Harvard University
Data Science II - Spring 2018
Final Project - Milestone 3 - EDA & Revised Project Statement

Movie Classification Using Plot Descriptions - Canvas Group #11

Andrew Lund, andrewlund@g.harvard.edu
Nicholas Morgan, nim607@g.harvard.edu
Amay Umradia, amayumradia@gmail.com
Charles Webb, cwebb@college.harvard.edu

Data Description

For our project, we have scraped the data of 1000 movies from both The Movie Database (TMDB) and the Internet Movie Database (IMDB). This data includes TMDB and IMDB movie IDs, genre labels, titles, plots, popularity metrics (score, vote average, vote count), and release date.

In our project we are most concerned with the content of the plot descriptions, since these will be our predictors of movie genres, but some of the other metrics, like popularity and release date, may offer insights into the distributions of our data, as well as answer some inherent questions we have about what we scraped.

The initial steps we have taken (prior to in-depth modeling and analysis) to explore our data include applying a binary representation to both the TMDB and IMDB plots. This will allow our models to employ one-vs-rest classification and precision-recall evaluation for each genre a movie may be classified as. We have also cleaned the plots including dropping stop words, lowering, and tokenizing. Finally, we have transformed the plots into both a bag-of-words Term Frequency Inverse Document Frequency (TFIDF) vector and word2vec (w2v) vector.

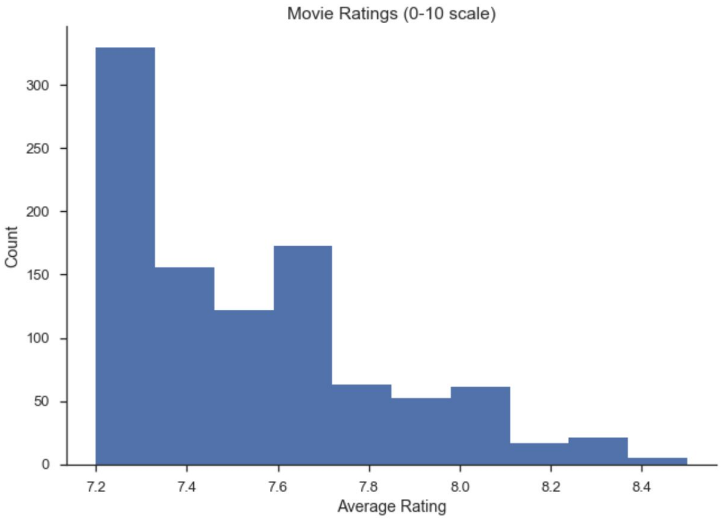
The bag-of-words TFIDF vector representations are essentially rarity scores for each word in a movie's plot. We first put the cleaned plots through sklearn's count vectorizer and removed both very common and very rare words, then applied the TFIDF algorithm. By dropping some words as previously noted, the resulting vectors are a manageable 1x1171 for each TMDB plot, and 1x2439 for each IMDB plot.

The w2v vectors were created using Google's Google News w2v model which was trained on a more than three billion word news corpus. It contains three million words, each represented by a 300-dimension vector. We decided that, for each plot, if a word in the cleaned plot (lowered, punctuation and stop word dropped, tokenized) is in the w2v model, add it to a running list for that plot, and if not, skip it. We then take the column-wise mean and store the plot as a 300-dimension average vector of its words. The assumption we made by taking the mean of each plot is that the resulting 300-dimension vector will point in the direction of one or more genres. We hypothesize that the word2vec representation of each plot might be richer for some movies since each plot is a 300-dimension vector and most TFIDF vectors will be far sparser. Though because it is a mean of the words, rather than the TFIDF score of each word in the plot, as represented in our bag-of-words transformation, some genre-specific words may be washed out in the w2v averaging.

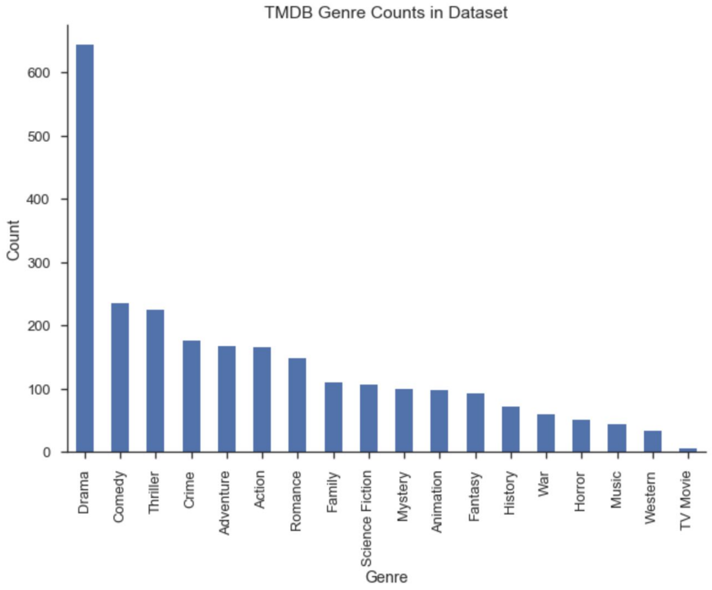
Visualizations

We produced a number of visualizations using matplotlib and seaborn to explore our scraped movie dataset.

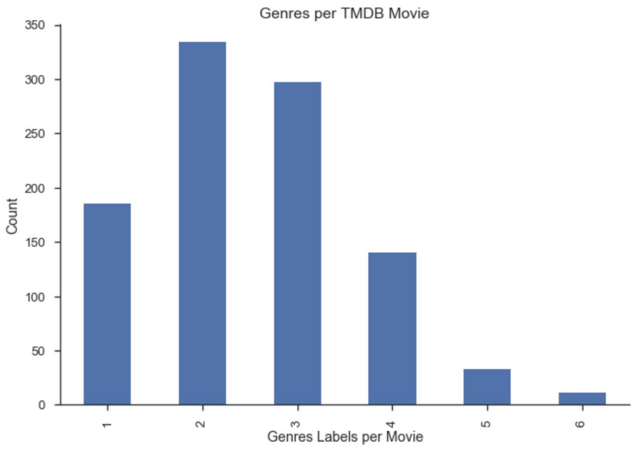
To the right we see the distribution of movie ratings, on a 1-10 scale. Most of our movies are highly-rated in the 7.2 to 8.4 range. This might lead to more descriptive and rich plots since the movies recieve more attention from critics and fans.

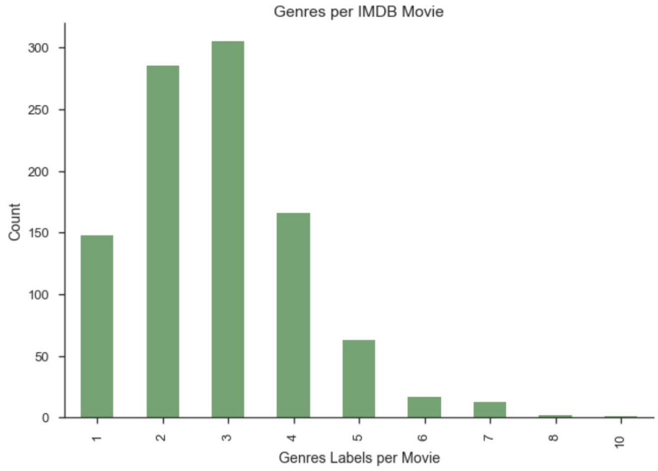


The next plot shows the distribution of TMDB genres for the entire dataset. Of our 1000 movies, more than 600 of them have drama as one of their classifications. This is not surprising, as many movies, be they comedies, thrillers, or action/adventure films, can easily be considered dramas as well.

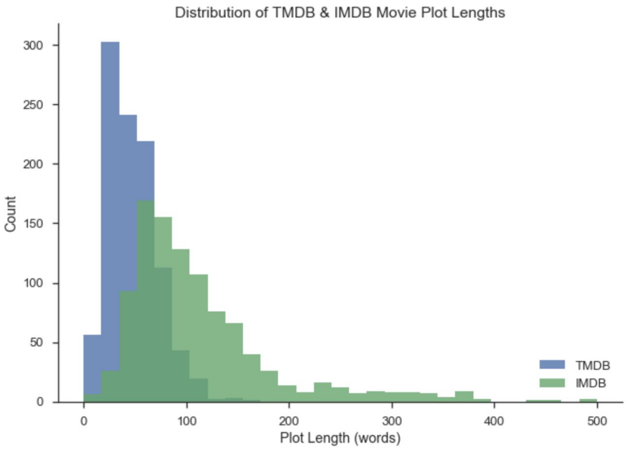


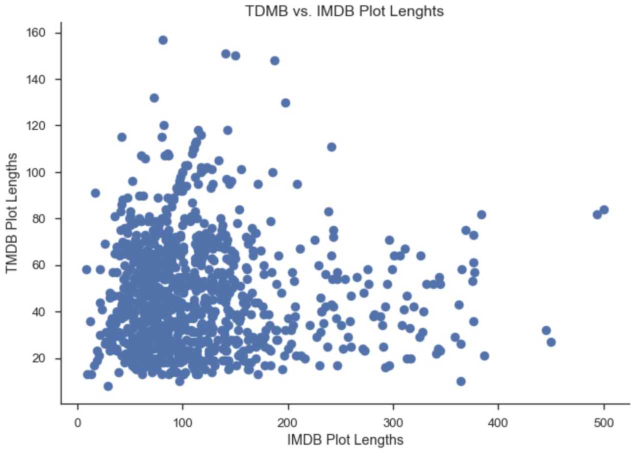
A reasonable follow-up to the overall genre distribution visualization would be, “How many genres is each movie classified as?” Below you can see those distributions for both the TMDB and IMDB sets. The average for number of genres for TMDB is 2.5, and 2.8 for IMDB. Some IMDB movies have up to 10 genres, while TMDB movies have 6 at most. This is an important consideration for us to use multi-label classification in our modeling, rather than designing models that classify each film as having only one genre.



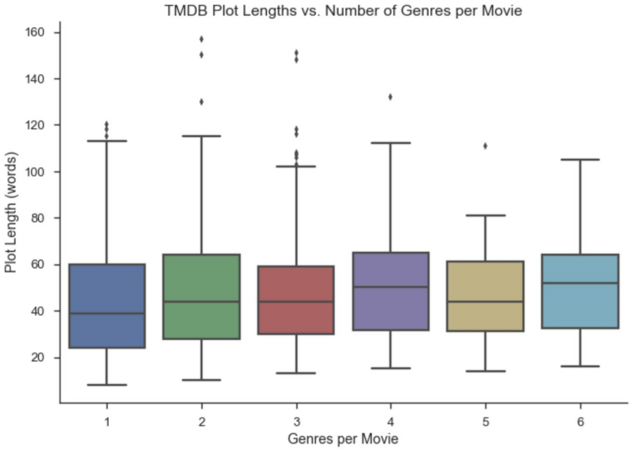


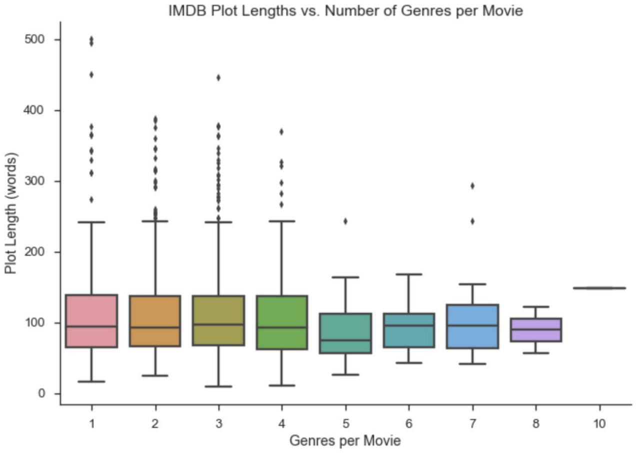
Another interesting look at our data is the word count of their respective TMDB and IMDB plots. As seen in the two visualizations below, TMDB plots are generally much shorter than IMDB plots, with average word counts of about 48 and 112, respectively. The scatter plot shows a one-to-one comparison of films and illustrates this trend nicely. This relationship leads us to hypothesize that IMDB plots may have more descriptors associated with specific genres, and may lead to increased genre classification precision and recall.





We take different look at the word length distributions in TMDB and IMDB plots in the two side-by-side boxplot visualizations below. They show the distributions of plot word length compared by genre labels per movie. Each genre count boxplot shows mostly similar distributions in their respective TMDB and IMDB datasets, with means mostly matching and a few longer outliers.





Revised Project Question

Our team's original project question was:

"Can we build a robust movie plot and genre dataset, then apply bag-of-words and word2vec representations of individual plot descriptions to accurately classify movie genres using conventional machine learning algorithms?"

We believe that original question still holds, and we add the following:

"Do the longer IMDB plots as well as richer word2vec plot representations lead to improved precision and recall classification of movie genres over the sparser bag-of-words and shorter TMDB plot vector representations?"