Analyzing Movie Ratings and Genre Popularity Over Time

Authors: Ofe Fonseca, Ryan McCormick, Tony Phan

Introduction

Over the years there have been thousands of movies each belonging to one or more genres. These genres come in all forms (action, adventure, horror, etc.) so that anyone will be able to find the perfect movie just for them. Despite the sheer number of genres to choose from, however, it can be difficult to find the better (most viewed or most rated) movies among the rest as the pool has been flooded with films that try too hard to encapsulate each one. From the occasional movie goer to even the most ecstatic film connoisseur, this over saturation of movies can cause a dislike towards a certain genre as it feels that it has lost its allure. This of course is a problem for those in the business of making movies as it causes people to feel less compelled to go and see a new movie.

With this problem at hand, the goal of this project is to familiarize ourselves with a large dataset and perform analysis to identify if there are any patterns related to popularity, ratings, and genres in the movie industry. By using a large dataset containing movie data from the past century, we were able to see how the creations and popularity of movies has changed over time. As a Big Data problem, being able to identify what movie genres have thrived over the years can help movie producers and film specialists create genres that fit viewers' preferences better or see what trends have ensued in the film industry.

Methodology

In order to solve this problem we first needed to decide what language and framework to use for the pre-processing of the dataset. Our top choices were utilizing MapReduce with Java or using Apache Spark with Scala. After looking into both of these options, we decided that Scala would provide the best functionality for how we planned to traverse and break down our data. With this in mind we also chose to use Apache Spark as our framework as we felt we had a better understanding of how our chosen language interacted with it.

Using Spark meant that there were a few things we needed to do prior to working on the dataset. This included setting up our master and worker nodes so we could split the workload between them all for faster processing. We also needed to set our ports correctly so we could access the web-ui on the lab computers. After doing this we were able to plan out the steps we would need to take in order to do our pre-processing. We used methods such as map(), reduceByKey(), and join() in combination with string formatting to create our RDDs (read-only collection of records).

To begin processing our data we wanted to first look at the ratings of every movie in our dataset. After looking at the ratings.csv file from our dataset we saw that the file contained the four columns userId, movieId, rating, and timestamp. The userId and timestamp columns were not necessary to collect the ratings so we first needed to remove these from the file and then save what remained as a new file called simpleRatings.csv. To do this we read our ratings.csv file into our program. From here we mapped each line of our file to a new RDD and removed the unnecessary columns. Next we mapped this new RDD in a way that re-ordered it into two separate columns, one containing the movieId's and the other containing the rating associated with the movieId. Finally we saved this as a new text file called simpleRatings.csv.

After creating the simpleRatings file we were able to move onto the next step in processing which was to combine this file with our movies.csv file which contains three columns that correspond to movieId, title, and genres. First we had to read in our simpleRatings.csv file and our movies.csv file. By mapping the simpleRatings.csv file we were able to find the average rating per movieId and store it into a new RDD. Next we converted the movies.csv into an RDD and joined it with our average ratings. Finally we formatted this RDD back into csv format and saved it as a new file called movieRatings.csv.

Our last step for the project was to analyze the data we found in our movieRatings.csv. To do this we used Jupyter notebook, a web-based computing platform that utilizes python. In our notebook we used Pandas to create a dataframe, NumPy, and Matplotlib for plotting the diagrams used to perform our analysis.

DataSet

"MovieLens 25M Dataset" is the title of the dataset being used and analyzed in this project. The data was acquired by GroupLens and is said to be approximately: 25 million ratings from 162 thousand users about 62 thousand movies(Movielens). The dataset is downloadable as a zip archive of size 262 megabytes conveyed between 6 CSV files. The actual size of the dataset is estimated at around 1.2 gigabytes. All the files in this dataset have their own set of attributes, some linking certain data together. In this project, the main set of attributes used will be movieID, rating, title, and genres. Keeping that in mind, the attributes used in the analysis of the data will be retained from only two of the six files in the dataset. Both ratings.csv and movies.csv were joined together and simplified to create one file with all necessary components.

Dataset content

Ratings.csv is the largest file out of the six and is one of the two used in the processing of the data. This file contains all 25 million ratings gathered in the format: userId, movieId, rating, timestamp. UserID is a unique number given to every user to identify them. MovieID is the unique number given to the movie. Rating is the rating given by the user and timestamp is the time the action was performed.

Movies.csv is the other file used in the project. This file is a list of all the movies from whom ratings were acquired. The file is formatted: movieId, title, genres. The title column contains the name of the movie and the genres column has a list of all the genres the movie belongs to.

Analysis

With many possible questions arising from the dataset chosen in the project. The best solution found was to pick a couple questions to analyze and record the findings. Having said that, all questions relate to one another. Analysis 1 looks at the amounts of movies created per year. Analysis 2 focuses on rating scores and the number of users rating the movie. Analysis 3 finds the most popular genre with certain characteristics. The final analysis will focus on the rating score and the length of the title.

Analysis 1: Number of Movies Released By Year

Most of our analysis focuses on how the movie industry has changed over the last century. With this, the first analysis we focused on was the number of movies released per year to see if there were any significant discoveries. At this point, we have already preprocessed and normalized the dataset so most of this analysis focuses on visually seeing and displaying the data. From the visualization we created, we noticed that the number of movies released per year was linearly increasing from 1940 to 1980 (slowly more movies were created over time). This was expected since entertainment was slowly gaining popularity during this time period and the demand for movies was increasing. To our surprise, however, in the most recent years (1980-2019) there has been a significant exponential increase in the production of movies.

This exponential growth can be attributed to most movie directors targeting and appealing to younger audiences. Likewise, around 1980-1990 was the invention of videocassette recorder(VCR) which also added to the popularity and demand of the production of movies. With this increase in recent years, however, another interesting result we found was there was actually a decrease from 2016-2019 in the number of movies created. Unfortunately we can only assume this change can be attributed to the data still being gathered for the more recent years but there may be some connection to China producing less movies for these years.

Analysis 2: Top Rated and Most Popular Movies

The next analysis we wanted to look at was which movies were considered the top movies of all time. With this, however, a challenge we ran into was defining what would qualify the movies to be better than the others. If we were to consider the top movies based on purely average ratings alone, we found that there were 828 movies with a 5.0 average rating (a perfect score). Of these 828 movies, all of them contained only a few votes(1-2) from users which is not sufficient to declare these movies as the best movies.

Since declaring top movies from average ratings alone had too many outliers and biases, we shifted towards looking at the most popular movies over the dataset. When considering the dataset, users were given the freedom to choose whatever movie they decided. Given this freedom, the number of votes a movie received was a good indicator of the most popular movies. While we could declare the top movies based purely on popularity, we wanted to delve one step further and give equal opportunity to less popular movies when considering the top movies in the dataset. With this, we graphed the relationship between popularity and average rating and saw that all the movies above 4.2 rating only had a couple votes associated with them (indicating these were outliers).

Using this information, we discovered that one of the most fair ways to declare the top movies in the dataset would be to find a balance between number of votes and the average rating. Ultimately, we decided to use 3 different filtering values for the minimum number of votes and compute the top movies by rating after the filter. The filters we decided to use were 425 (the mean number of votes), 2,000 (a value that we thought would filter out the outliers), and 70,000 (there are only 5 movies--these are the most popular movies). Through these filters, we discovered that the top k movies changed significantly when setting the considering the top rated movies with a minimum number of votes.

With this discovery, we concluded that deciding which movies were considered the top movies in this dataset was a matter of how we decided to split the data--yet there were still some interesting takeaways/conclusions: 1) Despite the changing top k movies by filter, there were some movies that appeared in two/three of the filters and these could be considered the top movies. 2) When using the higher filter values(indicating more votes for the movies), the top k movies were all created before the year 2000. While this could indicate that the best movies were created prior to the year 2000, it could also indicate a limitation of this dataset. Expanding on this, data and user reviews may still be getting generated for the most recent years (which may explain the odd occurrence in Analysis 1). Likewise, MovieLens has multiple movie datasets of different sizes and years of publication. This means that the dataset we used (25M version) could have contained data from previously generated datasets(all of which were created at different years). In other words, this analysis will not be able to generalize and conclude the best movie of all time(but rather only the best movie in the dataset) since the majority of this data could have been obtained before many of recent movies were even released (not giving equal chances to be the top movie). 3) Some genres appeared more frequently than others in the top k movies generated (this will be explored in the next analysis).

Analysis 3: Most Popular Genres (by year):

After completing Analysis 2, we discovered that some movie genres appeared more frequently than others when looking at the top movies by rating and popularity. With this, we decided to first see the distribution of genres in the dataset. Before we could see the distribution, however, we realized that some movies had multiple genres associated with them and others only had one. This meant that we had to preprocess the data one more time and split up the genres (this does indicate a limitation of this analysis since one movie could contribute to more than one genre). When the preprocessing was done and we saw the distribution of the entire dataset, we discovered that comedy and drama movies contained 2-3 times more movies than the other genres.

Through this discovery, we realized that it would be unfair to decide the most popular genre based only on the number of movies (just like in Analysis 2). This being said, we decided to use a similar approach from the previous analysis and apply filters to minimize the outliers and give equal opportunity to every genre. Likewise, we decided to focus our analysis to find the most popular genre for each year (1980-2019). The approach/filter we decided to use was first to obtain the top 20 movies for each year and calculate the genres that appeared the most. By only considering the top 20 as opposed to all the movies, we are essentially removing the bias (more comedy and drama movies created than other genres) and giving each genre a fair chance.

Without filtering only the top 20 movies, it should come to no surprise that the top movies for 1980-2019 consisted mainly of Comedy and Drama (due to the bias). With the filter, however, the most popular genres remained similar from 1980-2000 but varied significantly from 2000-2019. In these years, Comedy had fallen and Action and Adventure rose up to take its place (though Drama still remained popular). This discovery was significant because it indicated a shift in the popularity of movie genres, which could be tied to an external factor(introduction of IMAX, special effects, etc.). With this shift, we also discovered that Action has consistently dominated the movie industry (by popularity) in the most recent years (2015-2019) and it would not be far fetched to predict that the most popular genres in 2020 and 2021 would also be Action.

Analysis 4: Does the length of a title influence a movie's rating score?

Movie titles are the first impression of a movie. Most of the time titles either leave too little or too much for the imagination. With this, we wanted to explore if a prediction can be made about the success of a movie by using its title. In other words, we wanted to see if the title length had any correlation between the movie ratings and popularity. To accomplish this analysis a certain filter had to be established. Some of the titles in the dataset contained more than just the title of the movie; it also contains the year and other names the movie had in parenthesis. So to produce a clear visual picture, the definition of a title will be the title not enclosed in the parenthesis. The length will then be the size of the string. Just like in the other analysis we will be using the normalized data.

The analysis starts with calculating the length of the titles and inputting them in a column called “title_L". This newly added column will be the one examined throughout the analysis. The correlation between the length of the title and the average rating is then taken. The first time the correlation will be computed with no filters being set and therefore the result seems a little far-fetched. From this, we obtained a correlation value of 0.033 (meaning as the title length increases the ratings received will also increase). Despite a positive correlation, when we generated a visual graph it discredited the correlation found. The graph showed a centralized cluster of data but it was not located where the correlation implied it will be at. Instead, the cluster was found in the middle meaning the large number of outliers in the dataset have skewed the results. To better understand the problem a box plot was created and filters were applied to the dataset. After the filters, the entries had to have a title length of 40 or below, at least 425 votes, and a rating between 1.5 and 4.25. With these filters, the data was more centralized (but there were still too many ratings to look at for every length).

The next step of the analysis would be to combine all the ratings with the same length together and average them out. This task was performed because the movies were grouped by the length of their titles and only one rating was required per group. After these steps, we once again obtained the correlation value and created a graphical implementation. The new correlation we discovered was -0.033 which matched the graph produced and showed a negative parabola with a maximum at length 10 and 14. The analysis showed that shorter titles had higher ratings than longer titles. Finally, we conducted the analysis again but decided to include the full title (excluding the whitespace and the year). The analysis followed the same process as before and once again displayed that shorter titles had higher ratings than longer titles.

There are some limitations to the analysis such as the amount of data being used after the reshaping. As well as the fact that not everyone rates the movies they see. The last limitation would be the definition of what is considered a title. All these influence the finding yet they do not change the fact that there is a negative correlation between ratings and the title length.

Project Contributions

For this project we decided to have individual tasks and shared tasks where everyone contributed equally. The following is a list of shared tasks with a brief description of the contributions.

Shared Contributions:

1) Project Proposal: Each team member contributed to the initial idea and project proposal portion of this assignment.
2) Final Project Report: This report was worked on by every team member and everyone had their own section to complete (though editing was a group effort).
3) Presentation: The presentation slides will be completed by all team members.

Ofe Fonseca:

1) Perform Analysis 4. Examine correlation between title length(official title) and ratings received. Also extended the analysis to include the correlation between title length(every single character except the white space and year) and ratings.

Ryan McCormick:

1) Removed unnecessary columns from ratings.csv. The result was a simpleRatings.csv file which only contained movieID and a rating.
2) Computed the Average rating for each movie and created the "votes" column (seen in movieRatings.csv file).
3) Joined the ratings containing the averages (Task 2) with the movies.csv file. Joining these two files would result in the main file we use (AverageRatings.csv).

Tony Phan:

1) Combined the data within the simpleRatings.csv (created by Task 1 by McCormick). The data at this point still contained individual ratings for movies so it was necessary to combine by key (movieID) to compute the average.
2) Get the year the movie was released (from the title) and create the "year" column. This task was needed to simplify analysis.
3) Normalize and remove outliers in the dataset. Some of the movie titles in our dataset contained many special characters (commas, semicolons, etc.) which made loading and analyzing the data difficult. Some of the movies also did not contain a year of release within the title and needed to be filtered out. This task was aimed towards cleaning the data so it can be used in a Dataframe.
4) Performed Analysis 1, Analysis 2, and Analysis 3. With the preprocessing and normalization of the data completed, some analysis was conducted to better understand the data. Analysis 1 viewing the number of movies created by year. Analysis 2 dove into the average rating for each movie and checked the most popular movies. Analysis 3

broke up the genres into their own sections and displayed the distribution and popularity of movie genres over the last few years.

Bibliography

Movielens 25M dataset. GroupLens. (2021, March 2). Retrieved October 21, 2021, from https://grouplens.org/datasets/movielens/25m/.

Hayes, Adam. “How Should I Interpret a Negative Correlation?” Investopedia, Investopedia, 2 Dec. 2021, https://www.investopedia.com/ask/answers/040815/how-should-i-interpret-negative-correlation.asp.

Anon. 2012. The History of Movies. (2012). Retrieved December 3, 2021 from https://saylordotorg.github.io/text_understanding-media-and-culture-an-introduction-to-mass-communication/s11-01-the-history-of-movies.html

Brent Lang. 2017. Global box office hits record $38.6 billion in 2016 even as China slows down. (March 2017). Retrieved December 3, 2021 from https://variety.com/2017/film/news/box-office-record-china-1202013961/