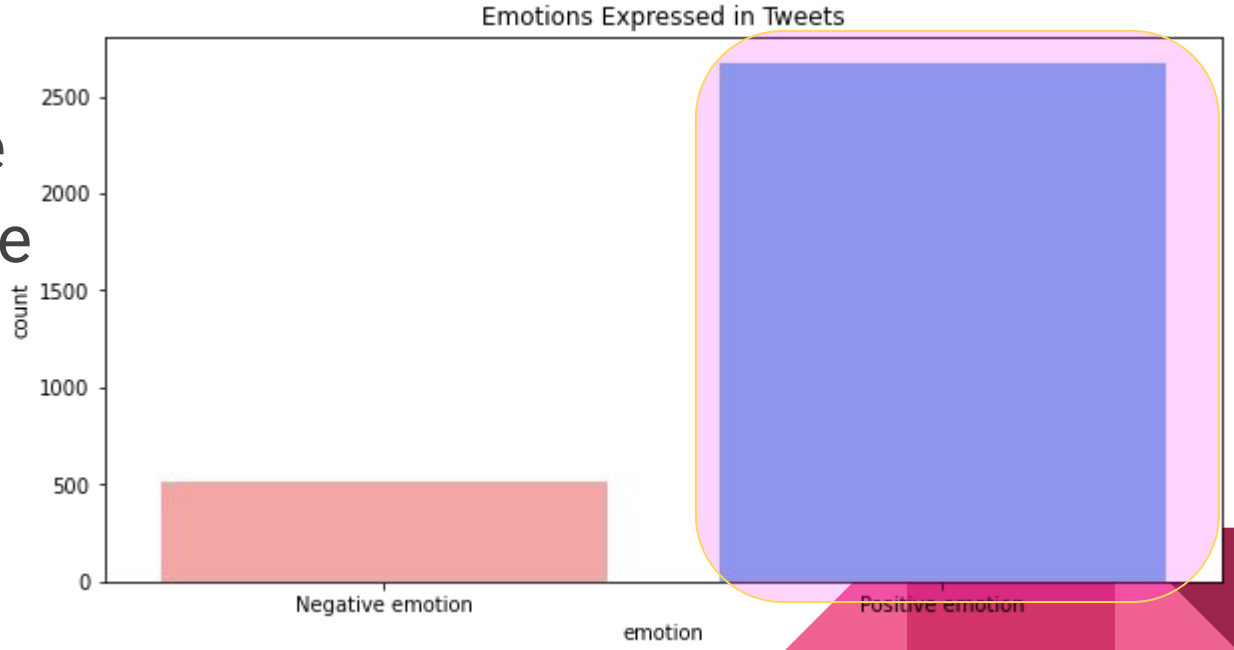
Data



- The tweet data is from CrowdFlower via data.world.
- About 9,000 records
  - Including emotion expressed in the tweet, the tweet itself and the device that the emotion is directed towards.

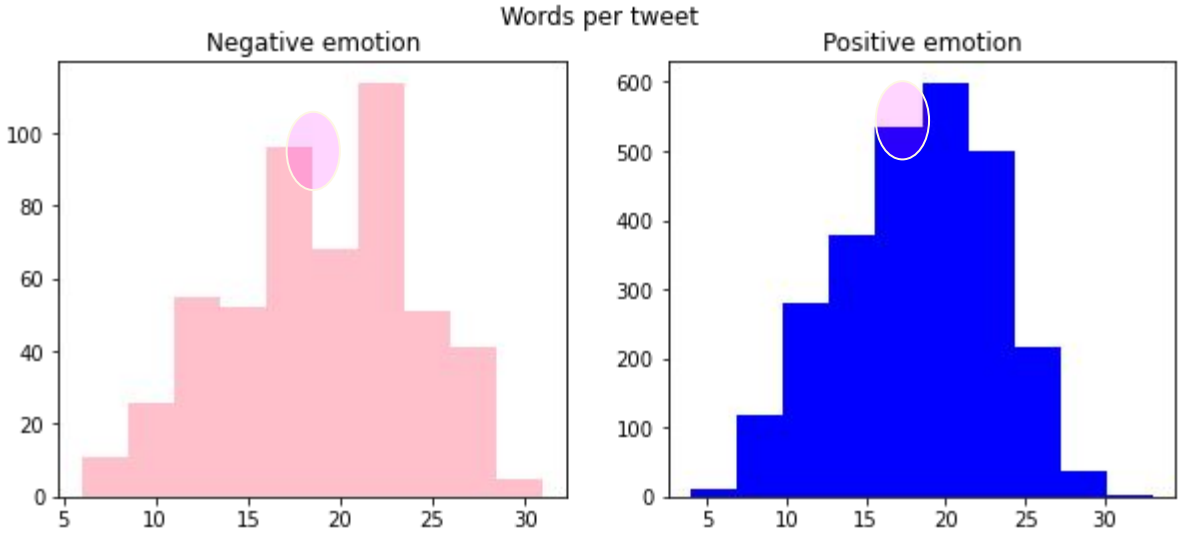
Focusing on only Positive and Negative

- 83% Positive
- 16% Negative



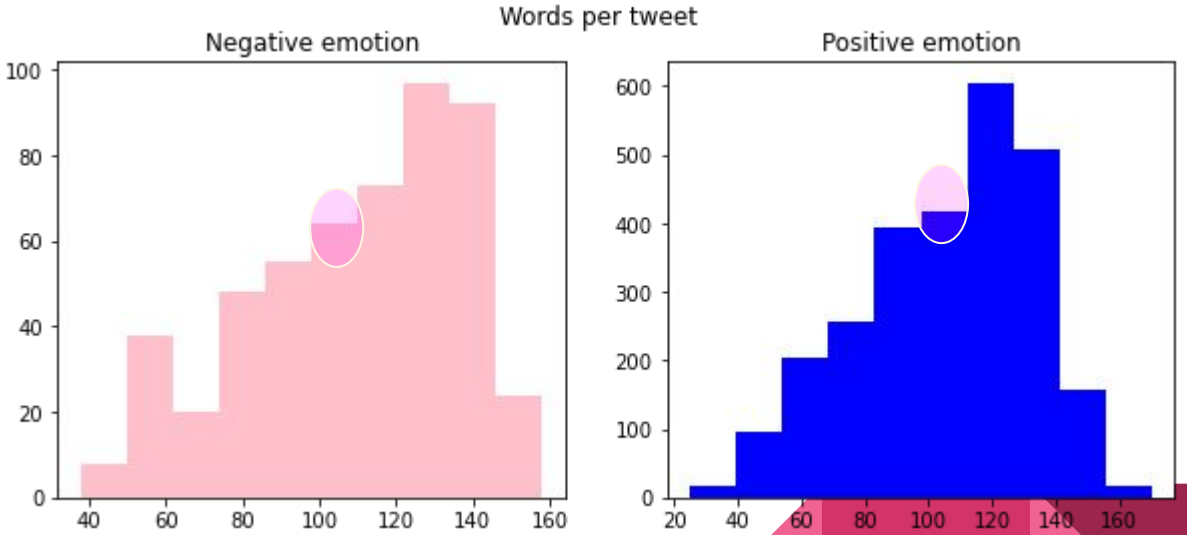
Words per Tweet

- Same average of words per tweet at 18.

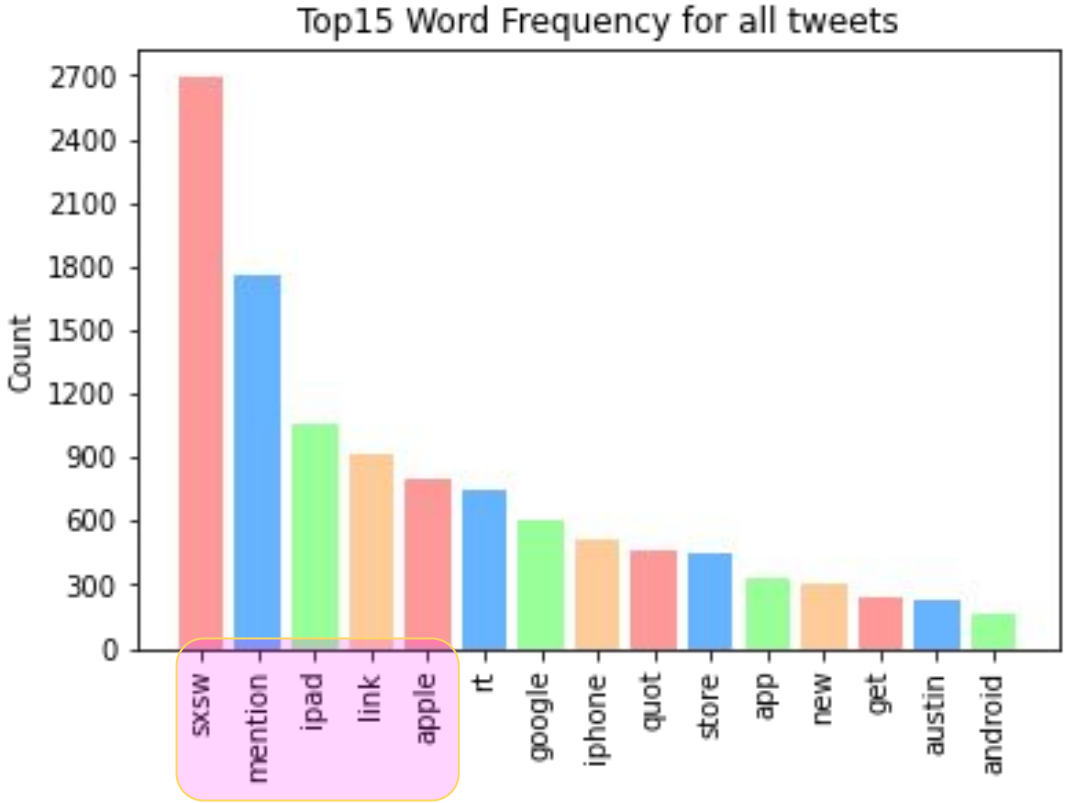


Review Length per Tweet

- Similar review length per tweet. 108 per negative and 105 per positive.

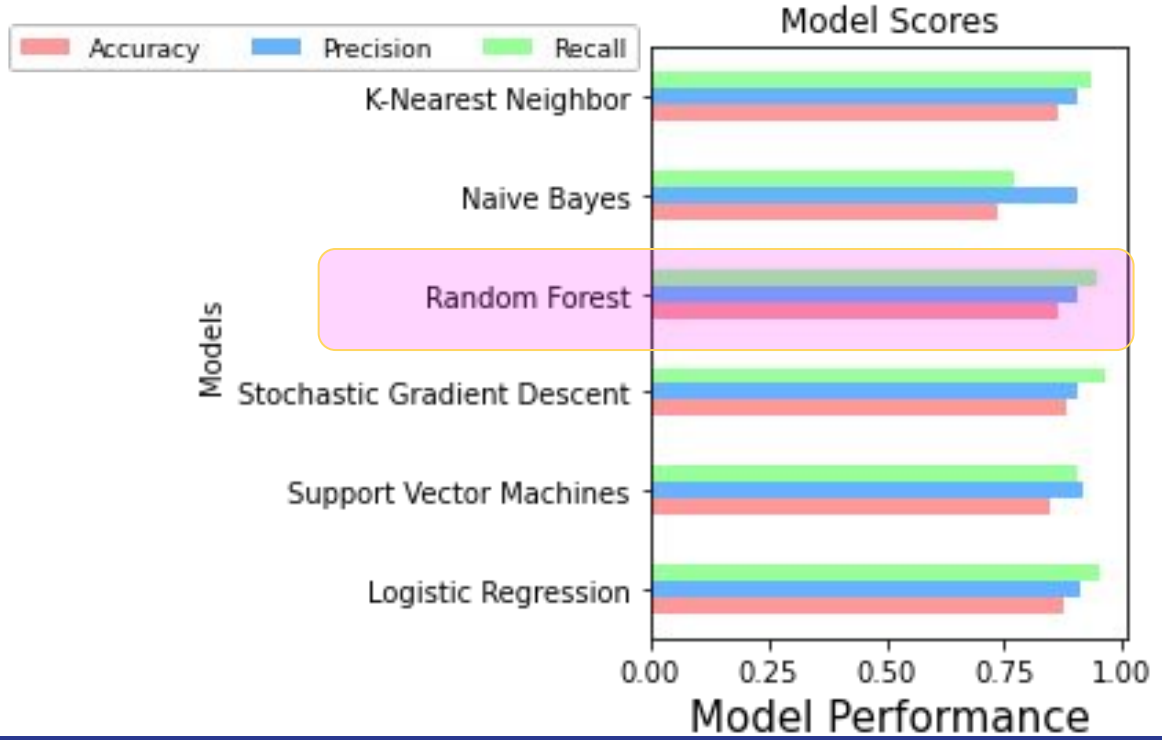


Most Frequent Words Across the Tweets



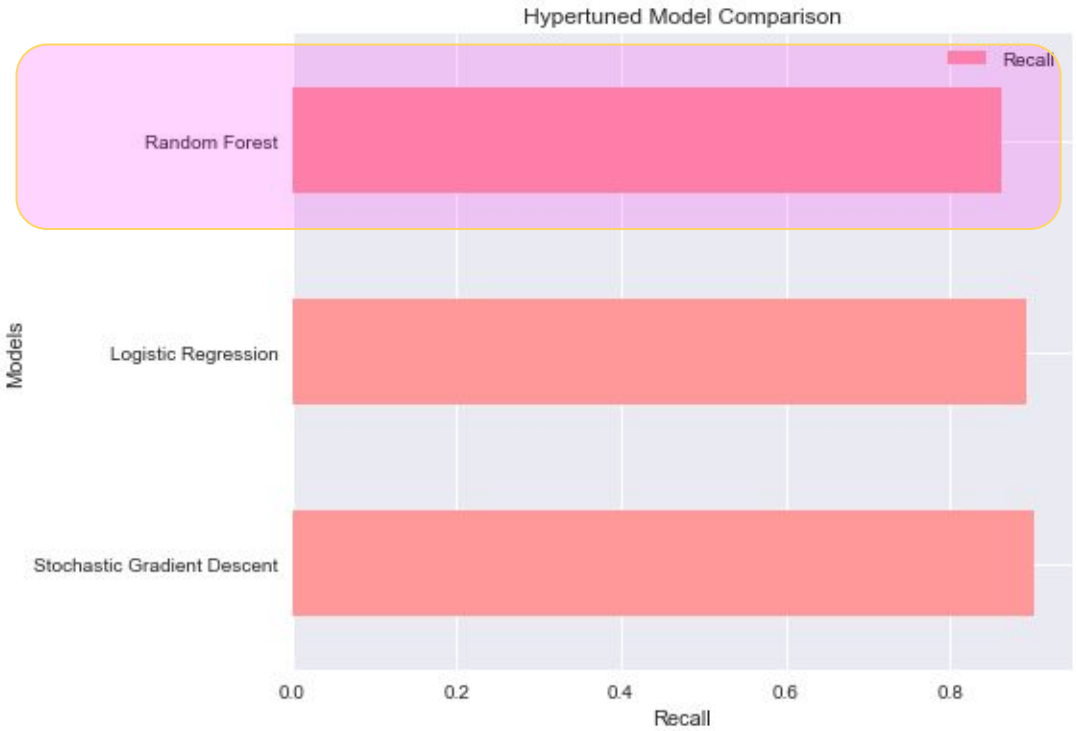
- Sxsw, mention, ipad, and link were the most commonly used words

Best model to utilize



- Many models were performed upon the data to find the ones that were initially best at finding the positive tweets.

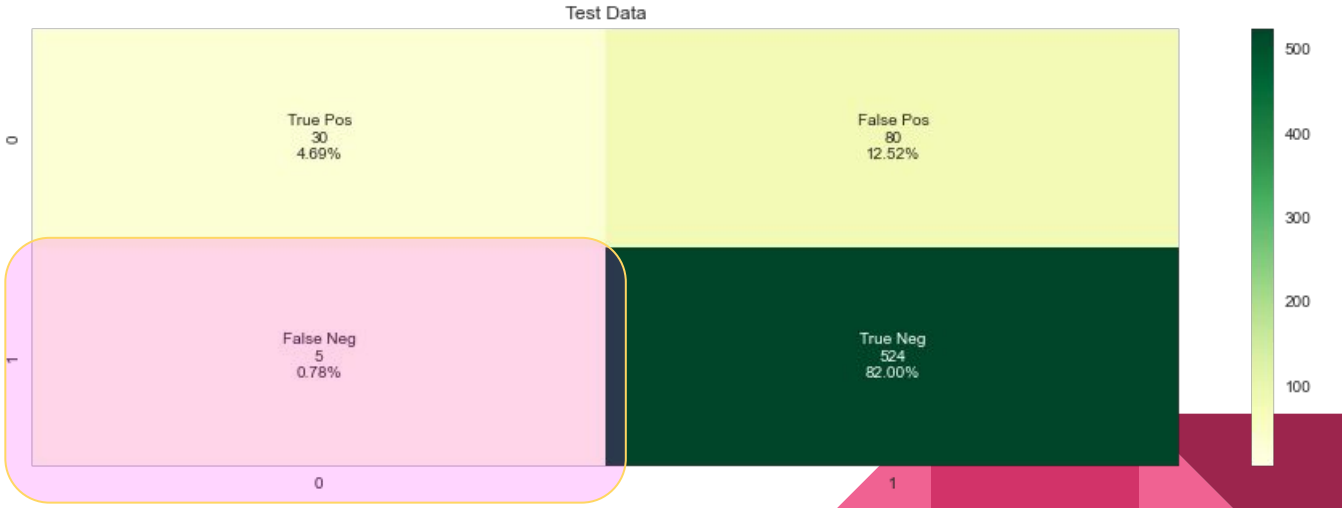
Evaluating models after Improving



- The model that performed best for the data in terms of most positive predictions was the Random Forest model.

Falsely Reported Customers

- For the best model: .78% of the time tweets were falsely reported as ‘negative’ when-they were actually positive



In conclusion

- I would recommend that Twitter can use this Random Forest model in sorting through the massive amounts of tweets sent everyday in order to make their workflow easier
- Ensure more targeted marketing
- This will lead to a better understanding of the emotions of people when they are tweeting about different devices and products

Recommendations

- The most frequent words, especially the top 5 of sxsw, mention, ipad, and link can be utilized at Twitter marketing to understand how to promote and market more effectively.

- They could also consider expanding the number of characters allowed per tweet (as they have expanded it once before) because the amount of words and characters per tweet did not appear to have a difference between the positive and negative tweets.

Next Steps

- Utilizing multi class classification for a ‘neutral’ emotion
- Attempt to code correctly sarcastic or ironic tweets
- Utilizing other vectorizers other than countvectorizer such as tfidfvectorizer.
- Hypertuning other models

Thank you for your time

Rachael Bryant

Email: rachaelbryant94@gmail.com

Github: @rabrya0072

***Made with word cloud in Python**