Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Crowdsourcing to Predict Current Popular Movie

Contents

Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Team Members

- Abhilash Mandadi Graduate Team Lead
- Josh Alphonse Undergraduate
- Chris Glynn Graduate

Motivation

Potential Applications

Use of this application will be to predict the current popular movie utilizing crowdsourcing.

Users

Potential users will be general population who would like to know the popular movie to go see. Future uses could include movie critics to use in their work. Movie creators could use the features extracted to find out what makes a movie popular.

Major Components

Data Collection

Use Twitter REST Search API Tweepy[1] in conjunction with OnConnect Data Delivery API[2]. OnConnect Data Delievery API will provide a list of current movie titles for a certain geological area. The list of movie titles will be used as the query to the Twitter REST Search API Tweepy. The form of the query text is 'movieTitleA OR movieTitleB'. In addition to the query text, the query will contain the same general geological location.

Classification

Create a classification model to remove tweets that are not relevant. Use training data to create the classification model necessary to eliminate collected tweets that are not relevant.

Feature Extraction

Extract features that will be used in predicting popular movie. Possible features to include Sentiment Scores, total number of words, frequency of some terms etc.

Brief Survey of Existing Work

The significance of our study is that the data is collected based on geographical location. Our study will focus on textual twitter data as opposed to other social media platforms. This is an identifiable gap between our study and to those of a similar topic. The Spatial data is not independent. The classification model helps eliminate uncertainty through the use of training data. The data collected can of scientific and commercial value to movie theaters and consumers in a specific area. This study aims to explore the predictive capabilities of data mining.

[1] (Tweepy, 2017)
[2] (gracenote A NIELSEN Company, 2017)

Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Data Collection

Query Terms

Queries used for predicting the current popular movie need to be dynamic. The queries are created using a three-step process. First, list of movie titles being shown in a certain area are queried using OnConnect Data Delivery API. The movie titles are then processed to remove punctuation and stop words. String.punctuation and stop_words libraries are leveraged. From there each word is used as a query. If multiple titles create the same query each query is only executed once. For example, the movie title "Pirates of the Caribbean: Dead Men Tell No Tales" would produce seven queries. One query for each "Pirates", "Caribbean", "Dead", "Men", "Tell", "No" and "Tales".

Final Query

In addition to defining the query text to search on the Twitter query performs a search from the Albany geographical location. Number of results is a maximum. For the first test a max result set of a thousand queries is set. This is due to the manual process of classification. In the future, much larger result sets will be used.

Query Considerations

- Currently the result set is too noisy. The result set includes the requested tweets, but a lot of negative tweets as well.
- Not all tokens are being parsed correctly. For example, "Disney's" is being processed to be "Disneys". Words that are similar in nature will need to be handled.
- Different queries can and will return duplicate tweets. This is currently handled in post processing. Should this consideration be handled in the collection of tweets?

Data Quality

Pre-Processing

After the tweets are collected the retrieved tweets are processed to speed up manual process of classifying tweets. This processing includes adding keys "query" and "positive". Query key stating if the tweet matches the query or not. Positive key stating if the tweet is relevant or not. Both keys are set to false by default. Then any tweets that are in the retrieved set as well as the random sample have the "query" key set to true. In addition, keys that are not used in classification are removed. The updated files for manual classification are saved to cleanRetrieved.data and cleanRandomSample.data.

Metrics and Calculations

Utilizing the cleanRandomSampleTraining.data and cleanRetrievedTraining.data, which contain retrieved tweets as well as the random sample that has been classified the metrics can be computed. In a design consideration, one is added to the numerator and denominator to ensure dividing by zero does not occur.

API Recall

API recall is computed by dividing number of retrieved tweets by the number of tweets that match the query but not retrieved. API recall for the application is 2.5%.

Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Quality Precision

Quality precision is calculated by dividing the number of positive returned tweets by the size of returned tweets. Quality precision is 6.7%.

Quality Recall

Quality recall is computed by dividing number of positive retrieved tweets by the size of all returned tweets. Quality recall is 20%.

Summary

The data retrieved is reasonable. API recall is low. API recall should improve dramatically with a larger dataset. A larger dataset will be feasible when the automated classification is complete. Quality precision is quite low pointing to the need for more precise query or more post processing to remove the negative tweets. Classification could be utilized to remove negative tweets from the retrieved tweets.

Issues Encountered

- Query length: Max query length of five hundred characters.
- Complicated queries: Complicated queries are not allowed. Documentation states complicated queries will not be processed but does not state what a complicated query is.
- Throttling: Attempting to perfect queries required querying multiple times. Once the rate limit was reached, could not test for up to fifteen minutes.
- Object not JSON serializable: Object returned by the Tweepy API is not JSON Serializable by default.

Data Exploration

Data Set Description

Set of collected tweets with attributes: length, number of query terms and tweet text.

Potential Patterns

Positive tweets are longer and contain more query terms than negative tweets. Negative tweet lengths and number of query terms are more spread out. Negative tweet lengths are evenly spread out, while positive tweet lengths are denser at longer lengths. Positive tweet query term count is evenly spread with small variation. Negative tweet query term count is dense at low number of query terms. However, there are quite a few negative tweets that contain a high number of query terms compared to positive tweets. Words used most in positive tweets are related to title and actor names.

Summary Statistics

Positive Tweet Length Mean: 106.166666667
Negative Tweet Length Mean: 91.4590354445
Positive Tweet Length STD: 27.3825775915
Negative Tweet Length STD: 40.4995775618
Positive Tweet Length Max: 135
Negative Tweet Length Max: 149

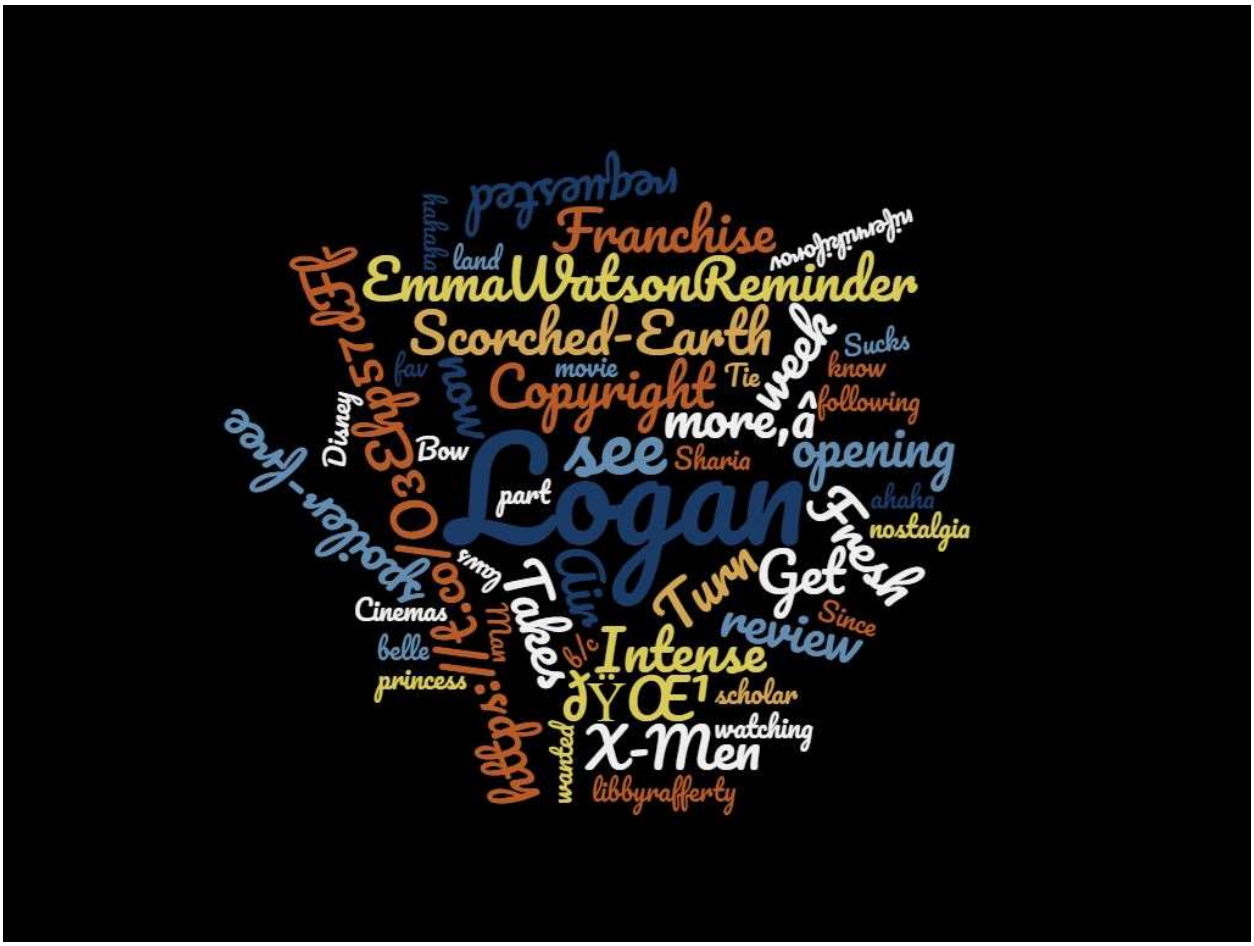
Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Positive Tweet Length Min: 57
Negative Tweet Length Min: 9
Positive Tweet Length Median: 120.0
Negative Tweet Length Median: 96.0
MAD Positive Tweet Length : 9.0
MAD Negative Tweet Length : 38.0
Positive Tweet Query Count Mean: 15.3333333333
Negative Tweet Query Count Mean: 10.5496804184
Positive Tweet Query Count STD: 10.8730042869
Negative Tweet Query Count STD: 15.7164740034
Positive Tweet Query Count Max: 27
Negative Tweet Query Count Max: 98
Positive Tweet Query Count Min: 1
Negative Tweet Query Count Min: 0
Positive Tweet Query Count Median: 15.5
Negative Tweet Query Count Median: 3.0
MAD Positive Tweet Query Count : 10.5
MAD Negative Tweet Query Count : 3.0

Visualization

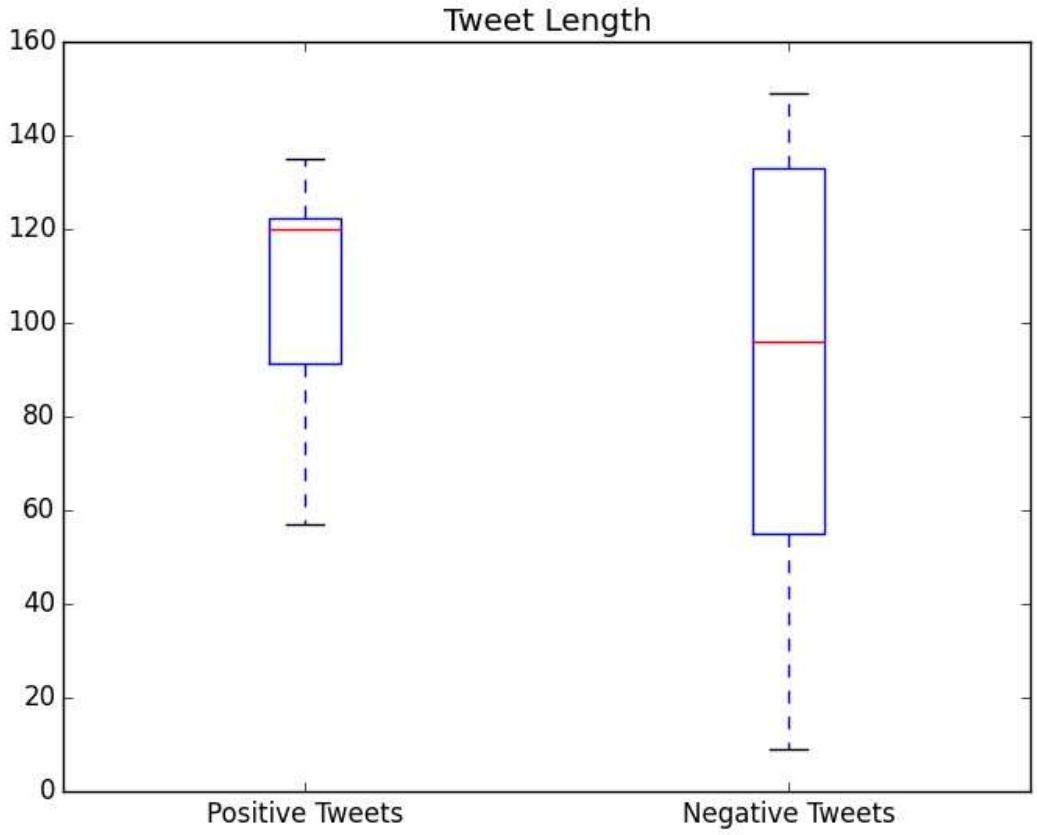
Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Positive Tweet Word Cloud



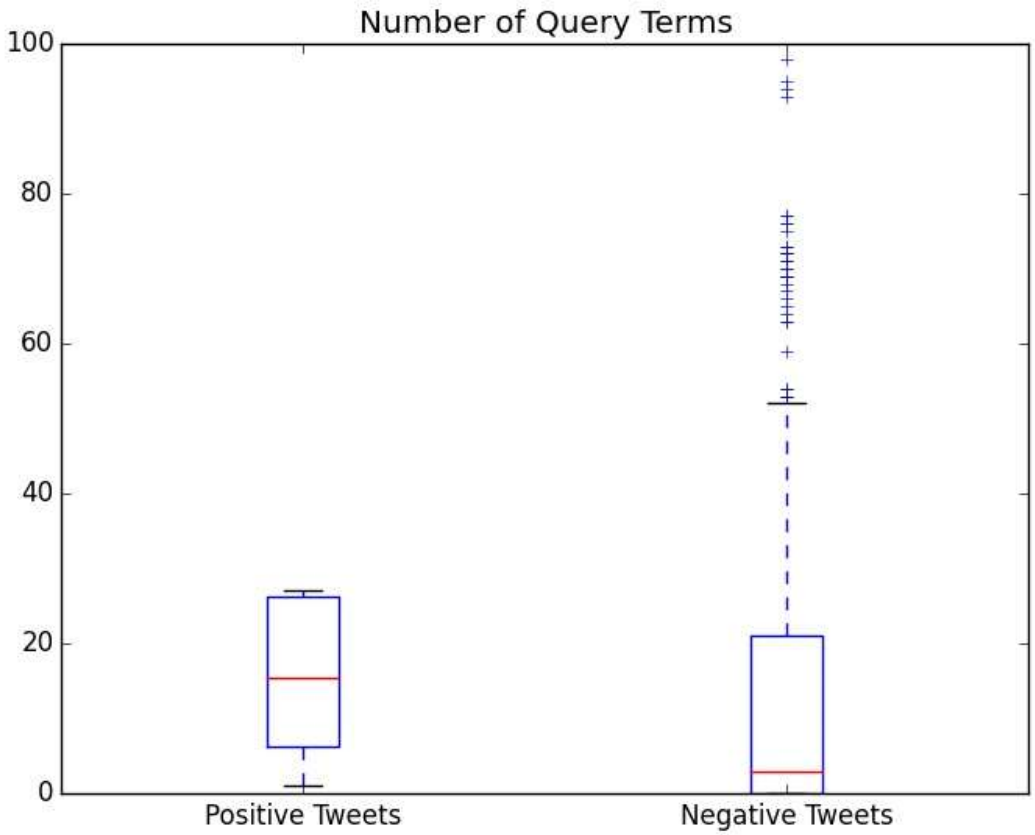
Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Tweet Length Box Plot



Name: Abhilash Mandadi, Josh Alphonse, Chris Glynn
Student ID: 001153596
Course Name/number: Data Mining/ 531
Section number: 0001

Tweet Query Term Box Plot



References

Aggarwal, C. C. (2015). *Data Mining.* Springer.

gracenote A NIELSEN Company. (2017, February 17). Retrieved from http://developer.tmsapi.com/: http://developer.tmsapi.com/

Tweepy. (2017, February 17). Retrieved from Tweepy: http://www.tweepy.org