Mid-Term Report:
Socially-Sensed Media Analysis and Performance Prediction

Manuel Marroquin
CSE Department
University of Notre Dame
mmarroqu@nd.edu

Hunter Mortemore
CSE Department
University of Notre Dame
hmortemo@nd.edu

Andrew Nemecek
CSE Department
University of Notre Dame
anemecek@nd.edu

Overview

The project aims to create a tool that will gather and utilize data from Tweets to analyze movies and television shows based on the general attitude of social users toward a piece of media. The tool will utilize language and sentiment analysis on gathered tweets to determine the attitude of each particular data point toward a piece of media. It will then aggregate these data points in order to determine how social media users in general feel about the given movie or show.

The user will be able to interact with the tool by looking up specific movies and viewing their ratings. Given further development, the tool could even make recommendations to the user based on what their own social circle likes or based on preferences that they enter in to the tool themselves.

Once developed, the tool could be used to predict box office performance for upcoming movies and shows. In addition, critical ratings of movies and shows as well as ticket prices could also be predictable based on the social attitude toward each piece of media. The tool could be used to make these predictions based on its gathered data and generated ratings.

Related Works

When individuals share their negative sentiments regarding a matter on social media it is not uncommon to see sarcasm being used as a tool to get their message across to their human audiences. Sarcasm's ambiguous natures makes it hard for humans to make out the true meaning of messages, much less a computer.

As such sarcasm has posed a challenging problem in the field of sentiment analysis and natural language processing that is very much relevant today. In Davidov et al. (2010) a semi-supervised sarcasm identification model is developed and used on both Amazon reviews and Twitter tweets. Given a set of sentences that had been labeled between 1-5, depending on their intensity of sarcasm, or lack thereof, features were extracted. Of these features that two that were utilized were syntactic and pattern based features. These two features were used in the creation of a classification model for sarcasm that yielded an F-Score of 0.83 for a large set of Twitter tweets.

This particular paper is of interest to our project as we believe that some of the strongest negative sentiments are expressed through sarcastic messages. As such our solution must take into account the subtleties of sarcasm in determining the Twitter users true sentiment towards movies and shows. In accounting for sarcasm we can achieve a more reliable and trustworthy rating that does misinterpret the sentiment of a message.. In addition to this we will keep in consideration the steps that authors use to properly process and clean their sentences in order to not mislabel a tweet based on such things as the title itself.

Proposed Solution

For our solution we intend to gather tweets from Twitter by querying a large number tweets through the title of the movie or show using Python. The tweets that will be analyzed will be tweets from up to 14 days back. As such, our tool would be best suited for media that has been in theaters for a couple days or will premiere in the near future, not for movies from previous years. From these queried tweets we filter out tweets that do not deal with the median in question. Following this, we clean up tweets by removing URLs, hashtags, usernames, and any other

elements that may alter the sentiment of the given tweet much like in Davidov et al. (2010). In order to keep consistency we replace the title of a given media with a generic string such as “MOVIE TITLE” to reduce the impact the title has on the sentiment analysis. We use existing libraries, specifically TextBlob for sentiment analysis to generate a number indicating the sentiment of a given tweet between -1 and 1. By using this generated sentiment score we will scale the score to generate a rating between 1 and 100. For each tweet we intend to adjust the outputted score by taking into account sarcasm and other factors such as number of retweets.

Once each tweet’s individual information is analyzed, our tool will aggregated these ratings and get an average overall rating for the media in question. This overall rating will take into account such things like number of likes, retweets, and if time allows sarcasm analysis of each tweet to further weigh tweets to better determine a rating that truly represents the sentiment of the Twitter community towards a given media. To get a general feel for the accuracy of our rating system, we will compare our rating to other popular rating websites such as Rotten Tomatoes. If we find huge discrepancies between our tool and existing rating websites, we will fine tune our tool to better represent public sentiment.

Once our tool has met our expectations we plan to create a web app which would be made up of a dashboard to allow a more user friendly interaction and easier display of data. Users will be able to see the current highest and lowest rated movies or shows and example tweets of what users are saying about the media being analyzed. Our tool will keep a database on previously searched movies to keep a running score that will be constantly updated, and which won’t need to rely on constantly calling the TweePy API in order to not reach rate limits..

Work Completed

Work completed over the course of the first half of this semester includes most of the back end software which will operate the tweet gathering and rating system. Two Python scripts have been written which make up the bulk of our back end implementation. The first script uses the Tweepy API to gather tweets from Twitter. It currently does this utilizing keywords related to the title of the movie or piece of media. This same script then cleans these tweets which removes user handles, RT tags, and replaced movie titles with a neutral phrase. This prevents any of these items from impacting the evaluation of each tweet when they are passed onto the next script.

The second script we have written handles the creation of a score based on the aggregated sentiment analysis of the tweets. It receives the clean tweets from the first script and then proceeds to utilize the TextBlob API to analyze them and give them a score between -1, most negative, and 1, most positive. These scores are then shifted and scaled to make them between 0, most negative, and 10, most positive. Lastly the scores of all the tweets are aggregated and averaged to produce a final sentiment score for the movie which is currently outputted to the command line.

In addition to initial software written, a list of media that is intended to be utilized to obtain results for algorithm and process evaluation has been created as well. This list can be found in Appendix 02 of this document.

Initial data has also been gathered. This data and analysis of it can be found in the *Initial Results* section of this paper.

Initial Results

Initial result data can be viewed in graphs 1 through 4 in Appendix 03 of this document. Each graph represents a different movie used for initial data gathering and shows the score outputted by the system we are building alongside the scores outputted by movie rating sites Rotten Tomatoes and IMDb. The sample sizes, or number of tweets, used to create the given sentiment scores were relatively small and all were under 100 tweets. All scores are scaled to be between 0, most negative, and 100, most positive, for comparison.

Two main trends can be seen when looking at these graphs. First, each of the sentiment scores outputted by our systems for the movies are very similar, with a range of only 7. However, the other scores have ranges of 21 and 28. This trend could be the result of small sample sizes. It could, however, also be a result of a flaw in our algorithm which tends to see tweets in a very neutral way. If we continue to see this same trend as sample sizes increase we will need to reevaluate the sentiment analysis library that we are using and consider using an alternative option or building our own in order to get more accurate sentiment analysis of tweets.

The second main trend is that the sentiment analysis scores outputted by our system do not necessarily reflect the scores outputted by the other rating systems. In other words, when other rating systems have high ratings it does not necessarily mean that our system’s outputted score is higher. This could, once again, be a result of small sample sizes or a flaw in our sentiment analysis system. It could also, however, be a result of the advertising budget of the films. Films with higher budgets, even though they may be rated poorly by watchers, experience a higher number of positive tweets that are generated by the advertising. We could have our system take this into account down the line by including data such as budget, producer notariety, or even by filtering out advertising tweets in the future.

Essentially the main takeaways from initial data are that we need to retest with larger sample sizes. If we see the same trends that we see in this data then we should reevaluate our sentiment analysis algorithm and potentially add additional information into our system for a more accurate evaluation of online sentiment.

Completed Milestones and Plan of Action

The project contains four major components: data collection, sentiment analysis, ratings calculation, and user interface development. The following milestones have been completed

1. **Data Collection**
    a. Data is collected from Twitter.
        i. A Python script is used to pull Tweets from Twitter based on keywords such as the movie/show title and relevant hashtags.
2. **Sentiment Analysis**
    a. Existing sentiment analysis tools are evaluated to determine the best fit for the needs of the project.
        i. Sentiment analysis tools were evaluated based on the tools that they offer and how well the tools work with the objectives of the project. The tool that was picked was TextBlob, a Python library that offers sentiment analysis on text strings.
    b. The sentiment analysis tool is set up to analyze Tweets and output a sentiment score.
        i. The Tweets are cleaned up and fed into the tool, which outputs a score between -1.0 and 1.0. The score is then standardized to a range from 1 to 10.
3. **Ratings Calculation**
    a. A simple algorithm is developed to calculate ratings.
        i. A basic algorithm was created that translates a sentiment score into a rating for each movie and show.
4. **User Interface Development**
    a. A simple version of the tool is created as a console app
        i. A Python script was written that will allow the user to input the name of a movie or show and the app will output the calculated rating..

With the progress made thus far and the proposed milestones, no change will be made to the proposed plan of action. Development will continue on the following milestones according to the following plan of action:

1. **Data Collection**
    a. Data is collected from Twitter
        i. Data will continue to be collected as movies are released.
2. **Sentiment Analysis**
    a. All milestones have been completed.
3. **Ratings Calculation**
    a. An accurate algorithm is developed to calculate ratings.
        i. The basic algorithm will be improved iteratively to create an algorithm that produces accurate results. Accuracy will be determined by comparing the calculated ratings to “ground truth” ratings. The ratings that will be considered the ground truth will be the ratings on Rotten Tomatoes.
        ii. Iterations will be produced throughout the semester with a final algorithm planned to be completed by 4/29
4. **User Interface Development**
    a. A web interface is developed with full functionality
        i. A web app will be created using a service such as GitLab Pages. This app will have more features than the console app such as the ability to view Tweets about movies/shows in addition to their rating.
        ii. Planned completion: 4/29

Works Cited

[1] Davidov, D., O. Tsur, and A. Rappoport. 2010.Semi-supervised recognition of sarcastic sentences in twitter and amazon. In CoNLL

Appendix 01

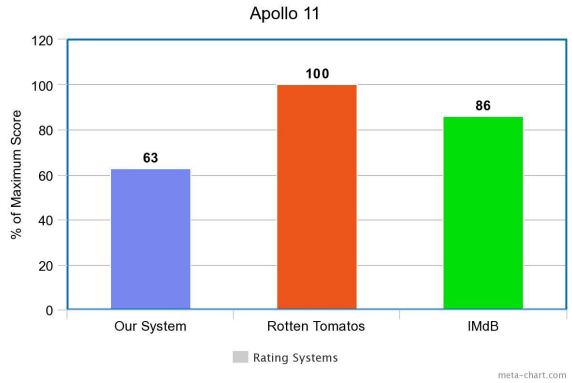
The following GitLab repository contains the code for this project:

https://gitlab.com/HunterMortemore/socially-sensed-movie-analysis.git

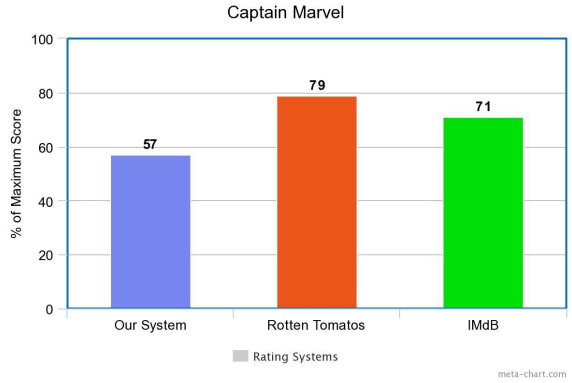
Appendix 02

Movies:	
The Hole in the Ground	(01MAR)
Apollo 11	(01MAR)
Giant Little Ones	(01MAR)
Captain Marvel	(08MAR)
Hotel Mumbai	(22MAR)
Out of Blue	(22MAR)
Dumbo	(29MAR)
A Vigilante	(29MAR)
Shazam!	(05APR)
The Best of Enemies	(05APR)
Television:	
The Widow	(01MAR)
American Idol, Season 17	(03MAR)
A.P. Bio	(07MAR)
Pretty Little Liars: The Perfectionists	(20MAR)
Million Dollar Mile	(27MAR)
Game of Thrones, Season 8	(14APR)
The 100, Season 3	(30APR)

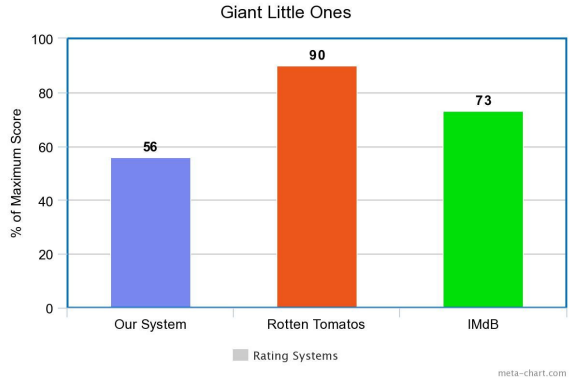
Appendix 03



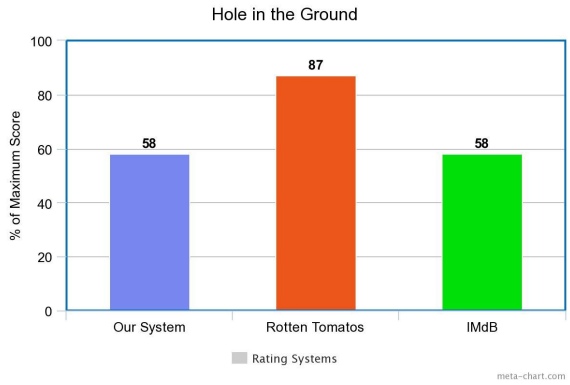
Graph 1: *Apollo 11* Rating Scores



Graph 2: *Captain Marvel* Rating Scores



Graph 3: *Giant Little Ones* Rating Scores



Graph 4: *Hole in the Ground* Rating Scores