Analysis of Movie Genre and Plot Overview

Brian Lee and Ha Thu Nguyen

Overview

The movie industry is full of intriguing mysteries and presents an interesting opportunity to apply data science. Therefore, the purpose of this project was to analyze the relationship between the success of a movie and a movie's plot. In particular, the problem was refined to the perspective of a production company to help make necessary decisions. The specific problems that are addressed within this project range from data analysis to machine learning models that can provide useful insights. The summary of the different problems we are addressing are shown below:

1. What is the current trend of movie genres in the industry?
2. Which genres tend to be more highly rated by viewers? Is there a relationship between movie genres and movie rating?
3. Based on a movie's plot overview, which genre should this movie be classified to?
4. What is the likelihood that a movie will return 'high revenue' before the movie is released?
5. Is it possible to build a movie recommendation system to determine a target audience?

The five problems are all questions from the perspective of a production company. For example, by building a movie recommendation system based off of the similarity of a movie's plot overview, a production company can potentially target a specific audience that watches similar movies.

Acquiring Data

The data used for analysis originates from The Movie Database (TMDb). TMDb is a popular database for movies and TV shows. Specifically, the dataset we used was a CSV file extracted from the following link from Kaggle. In summary, the dataset includes roughly 4800 movies and relevant information such as genre, popularity, and plot overview.

Data Processing

The dataset from TMDb included some movies with missing plot overviews. After dropping those movies and some unused columns, we were left with a total of 4799 movies. It is also to be noted that the dataset from TMDb included columns that contained JSON structure. Therefore, the JSON columns such as 'genres' were parsed accordingly in Python.

Due to the fact that the plot overviews are in text format, the text was processed into a vectorized form in order to effectively use this information in our analysis. First, the text in the plot overview was cleaned by:

1. Conversion to lowercase letters
2. Removal of punctuation (e.g. !, #, ect.)
3. Removal of stopwords (e.g. and, at, ect.)
4. Word stemming (e.g. likely -> like)

To get a better understanding of the text in plot overviews, a word cloud of the most common words is shown in Figure 1. Furthermore, on average, a plot overview for a movie consisted of roughly 305 words. The histogram in Figure 2 displays the distribution of the number of words in a plot overview.

Figure 1: Word Cloud of Most Common Words in Plot Overviews

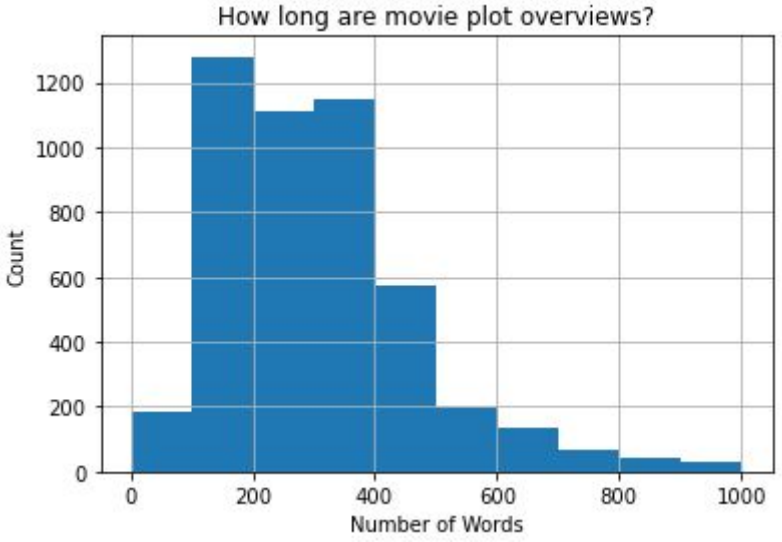


Figure 2: How Long are Movie Plot Overviews?

Finally, the text in the plot overview was converted into numerical feature vectors to use in machine learning models using a technique called TF-IDF (Term Frequency Inverse Document Frequency). TF-IDF is a technique to quantify the number of times a word appears in a document, while assigning a weight to each word to determine the importance of the

word to the document. The result of converting text into TF-IDF using the sklearn library was a sparse matrix, which was used as features in our predictive model.

Movie Genre Analysis

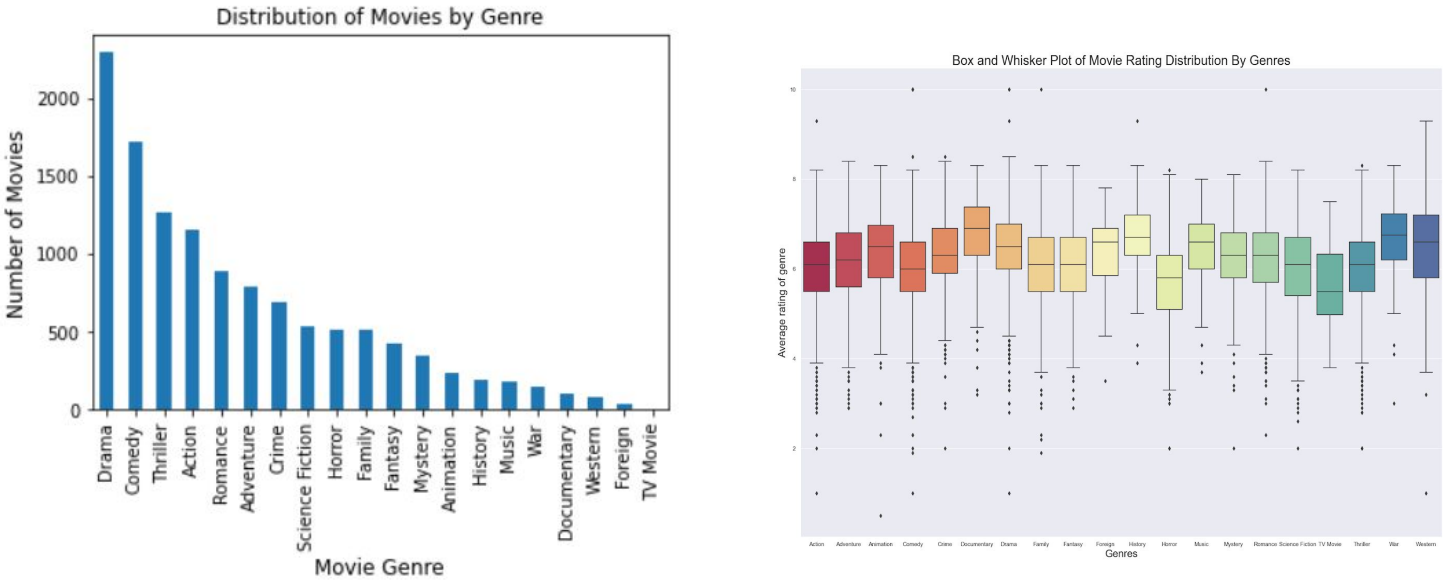


Figure 3: Analysis of Genres

In our dataset, we clearly have imbalanced genres. During the period of 1916-2017, the ‘drama’ genre had over 2000 movies in total, while ‘TV Movie’ only had 8 movies in total. However, the average voting for each genre was somewhat similar in general, which was around 6.0-7.0.

In terms of the trend of movie genres, it can be seen from Figure 4 on the next page that the film industry had been growing faster during the late 90s to early 2000s as the number of movies increased rapidly during that time and rocketed up in the late 90s. The plot also shows that the ‘drama’ genre has remained as the most produced genre over the time, followed by the ‘comedy’ genre. However, it is to be noted that this relationship cannot be formally justified because the dataset from Kaggle might have focused on scraping movies that were more recent.

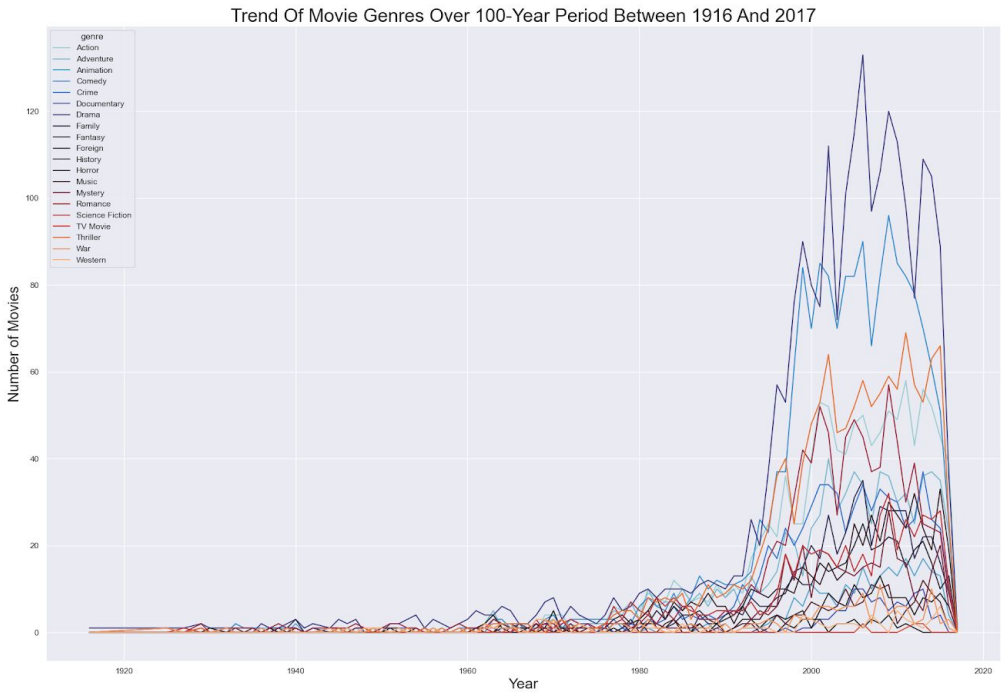


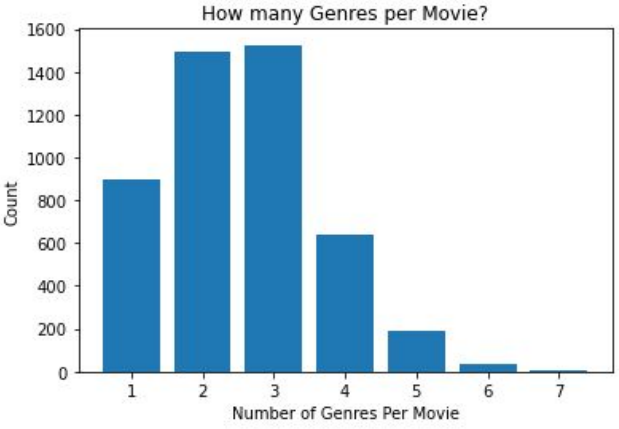
Figure 5: Trend of Genres Over Time

Now, the question we have on our hand is "Do the genres affect movie rating?". To answer this question, we looked at the audiences' voting distribution by genres for our dataset. The plot illustrates that the voting by genres is normally distributed. In addition, performing the normality test gives a p-value of 0.27 (> 0.05), indicating there is no evidence that the distribution is non-normal. We proceed to categorize the genres rank by the quantile, where a movie with rating above the 75th percentile (7.0) is considered high rated, a movie with rating below the 25th percentile (3.7) is considered low rated, and movies with rating lying in between are considered medium rated. After grouping by rank and aggregating, we ended up with a contingency table with categories of ranks (high, medium, low) and categories of genres. Since more than 80% of the contingency table contained at least 5 observations in the cells, we performed a Chi-squared test to see if there were any relationships between the genres and the rating. A p-value of 1.1×10^{-60} (< 0.05) proves that the genres do have an influence on the average rating given by the audiences.

```
P-value for normality test of voting distribution: 0.2717620351281541
The 75th percentile is 7.025
The 25th percentile is 3.6750000000000003
P-value for Chi-square Test is 1.1021277628681791e-60
```

Figure 6: Results of Chi-square test

Movie Genre Classification



While exploring our dataset, we realized that most movies are associated with several genres, rather than only a single genre. Therefore, our attempt was to attack this task as a multi-label classification problem. For each movie, we used the MultiLabelBinarizer function to transform the list of genres into a binary label representation, where the genre will have the value of 1 if the movie is associated with it and have the value of 0 otherwise. We used the cleaned and processed movie plots as described in the Data Processing section to train our model with the OneVsRestClassifier. For this model, we created a pipeline with TfidfTransformer, and tried out several function estimators to fit our training data, some of which are Gradient Boost, Stochastic Gradient Descent (SGD), KNeighbors, Random Forest and SVC Classifier. Most of the models gave a precision score of around 45%-50%, and the SGD estimator performed the best. It managed to give a F1-score of 55%-65% for most of the genres. However, the F1-score was low for instances of genres with a low number of movies associated with them. This means that those rarely occurring genres were never classified.

With 20 unique genres for our dataset in total, this may not be the most efficient way to train the model, since it needs to do the work 20 times and compare the results to pick the highest one. However, at the very least this model still works and produces some useful information. The limitation of this analysis is the imbalanced data with some of the rare genres. If we were to have a more balanced data, or more data in general, our model would have predicted more accurately and given a higher precision score. For our current dataset, we believe a precision score of 50% is a reasonable baseline to predict the genres based on movie plots.

OVR - SGD Classifier:

	precision	recall	f1-score	support
Action	0.57	0.69	0.63	270
Adventure	0.54	0.55	0.55	212
Animation	0.47	0.37	0.42	70
Comedy	0.62	0.65	0.63	420
Crime	0.53	0.54	0.53	163
Documentary	0.60	0.35	0.44	26
Drama	0.68	0.66	0.67	569
Family	0.55	0.56	0.55	127
Fantasy	0.49	0.50	0.49	105
Foreign	0.00	0.00	0.00	10
History	0.39	0.31	0.34	49
Horror	0.59	0.59	0.59	133
Music	0.60	0.43	0.50	60
Mystery	0.32	0.31	0.31	80
Romance	0.59	0.64	0.61	236
Science Fiction	0.64	0.59	0.62	142
TV Movie	0.00	0.00	0.00	3
Thriller	0.59	0.58	0.59	317
War	0.56	0.43	0.48	35
Western	0.67	0.20	0.31	20
micro avg	0.59	0.58	0.59	3047
macro avg	0.50	0.45	0.46	3047
weighted avg	0.58	0.58	0.58	3047
samples avg	0.59	0.61	0.56	3047

Precision score: 0.500388863146045

Figure 7: Results of Genre Classification

Likelihood that a Movie will Return 'High Revenue' Before the Movie is Released

This problem was inspired from a production company's perspective. Suppose, a production company wrote a plot summary for a new movie that is coming out. The production company can utilize this model to assess the likelihood that a movie will return 'high revenue' using the plot summary as a predictor variable.

This model was structured as a binary classification problem with using only the plot summary in TF-IDF form as a predictor. After assessing the distribution of movie revenues, 'high revenue' was defined if the movie revenue exceeded the 75th percentile. Specifically, the target variable was defined as:

- 'High Revenue' if Revenue ≥ $93,637,756
- 'Low Revenue' if Revenue < $93,637,756

The results of the model performance on the test set is shown in Figure 8. Although the overall accuracy is 73%, the model can potentially lead to useful insights because it is able to correctly identify if a movie will return high revenue 45% of the time. The limitations of this model are also discussed in the Limitations section.

```
              precision    recall  f1-score   support

           0       0.80      0.85      0.82      1073
           1       0.45      0.36      0.40       359

    accuracy                           0.73      1432
   macro avg       0.63      0.61      0.61      1432
weighted avg       0.71      0.73      0.72      1432

Confusion Matrix:
 [[913 160]
 [228 131]]

Accuracy:  0.729050279329609
```

Figure 8: Test Set Results - Likelihood of Movie to Return High Revenue

After training the model using a Random Forest model, the variable importance was analyzed. The top 10 most important words to predict if a movie will return high revenue is shown in Figure 9. Due to the low variable importance of the top 10 words, there seems to be no specific word that has a strong effect on revenue.

Word	Variable Importance
world	0.013482
name	0.0109639
mother	0.00864634
life	0.00701547
earth	0.00656843
friend	0.00617121
face	0.00608967
must	0.00595519
find	0.00586834

Figure 9: Variable Importance

Movie Recommendation System

Finally, a movie recommendation system based on the similarity of movie plot overviews was created. The purpose of this recommendation system was to give useful insights to the production company. For example, a company can input their movie plot overviews, and use the results of the recommendation system to target this new movie to the desired audience that tends to watch the recommended movies.

The recommendation system was created by developing a similarity score based on how similar the movie plot overviews are. To measure the similarity between movie plot overviews, cosine similarity was used to measure how similar the documents are irrespective of their size. The cosine similarity is a powerful metric because even if two movie plots are far apart by magnitude using Euclidean distance, their angles might be closer. Once again, the TF-IDF vectorized form of the plot overview was used for the recommendation system.

The recommendation system calculates the pairwise cosine similarity scores for all movies with the desired movie. For example, when a user inputs ‘The Hobbit: The Battle of the Five Armies’ for recommendations, the model returns the 10 movies with the most similar plots. In this specific example shown in Figure 10, the model recommended other Hobbit movies, Dragon Nest, Lord of the Rings, and Harry Potter. The similarity scores tend to be low because of the sparse matrix with various different words in the plot overviews. However, the recommendations are quite effective.

Movie	Similarity Score
The Hobbit: The Desolation of Smaug	0.365451
The Hobbit: An Unexpected Journey	0.259603
Dragon Nest: Warriors' Dawn	0.138759
The Lord of the Rings: The Return of the King	0.113476
The Lord of the Rings: The Fellowship of the Ring	0.111949
Harry Potter and the Order of the Phoenix	0.094737
The Men Who Stare at Goats	0.088712
Roadside	0.086426
A Time to Kill	0.081974
Mirror Mirror	0.080809

Figure 10: Movie Recommendation for The Hobbit: The Battle of the Five Armies

Conclusion

In conclusion, our analysis was able to answer the key exploratory questions on movie genre, and explore the relationship between movie plot overviews and popularity. One can possibly say that there is a relationship between the plot overviews (text) and popularity. However, the relationship is not too strong and the movie industry is full of uncertainties. That is why we have performed a wide range of analysis ranging from genres to a recommendation system to assist with decision making.

Limitations

A big limitation and challenge was on the prediction of genres based on plot overviews. As mentioned before, movies can have multiple genres, and some of the genres are more common than others. This creates an imbalanced dataset which was a challenge to deal with. In the future, it would be beneficial to gather more movies in the dataset with a wide range of genres, or perform undersampling/oversampling. Also, genres are not as strictly defined as one might think. It would be useful to look at the correlation between genres, and possibly limit the dataset into fewer genres.

For the predictive model to predict the likelihood that a movie will return ‘high revenue’, we only included the plot overview as a predictor variable. However, other features such as actors have the potential to be a powerful predictor.

Finally, further experiment into the TF-IDF vectorizer and the ngram range might help boost the performance of some of the models. For example, in this project, we analyzed uni-grams (single word). In the future, it might be helpful to experiment with bi-grams (two words).

Project Experience Summary

Brian Lee's Accomplishments:

- Performed data pre-processing including converting text into a vectorized form using the TF-IDF vectorizer
- Created data visualizations to better understand the relationship in genres and movie plot overviews
- Developed a multi-label model to predict movie genres
- Developed a predictive model to predict the likelihood that a movie will return high revenue
- Created a movie recommendation system based on similarity scores between plots
- Documented and wrote a report to communicate findings

Ha Thu Nguyen's Accomplishments:

- Parsed JSON strings and converted to Python strings to facilitate data preprocessing and data analysis.
- Built a multi-label machine learning model with a precision score of 50% to predict movie genres based on movie plot.
- Performed statistical data analysis and Chi-square test to prove the influence of movie genres on audiences' rating.
- Used Seaborn data visualization library to create graphs to support data analysis.
- Contributed in communicating the final results through report writing.