Wikidata, Movies, and Success

Project Report

Abstract

This project report entails analysis of different aspects (variables) of film, such as movie budget, plot, and ratings by audience and critics. Questions we tried to answer in the project are as follows:

- Can we predict the success of movies using the plot summaries in any useful way?
- Are the movie ratings of the audience significantly related to the success of movies?

Data Preparation For Both Analysis

The data we used are:

- WikiData
- Rotten Tomatoes
- OMDb API

For exploratory data analysis, though the provided WikiData dataset had data of Boolean whether each movie made a profit, it did not have the data for how much it cost to make a movie (cost) as well as how much it made from the box office (box office). For this reason, we obtained a new set of data by running a set of provided code for WikiData.

Some of the values were either missing or invalid (i.e., NaN) for WikiData and Rotten Tomatoes, so they had to be removed from the data. Also, a couple of potential outliers have been removed from the residuals of box office and audience rating. The two datasets were then inner-joined together based on IMDb identification number. At last, we were left with a total of 746 records.

For natural language processing & sentiment analysis, we made use of OMDb and Rotten Tomatoes data. Briefly speaking, as we have a seperate section ready for explaining how we processed the data, after loading the two data sets into Pandas DataFrames, we began by removing the punctuations, stop words, and lower-casing each words in the OMDb plots. For Rotten Tomatoes DataFrame, we removed rows with average_audience ratings equal to NaN since we wouldn't need it for our analysis. Then, we joined the OMDb and Rotten Tomatoes DataFrames into one, based on the IMDb identification number. In the end, we ended up with a joined DataFrame with 8755 rows that was ready for analysis.

Exploratory Data Analysis

Box Office Gross Profit vs. Audience Average Rating

The first thing that came to our mind when we asked ourselves what makes us watch a movie, it was the ratings on movie websites, such as IMDb and Rotten Tomatoes. So, we decided to see if it is the case for others as well. The scatter plot below shows the relationship between box office gross profit and audience average rating from WikiData and Rotten Tomatoes.

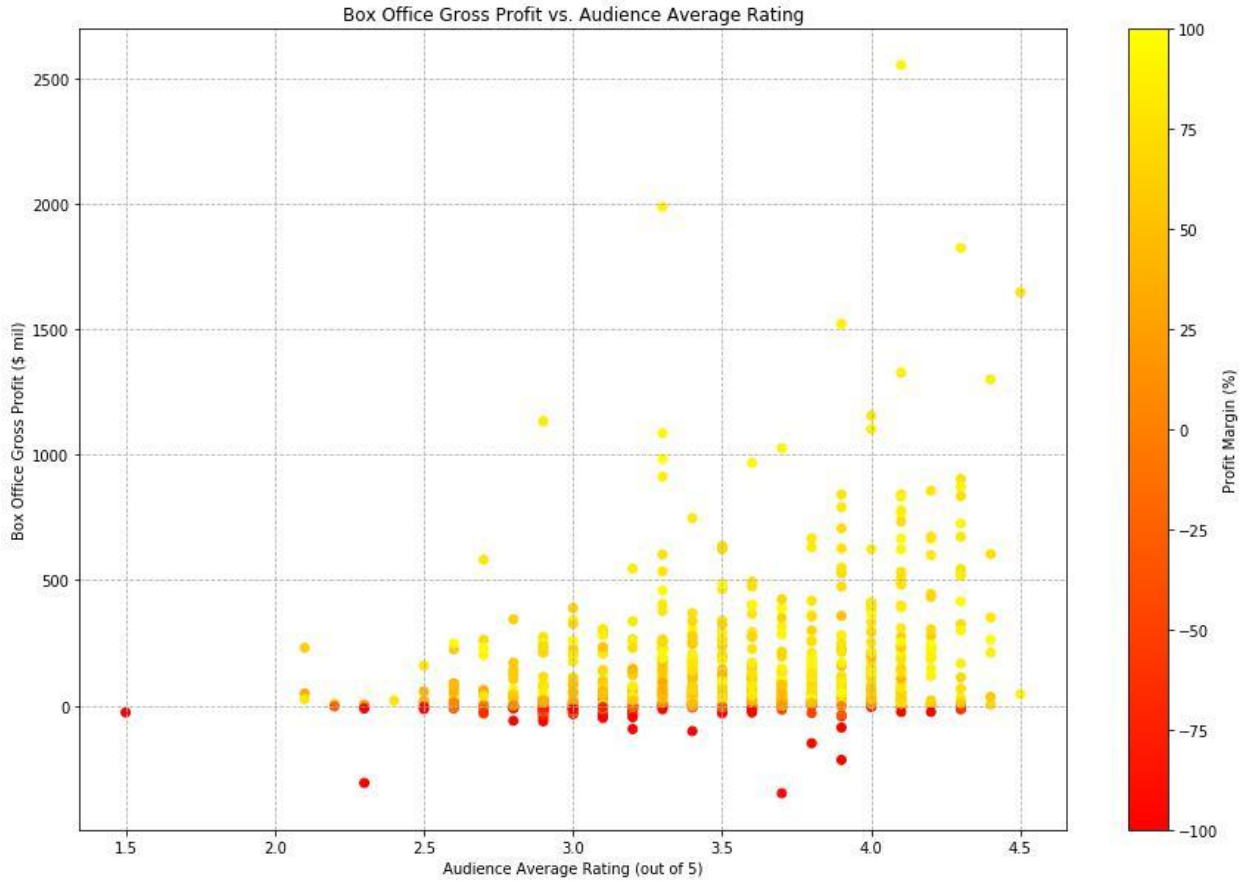

Figure 1: Relation between box office gross profit and audience average rating

Based on the scatter plot above, we can observe an increase in gross profit as audience average rating values increase. Also, we can observe higher gross profit margin values for higher audience average rating values.

Note: Some movies were tremendously unprofitable; so the profit margin values of the movies have been adjusted to -100 because what matters in this analysis is not to visualize how unprofitable they were but they were unprofitable.

Correlations between Key Variables

Key variables we thought important were as follows:

- Box office
- Cost
- Made profit? (Boolean calculated from cost and box office)
- Gross profit
- Profit margin[1]
- Audience average rating (out of 5)
- Audience percent who “liked it” (out of 100)
- Audience ratings: the count of audience reviews
- Critic average rating (out of 10)
- Critic percent who gave a positive review (out of 100)

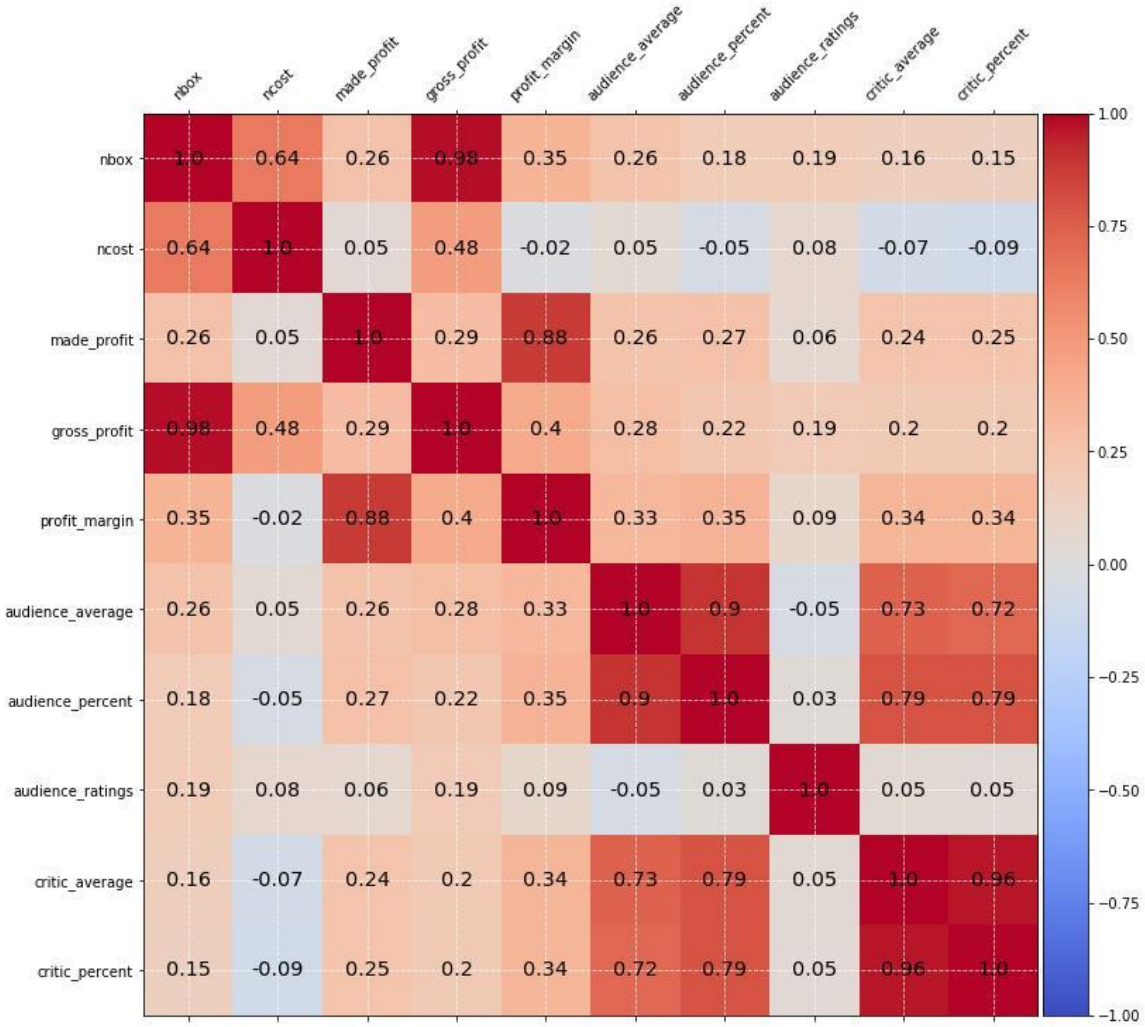

Figure 2: Correlation matrix of key variables

[1] Profit Margin = Revenue - Cost / Revenue

Based on the correlation matrix above, we were able to observe a number of interesting correlations:

1. a **very strong positive**correlation ($r \approx 0.98$) between **nbox**(box office gross) and **gross_profit**(box office gross profit after subtracting the cost);
    - This may be an indication that based on budget of a movie, most movies are profitable despite how much it cost to make one.
2. a **strong positive**correlation ($r \approx 0.64$) between **nbox**and **ncost**(film budget);
    - This may be telling us that for a movie to be successful in terms of how much it accumulates at box offices, the movie needs a bigger budget.
3. a **strong positive**correlation ($r \approx 0.73$) between **audience_average**(audience average rating) and **critic_average**(critic average rating); and
    - This was an interesting result because we thought critics tend to give much lower ratings than audience, but it turns out that they are linearly correlated to one another.
4. a **moderately strong positive**correlation ($r \approx 0.48$) between **ncost**and **gross_profit**.
    - The bigger the budget, the more profitable a movie is. Though this may not always be the case.

Ordinary Least Squares Analysis and Statistical Hypothesis Test

After the exploratory data analysis, we realized that an ordinary least squares analysis may lead us to telling whether a movie rating of the audience is statistically related to the success of a movie. Our hypotheses were that audience ratings are significantly (and linearly) correlated to the success of movies. (We defined success in terms of gross profit.) To test our hypotheses, we examined the slope of a best fit line of the two variables: audience average rating and box office gross profit.

1. Hypotheses

Is the movie rating of the audience significantly (linearly) related to the success of the movie? (**Note:**we define success in terms of the gross profit.)

H_0:β=0 H_1:β=/0

2. Test Statistic

Assumptions of ordinary least squares:

1. The sample is representative of the population.
2. The relationship between the variables is linear.
3. The residuals are normally distributed and iid.

OLS Regression Results

Dep. Variable:	y	R-squared:	0.079
Model:	OLS	Adj. R-squared:	0.078
Method:	Least Squares	F-statistic:	65.30
Date:	Thu, 01 Aug 2019	Prob (F-statistic):	2.53e-15
Time:	18:44:42	Log-Likelihood:	-5255.9
No. Observations:	761	AIC:	1.052e+04
Df Residuals:	759	BIC:	1.053e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x	150.2603	18.594	8.081	0.000	113.759	186.762
intercept	-377.6734	65.486	-5.767	0.000	-506.229	-249.118

Omnibus:	631.052	Durbin-Watson:	2.046
Prob(Omnibus):	0.000	Jarque-Bera (JB):	16323.625
Skew:	3.621	Prob(JB):	0.00
Kurtosis:	24.502	Cond. No.	28.4

Figure 3: Ordinary least squares analysis results

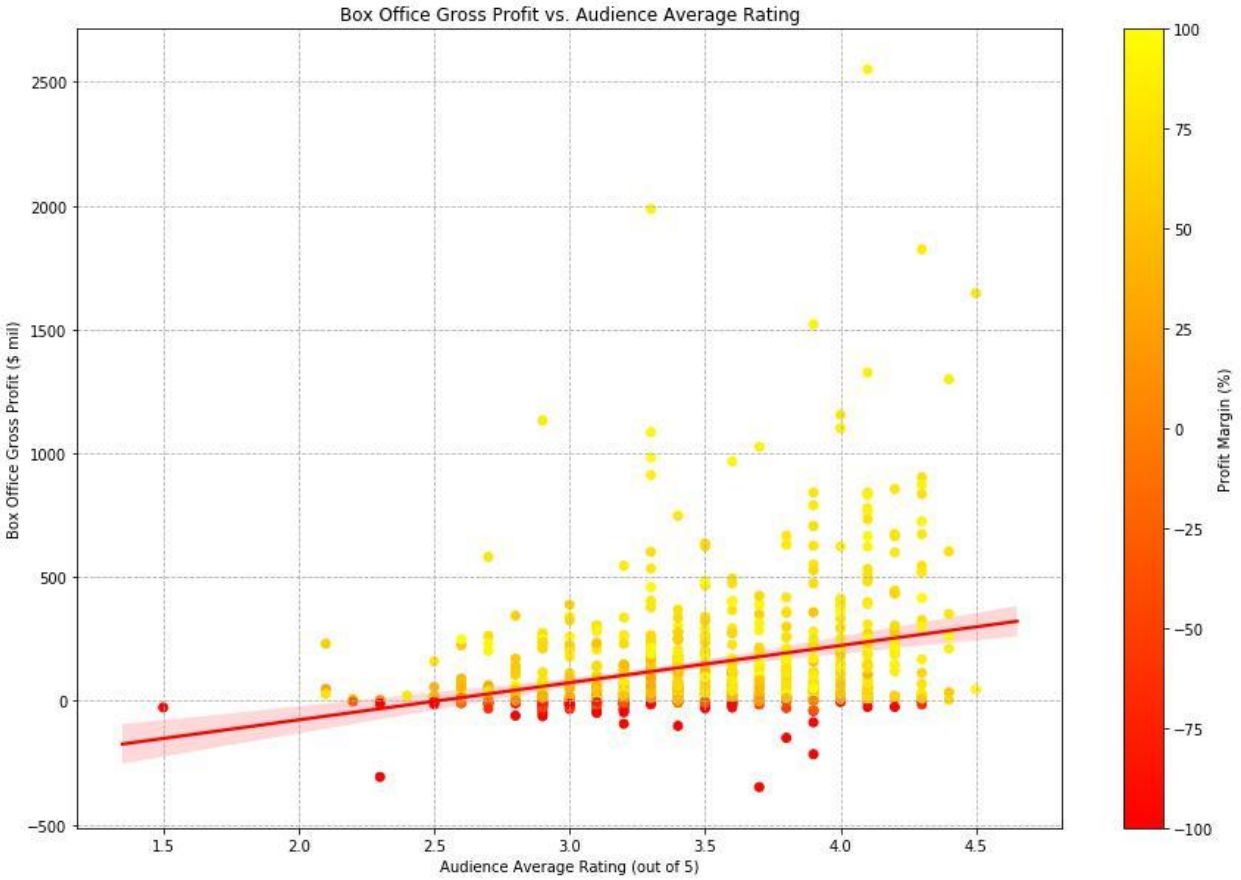

Figure 4: The best fit (linear regression) line

3. p-value

From the OLS analysis, we obtained $p - value \approx 2.525443957827175e - 15$.

4. Conclusion

Since the $p - value \approx 2.53 \times 10^{-15} \ll \alpha = 0.05$, we were able to reject the null hypothesis, H_0 , in support of the alternative hypothesis, H_1 , such that the slope is significantly different from zero. That is, we found that the movie rating of the audience is significantly (linearly) related to the success of a movie.

Movie Success Prediction with Linear Regression Model

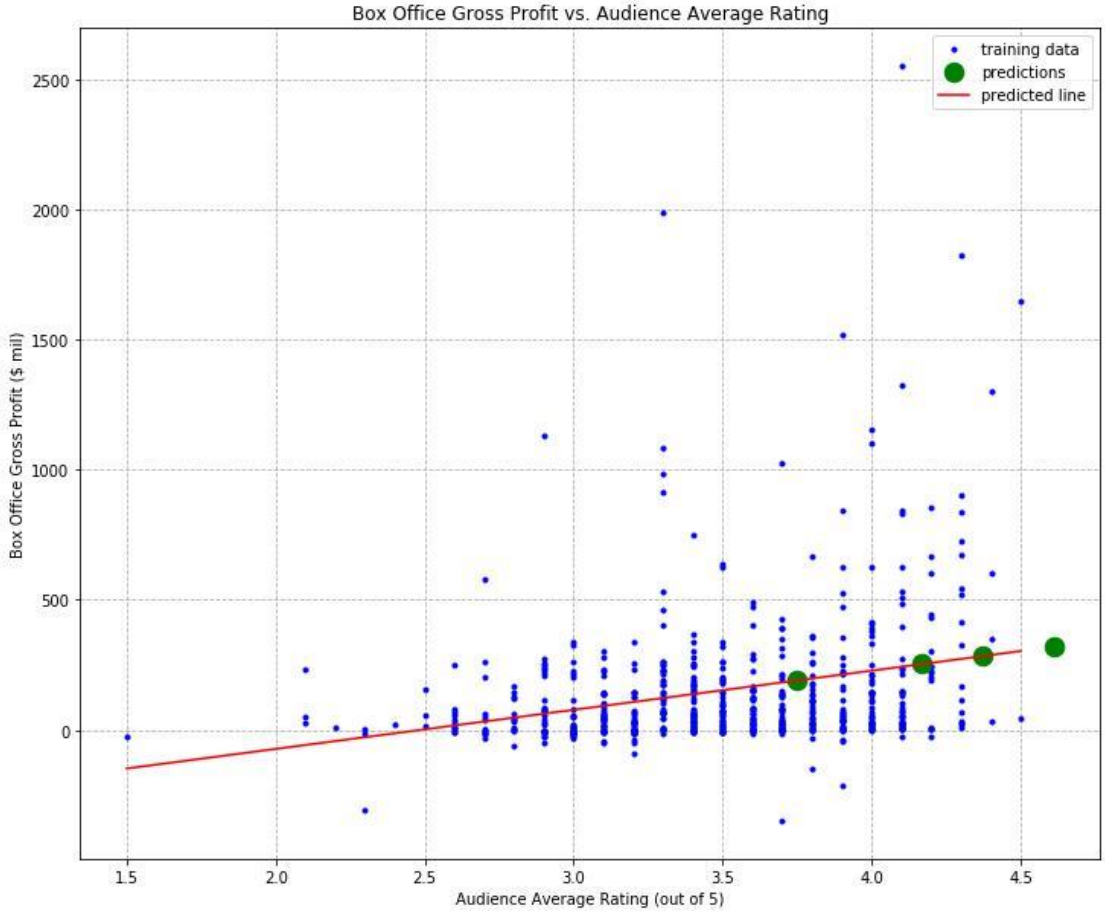

Figure 5: Scatter plot of box office vs. audience rating with predictions using degree 1 polynomial (linear) regression model

	Toy Story	Toy Story 2	Toy Story 3	Toy Story 4
Audience Average Rating[2]	3.75	4.17	4.37	4.61
Actual[3]	$373.6 M	$497.4 M	$1.067 B	$924.4 M[4]
Predicted	$193.5 M	$263.3 M	$296.5 M	$336.4 M

The scores from the model were very bad: low variance (R^2) scores and high MSE [5] value. Based on the values, we have no confidence that the score is related to how well the model will predict new never-before-seen values. This may be possibly happening when the data is actually more complex than a simple straight line. For example, in the case when the data seems to have a linear relationship but is in fact non-linear. We tried the polynomial regression technique in hopes of getting a better prediction result.

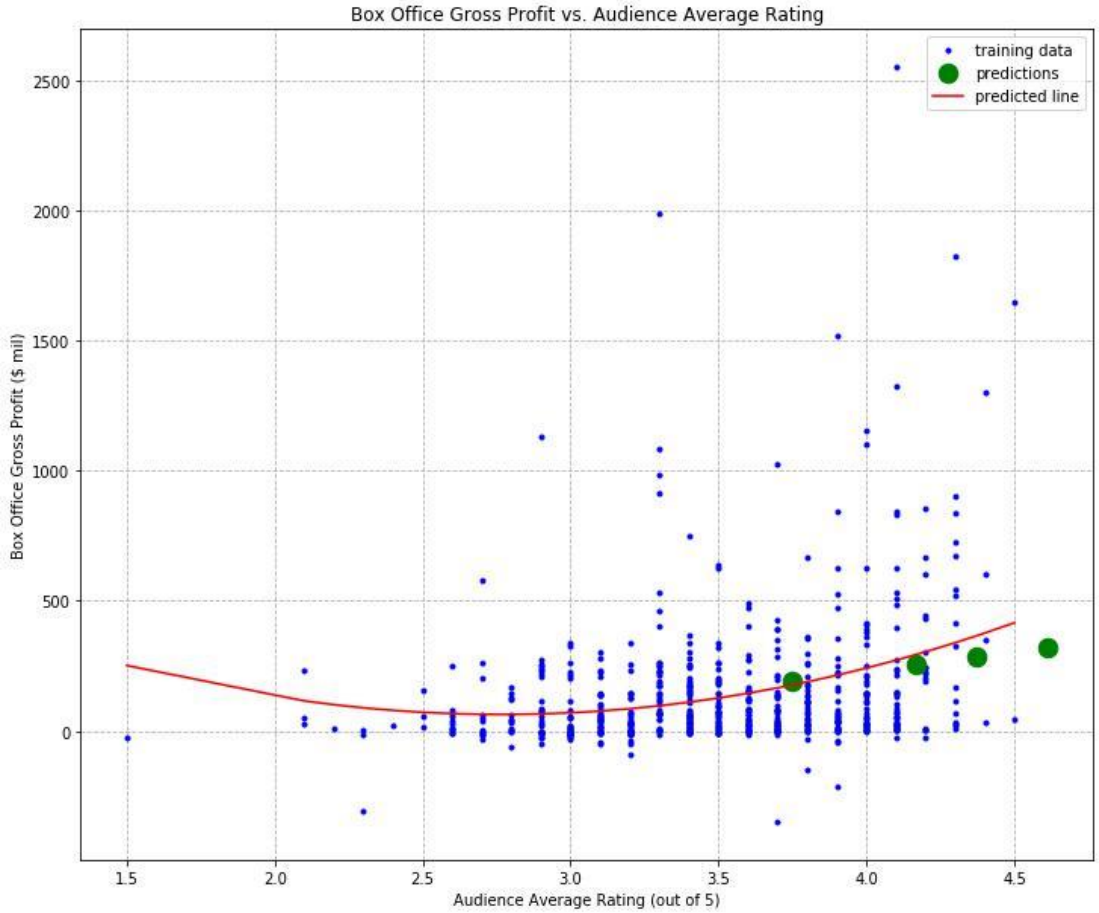

Figure 6: Scatter plot of box office vs. audience rating with predictions using degree 2 polynomial (quadratic) regression model

2Data from Rotten Tomatoes
[3] Data from Box Office Mojo
[4] As of July 31, 2019
[5] Mean Squared Error

Polynomial regression with degree 2 also failed to get a good result with similar scores as degree 1. At this moment, it looks like more enhanced tests and analyses are necessary for a better result of the prediction.

Natural Language Processing - Sentiment Analysis & Movie Success Prediction

Initial Data Processing

For this part, we used the OMDb and Rotten Tomatoes data. After reading in each JSON file into separate pandas DataFrames, the first thing we did was writing a function to go through each plot summary and transforming them into a useful format, so they are easy to work with. The initial text processing of data includes removing the punctuations and stop words which don't give any additional information. In addition, each word inside the plots was lower-cased so we don't have to deal with "Happy" and "happy" being different later on in our analysis.

Calculating the Sentiment Scores

After the initial data processing is done, the polarity scores based on the plot summary were calculated using vaderSentiment library, which is a handy sentiment analysis tool. The polarity scores were calculated for the purpose of carrying out a sentiment analysis. After classifying the plots into three different categories (positive, negative, and neutral), they were converted into numeric values, with positive, neutral, negative being 1, 0, -1, respectively. For visualization purpose, we generated the word cloud diagrams. Figure 7 indicates that words such as “love”, “friends”, and “help” appeared significant number of times in the plot summaries which were classified as positive. In contrast, Figure 8 shows that negative words such as “death”, “war”, “fight” appeared several times across the plot summaries which were classified as negative.

Figure 7: Positive words

Figure 8: Negative words

More Data Processing before Training the Data Set

In order to get the audience ratings of each movie associated with the plot summary data, we joined the OMDb and Rotten Tomatoes DataFrames together. While doing so, rows with NaN values were dropped while unnecessary columns were excluded, since we would not need them for our sentiment analysis and prediction. Before we start training our data, CountVectorizer was used to convert the text plots into a matrix of tokens. Since we needed two features for our X data to train-test split, a column numerical_category was added to the train and test data set. So our X data became plots and numerical values indicating positivity/negativity/neutrality of the plots.

Training Data

After everything was processed correctly, we trained-split the data. Our X data set contains two features (plot and numerical values indicating whether it is a positive/neutral/negative plot) and y data set contains the numerical values referring to whether the movie is a success/neutral/flop based on the average audience ratings. Out of many models to choose from such as GaussianNB and K-nearest neighbours, we chose SVM since it was giving us the best result compared to the other classifiers. When performing the classification task with a relatively not large subset of training data, it seems to be doing better compared to others. In addition, we were able to fine-tune the parameters such as kernel type and cost value for SVM, which was a plus. The kernel type and C parameter were set to linear, and 0.01, respectively. Specifically, we chose 0.01 as the cost value to avoid overfitting and underfitting of the data. If we set the C value to be too small, we would end up with the margin between the support vectors and the decision boundary being too wide. This would lead to more training errors, although better validation scores. On the other hand, if we set the C value to be too large, the margin between the support vectors and the decision boundary would get too close to each other. Hence, we would get worse validation scores but with less training errors. After training, we got 0.6144358154408406 for the training score which is approximately 0.61, and is not significantly low.

Prediction

The validation score 0.61 shows that the plot summaries can indeed measure the success of a movie to a degree, although not perfectly. In figure 9 and 10, the two bar graphs show that the movies with positive plots have the highest average audience & critics rating, with neutral being the second highest and negative plots having the lowest rating. As one can see, this trend is consistent in both the audience and critic rating. As a side note, the reason for separating audience and critic rating into two separate bar graphs is because their scales are different, so the critic rating widens the bar graph a bit too much, which makes it harder to notice the subtle differences between each category. Anyways, this finding correlates to our first assumption that the movies with positive plots would have higher ratings compared to the movies with negative plots. Although there are many factors to consider when we define the definition of a successful movie such as profit, we believe the rating is certainly one important measure we can use in the prediction.

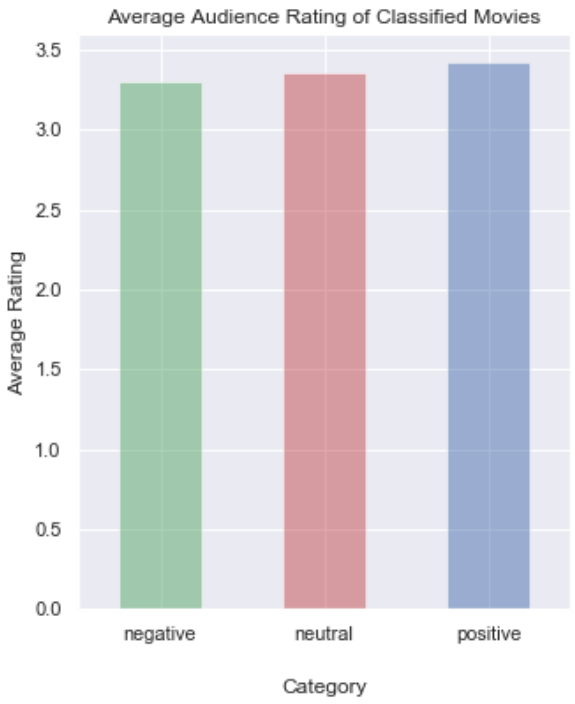

Figure 9: Average audience rating of classified movies

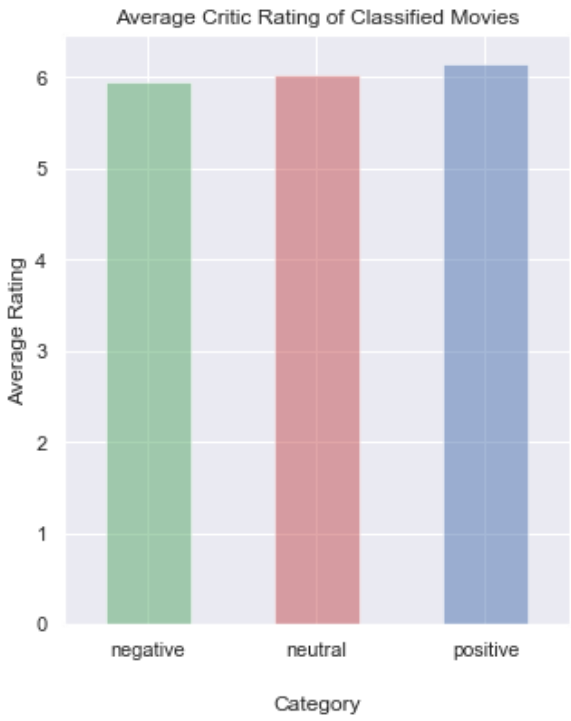

Figure 10: Average critic rating of classified movies

Obstacles and Limitations of the Project

Movie Success

We really wanted to see the relationship (if any) between gross (success measurement) and popularity of actors/directors. For example, we often talk about cast members of a movie, and when the cast is awesome, we keep the movie in our go-to-watch lists before its release. We thought the number of Facebook page likes or the number of Twitter followers of an actor or a director would be a great measurement of the popularity. We tried to get the data using either Facebook or Twitter APIs but due to their recent policy changes[6], it was difficult for us to get the data we needed. Also, we were not able to find similar data on the web.

Natural Language Processing & Sentiment Analysis

While converting the text plots into matrix of tokens, we had to decide whether to go with all the features in which there are 33582 of them or go with the reduced number, since it was taking up

6Though data are already public, Facebook now requires developers to submit a review for use of their data. More details can be found here.

too much time to process all of them. As a result, we decided to go with the feature size 10000 words, which we believe is sufficient enough to get some significant results out of it.

Another hard part was getting the two features into one combined X data for training and validation set. Our final X data contains matrix tokens for plots and numerical values indicating the sentiment of the plots. With the help of NumPy and SciPy functions such as np.atleast_2d which takes one dimensional array and view it as two dimensional array, and SciPy.sparse.hstack which horizontally stacks the sparse matrices, we were able to have the two features in the X data set successfully, while having numerical values indicating the successfulness of movies as the y data set.