CMPT353 PROJECT REPORT

Wikidata, Movies, and Success

GROUP MEMBERS:

YUCHENG CHEN 301286225
LAI WEI 301297816
XINZHOU ZHANG 301297377

Introduction:

For this project, we chose the interesting topic of exploring the definition of success for the movie industry. The field of film production is booming quite lively in recent times with great numbers of movies coming out every year at an increasing rate. Out of this huge cluster of movies of a wide range of genres, some movies can be deemed as more successful than others. However, different criteria will lead to different kinds of success in movies, and the determining factors can consist of complex layers from directors, plots, down to countries of origin. The goal of this project is to use technical methodologies such as statistical analysis to try to establish acceptable criteria for success in movies and to analyze which factor(s) contribute most to a particular movie's overall performance. This will both serve as evaluation and examination for the existing movies and provide future reference/advice for potential film makers.

Data Preparation:

We have 4 json datasets to do the analysis on, which are: wikidata-movies (16000 movie titles each with 14 fields of features), rotten-tomatoes (ratings/proportions of liking from audience and critics), omdb-data (awards won, plot summary, genre), and genres (maps genre WikiData entity identifiers to genre name). After reading these files into the program, we observed that : for wikidata-movies, most values for made_profit in wikidata-movies are missing, only 709 out of 40430 non-null entries, therefore we disregard that field since if we were to only use entries with non-null made_profit field, we would have a much smaller dataset that does not well represent the population therefore could not be used to make meaningful conclusions; for 'rotten-tomatoes', after dropping rows with any NULL entry we were still left with reasonably large amount of data points (16732) so we will proceed with this. Audience average ratings and critic average ratings are out of 5 and 10 respectively, while audience and critique percent who liked it are out of 100, so we scaled each field to standardize them such that they all range from 0-1, since this will produce more accurate result when doing machine learning, obtaining overall score, etc. These are the general data cleaning process; more adjustments were made during the implementation of each specific subtopic.

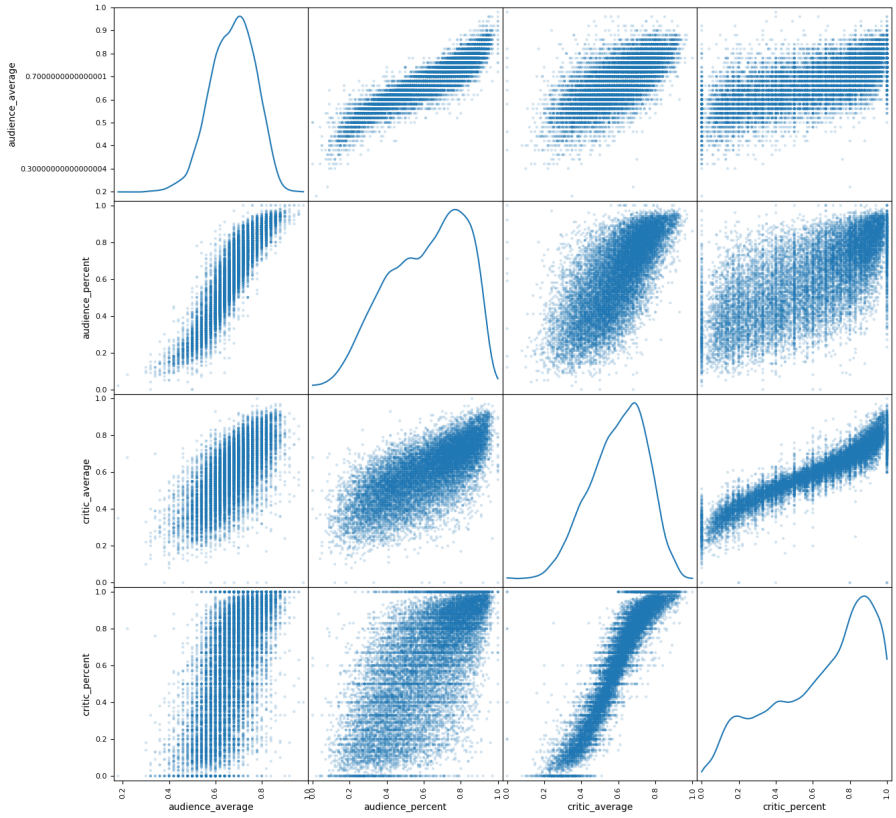
Implementation and Analysis:

Since the dataset sizes are relatively small, we decided to use pandas instead of spark. We explored the following four subtopics and broke the code up to four individual programs so that the runtime for each is within acceptable length.

1. Do the various criteria for success (critic reviews, audience reviews, profit/loss) correlate with each other?

The criteria for success are contained in the omdb-data json file, we deem them to be a combination of audience and critic average ratings and the percentage of the two groups that like the movie. As to the 'number of reviews from audience' aspect, we consider it to reflect the fame or controversy level but not an accurate representation of success of movie, since a bad/unsuccessful movie could also trigger heated discussion and receive countless negative reviews. After necessary data cleaning described above, we first used criteria.describe() to get an overall understanding of the data, and observed that on average, audience tend to give higher ratings than critics, but the percent of two groups that like a certain movie is similar. We then computed the pairwise correlation coefficient for the four criteria and found that correlation is manifest in every pair with lowest correlation coefficient being 0.668439 between audience_average and critic_percent. We proceeded to investigate the two pairs with correlation coefficient greater than 0.9, which are (audience_average and audience_percent) and (critic_average and critic_percent). We produced a scatter plot matrix to get a visual representation and examine the type of relationship between each pair of the factors. From the matrix we observed that aside from those two pairs, all others appear to be a cloud of data points with a positive trend. The two pair of interest appear to have a polynomial relationship, therefore we used polynomial regression feature of machine learning to train and fit using these two pairs. After a bit of parameter tweaking, we found that with degree = 3, accuracy score for the two cases are 0.86 and 0.91 respectively, which is reasonably high. So we suspect the polynomial relationship to exist between (audience_average and audience_percent) and (critic_average and critic_percent) Furthermore, we also suspect that there is probably a linear association between critic and audience average ratings, the corresponding linear regression slope is 1.107,correlation coefficient (r value) is 0.6991, and pvalue for T-test for the slope is <<0.05, so we can suspect reasonably that a positive linear association exists between critic and audience average ratings. However, machine

learning linear regression was used to test for accuracy score and we got 0.493 which is rather low, so we cannot conclude with confidence of that result.



2. What specific factors are related to a movie's success? Which are the most related?

We joined together wikidata and rotten-tomatoes datasets to put the factors and movie success criteria in one dataframe to do analysis on. We only kept movies that have rating counts greater than 3000 to make sure we have enough ratings for the averages to be credible. To create an overall numerical scale for success, we decided to give different weights to specific criterion. Since critics are mostly professionals and therefore tend to give more accurate ratings, we weighted audience_average, audience_percent, critic_average, critic_percent 30%, 10%, 40%, 20% weights respectively, and calculated an overall score out of 100, and then set the standard of being successful as overall score greater than or equal to 80. We added a column for success with ‘success’ or ‘no_success’ values. On to the factor cleaning: we modified ‘original languange’ into two categories:

English(1) and Others(0), since 84% of the movies are in English, and we wanted to see if the language being English has any effect on the success of a certain movie. For 'country of origin', we divided it up into USA(1) and others(0) for the same reason. Then, we cleaned up the data by putting 1 for the movies that are based on books and 0 for the ones that are not; and same for whether a movie is in series to investigate the effects of these two factors. Lastly for directors, we found that the most productive director is Woody Allen with 46 movies made in total, so we categorized the movies to be made by Woody Allen(1) or not(0). We are also interested in whether the publication year and month play a role in determining a movie's success, so we parsed 'publication date' and extracted year and month properties and stored them in two other columns.

Now we have a total of 7 factors: 'based_on', 'series', 'English', 'America', 'year', 'month', 'woody_allen'. We proceeded to use machine learning with RandomForestClassifier to train and test these factors against 'success' or 'no_success'. Random forest fits a number of decision trees classifiers to classify data so it is perfect for our situation. The accuracy score turned out to be 0.8456 which is reasonably high, so this model is satisfactory. We then used feature_importances_ feature of random forest classifier and found the feature ranking below:

1. based_on (0.709492)
2. series (0.155407)
3. English (0.063000)
4. America (0.030567)
5. year (0.025550)
6. month (0.013135)
7. woody_allen (0.002848)

We can see that the most related factor is based_on, this makes sense since movies that are based on books tend to be more successful such as 'Harry Potter', since they will have a larger fan base consisting of audiences that are already fans of the book that they are based on.

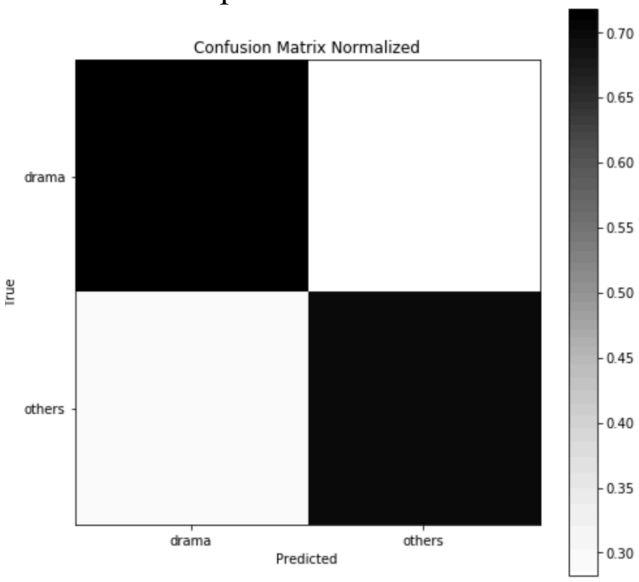
3. Does plot summary predict success in any useful way?

We employed the same procedure as the second question and added the column 'success' with entries 'success' or 'no_successes calculated using the weighted factors. After joing the 'plot_summary' in omdb dataset and 'success' in rotten-tomatoes dataset, we used

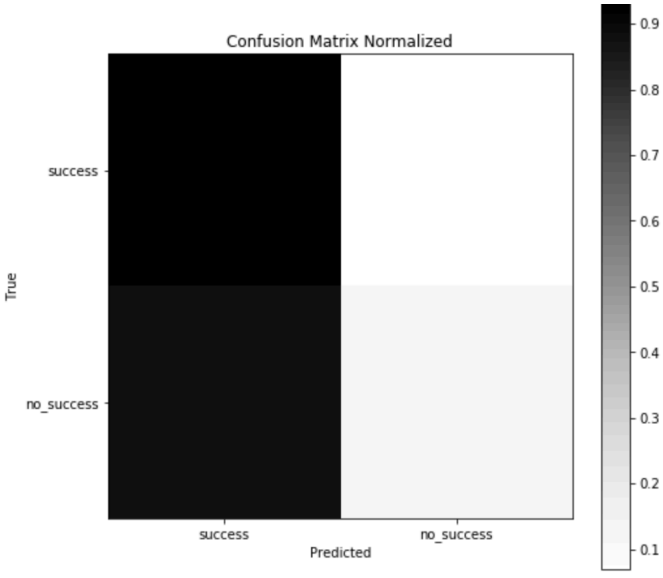
Machine Learning with Text TF-IDF. We first used TfidfVectorizer to fit and transform the plot_summary since it is text-based, then used multinomial Naïve Baynes to train the model. We found the accuracy score to be around 0.83 which is reasonably high. We then switched to a different method of transforming the text-based plot_summary which is CountVectorizer, the accuracy score is slightly lower at around 81%. Finally, we used GridSearchCV to find the best parameters for both model, and the results are: for TfidfVectorizer, the best parameters for MultinomialNB is 'alpha': 0.2, 'fit_prior': True; and for CountVectorizer, the best parameters for MultinomialNB is 'alpha': 1, 'fit_prior': True. With these changes of best-fit parameters, accuracy scores improve to 84% and 85% respectively. Therefore, we can conclude that we are able to predict the success of movie based on the factors.

Another aspect we explored is predicting the genre of movie based on its plot summary. (See Q3-ipynb) In particular, we simplified the problem to predicting whether the genre is drama or not based on plot summary. The procedure is similar as above but with y variable in machine learning as ‘drama’ or ‘other’ instead of ‘success’ or ‘no_success’. The accuracy score turned out to be around 0.71 before grid search for best parameters and 0.75 after for both methods of transforming text data.

On another note, we used normalized confusion matrix to evaluate the accuracy of these predictions. As can be seen below, the confusion matrix looks promising for predicting the genre, however not as good for predicting success. This may have to do with how the standard of success is determined and may need further specification so that it relates more to the plot.



Confusion matrix for predicting genre



Confusion matrix for predicting success

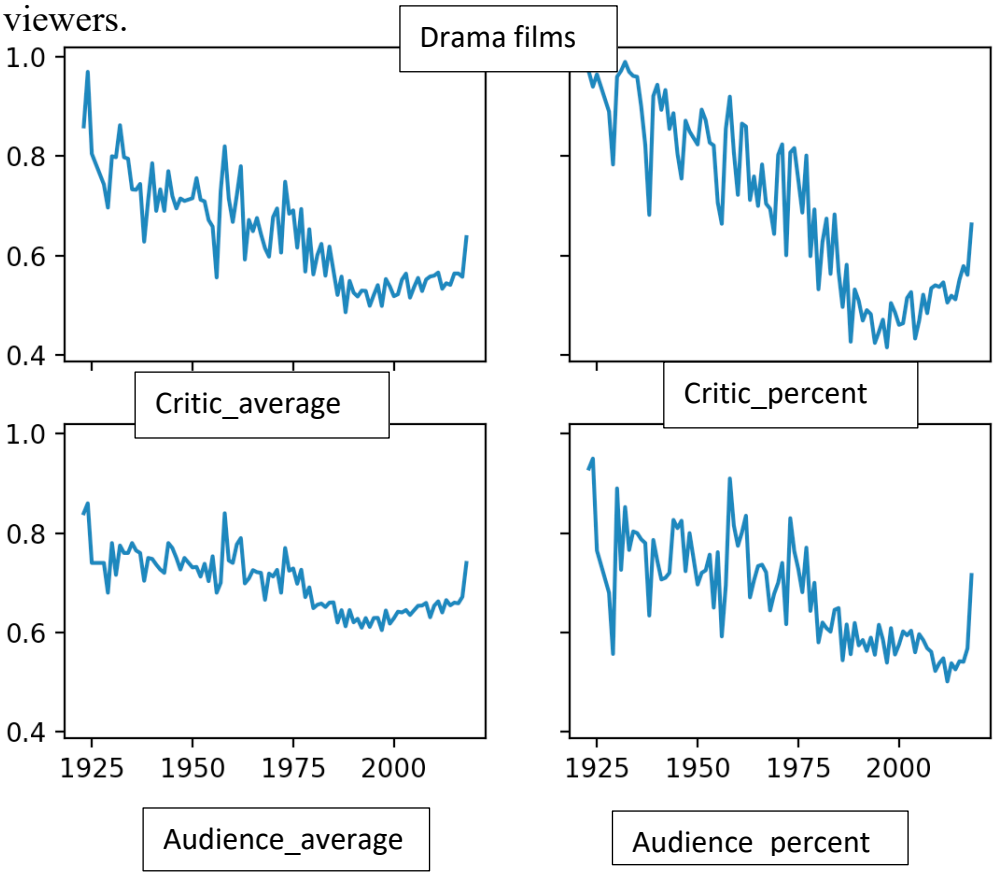
4. Have any of these things changed over time (depending on the movie's release date)?

For this aspect, we investigated whether people's ratings/likings for specific types of movies have changed over time, as well as the popularity/controversy level. For the data used to do analysis, we joined the wikidata-movies and rotten-tomatoes datasets on imbd_id. After cleaning and scaling data as described in the first part, we kept only columns of interest which are genre, publication dates, and the potentially changing factors (audience_average, audience_percent, critic_average, critic_percent, audience_ratings). Reading the json file did not automatically parse the publication_dates, so we converted the string to datetime object manually before we could apply timestamp function on them to get numerical values of time to do regression on. The main part is, we want to choose the most popular genre of movies to investigate whether there has been a change in its popularity/ratings. However, there are often multiple genres labelled for a movie, so we decided that if a movie belongs more than one genre, then it is an instance of each genre that it possesses. To implement this and separate movies into its genre subgroups, we used the following procedure. First, for easy analysis, we mapped 'genre' from wikidata id to its English genre name: we added a column to store the number of genres a movie has (size of the list) and found that maximum number is 11. So we created 11 extra columns and used. values.tolist() to split the list up and makes sure each genre of a movie is stored to a column. Then we used join function repeatedly with genre dataset on wikidata_id for genre to add another 11 columns that contain the corresponding English names. Lastly, we dropped the 11 columns with wikidata_id and were left with the straight-forward English genre names instead. After counting the number of instances of each genre, we found that the following 12 genres are most popular:

drama film	**6272**
comedy film	**2968**
action film	**1762**
horror film	**1660**
documentary film	**1340**
crime film	**1334**
thriller film	**1304**
film based on literature	**1263**
comedy-drama	**1241**
romantic comedy	**1185**
science fiction film	**1171**
LGBT-related film.	**1088**

Out of these 12 genres, we excluded films based on literature and comedy-drama because, we feel that 'films based on literature' is a quite general expression and can be applied too broadly therefore not a genre we are interested in, and comedy-drama is just a combination of comedy and drama therefore can be considered as an instance of comedy and drama each. So for the remaining 10 genres, we separated the movies into groups based on if they have a certain genre in their genre list, and plotted each factor against publication date to get a visual of the general trend. We observed that data points are generally way more in number also scatter more widely in recent times than before, and negative trend is obvious in several movie genres. To confirm our suspicion, we did linear regression on all 10 genres for audience and critic review as well as percentage of liking, and found that all slopes are negative with some significant and some not. However, the slopes are quite close to 0. This is not satisfactory, so we extracted publication year feature and grouped by year to calculate ratings over each year and did regression again, the result is slightly better, and the downward trend is way more obvious, and example plot is shown below.

From the plot of count of reviews against publication date, we can easily tell that all genres are getting popular with more reviews in recent times. This makes sense since movie industry is indeed booming and attracting more and more viewers.



audience_average

audience_percent

Conclusions:

From our finding during implementation, we can derive the following conclusions:

The criteria for success (audience and critic average ratings and the percentage of the two groups that like the movie) are highly correlated, with (audience_average and audience_percent) and (critic_average and critic_percent) having the highest correlation coefficient and possibly polynomial relationship with degree 3.

We found that whether a movie is based on books relates most to it success, whether it is in a series determines a lot too, since successful movies tend to be more popular and therefore tend to have sequels produced. The other factors in order of importance are: original language being English, country of origin being USA, and year, month of publication.

Furthermore, we can somewhat predict the success of movies based on plot summary, but we can predict more accurately the genre being drama or not from plot summary.

Lastly, we found that in general, people's ratings for genres of movies have a downward trend over time, with higher ratings for older movies and lower ratings towards modern time. What's more, people are more and more active in giving reviews resulting in way more reviews for more recent movies.

Limitations:

1. Missing values for profit-loss feature.

Most values for profit-loss feature is missing and the actual entries are 0 or 1 indicating either a movie is making profit or not. If we had more time, we could find other sources for movie net profit/deficit and extract the data to fill in the missing values, even perhaps calculate the exact profit made and use it as another criteria for success in movie to produce a more accurate result.

2. Dealing with multilabel features.

For a lot of features such as cast member and genres, one movie often possesses more than one entry (as a list). We had difficulty finding a good way to manipulate the data so that all entries can contribute to predicting the success of movie, etc.

3. prediction accuracy.

When making prediction of success based on plot summary, the confusion matrix shows that accuracy is not satisfactory. In the future we should gather more criteria for success that are more related to the movie plot and predict from it to make more accurate predictions.

Project Experience Summaries

Yucheng Chen:

Wikidata, Movies and Success – CMPT 353 May – Aug 2018

- Initial general data cleaning and manipulating the data for easy analysis
- Collaborating, brainstorming with two teammates about how to implement the data science to find answers for questions, as well as coming up with the overall structure of program.
- Utilize pandas learned in class to find out correlation among criteria for movie success
- Utilize pandas to come up with procedure to separate data into groups based on genre and do analysis on the potentially changing factors.
- Use SFU gitlab for documentation and version control
- Overviewing and putting together work of teammates to make overall analysis
- Fixing bugs and adding necessary comments for the programs as well as some final adjustment
- Initial drafting and formal write up of the project report

Lai Wei:

Wikidata, Movies and Success – CMPT 353 May – Aug 2018

- Collaborate with two teammates to build a project that investigates the relationship between movies' data and success. By analyzing the data of movies, we found out what specific factors are related to the success and what are not.
- Movies' data is extracted from WikiData, Rotten Tomatoes and OMDb API.
- Python, Jupyter Notebook, Pandas, Numpy, sklearn Machine Learning models, hypothesis tests were the main techniques used for the project implementation.
- Use SFU Gitlab for documentation and version control.
- I used RandomForestClassifier to do multilabel classification for the features of the movies, and I ranked the features by their importance to see which are the most related.
- I applied both Tfidfvectorizer and Word Count Vectorizer to the plot summaries of the movies to do natural language processing and explored if it is possible to predict a movie's success from its plot summary.
- I proofread the report and gave some opinions how to improve it

Xinzhou Zhang :

Wikidata, Movies and Success – CMPT 353 May – Aug 2018

- Get a team work with two teammates to build a project to investigate and analyze the relationship between movie's and success using SFU Gitlab for documentation.
- By looking at review scores in the rotton_tomatoes file, I analyze the data between each variable to see whether there is an exact linear or non-linear relationship or not using Machine Learning Method.
- Develop a model and program to predict the regression line using stats. method linear regression method and machine learning linear and polynomial method.
- Figure out which two variables are linearly correlate and polynomial correlate by plotting the scatter-plot between all of variables and finding what types of cluster the variables have.
- Fit the best linear regression line on a positive cluster plot and degree of polynomial regression to give a best fitted polynomial regression line.