SI601 Individual Project Report

Xiaoyue (Platina) Liu 28589009 xiaoyliu@umich.edu

Motivation

Netflix is a global provider of streaming movies and TV series, and now has over 75 million subscribers. There are lot of online reviews from Netflix customers. The Internet Movie Database (abbreviated IMDb) is an online database of information related to films, television programs and video games, including cast, production crew, fictional characters, biographies, plot summaries, trivia and reviews. There is already average rating data for each movie on a ten-point scale. My project is to combine and compare the two ratings and summarize the rating in each genre for each country. So that I can find which country produces the most highly rated movies in which genre. And also I can see how movie ratings change along the time from 1914 to 2005.

Datasets

My first dataset is the Netflix Prize Data Set from 2009:

http://academictorrents.com/details/9b13183dc4d60676b773c9e2cd6de5e5542cee9a

The movie rating files contain over 100 million ratings from 480 thousand randomly-chosen, anonymous Netflix customers over 17 thousand movie titles. The data were collected between October, 1998 and December, 2005 and reflect the distribution of all ratings received during this period.

In this dataset, there is one file named movie_titles.txt, which contains movie titles and year with an assigned movie id range from 1 to 17770 sequentially.

Movie information in "movie_titles.txt" is in the following format:

MovieID,YearOfRelease,Title

- MovieID do not correspond to actual Netflix movie ids or IMDB movie ids.

- YearOfRelease can range from 1890 to 2005 and may correspond to the release of

  corresponding DVD, not necessarily its theaterical release.

- Title is the Netflix movie title and may not correspond to titles used on other sites. Titles are in English.

There is another folder with files containing the corresponding review information for each movie id. In each of the files, the reviews are in the following format:

CustomerID,Rating,Date

- MovieIDs range from 1 to 17770 sequentially.

- CustomerIDs range from 1 to 2649429, with gaps. There are 480189 users.

- Ratings are on a five star (integral) scale from 1 to 5.

- Dates have the format YYYY-MM-DD.

My second dataset is obtained from an API which I can send request for movie information using movie titles and years. http://www.omdbapi.com/ You can get response in the data format of json object.

Example is as following:

http://www.omdbapi.com/?t=Wall+E&y=2008&r=json

{"Title":"WALL·E","Year":"2008","Rated":"G","Released":"27 Jun 2008","Runtime":"98 min","Genre":"Animation, Adventure, Family","Director":"Andrew Stanton","Writer":"Andrew Stanton (original story by), Pete Docter (original story by), Andrew Stanton (screenplay), Jim Reardon (screenplay)","Actors":"Ben Burtt, Elissa Knight, Jeff Garlin, Fred Willard","Plot":"In the distant future, a small waste-collecting robot inadvertently embarks on a space journey that will ultimately decide the fate of mankind.","Language":"English","Country":"USA","Awards":"Won 1 Oscar. Another 81 wins & 78 nominations.","Poster":"http://ia.media-imdb.com/images/M/MV5BMTczOTA3MzY2N15BMl5BanBnXkFtZTcwOTYwNjE2MQ@@._V1_SX300.jpg","Metascore":"94","imdbRating":"8.4","imdbVotes":"674,274","imdbID":"tt0910970","Type":"movie","Response":"True"}

Where you can change the xxx in “t=xxx” to movie titles. And xxx in “y=xxx” to year of release.

Using this API, I got the dataset containing title, year, rating, country, genre information for each movie from Netflix.

There are some movies from Netflix which we can’t find in IMDB, request for them will return empty json object. For those movies, what I did is just put N/A on the fields other than title and year. There are 10521 valid records retrieved.

In this part, sometimes because of the bad Internet connection, we have to stop between two requests. I set this time period to 5 seconds. It took about 40 minutes to get all the data for each movie.

Procedure

Data manipulation for Netflix dataset.

For each of the review files, the first line is movie id followed by a colon. We will extract movie id from the first line. For the following lines, each line is in CustomerID,Rating,Date format. As we only care about the rating, we parse the line into a tuple with three elements. For each file, I wrote a function **get_rating(filename)** to process it returning only the movie id and the average rating in the format of tuple.

For all the review files in the folder, using the function get_rating, we got the result as a list of tuples containing movie ids and corresponding average ratings.

Then we have to match the ratings with movie information in movie_titles.txt. Here I use SQL.

To use SQL to manipulate the data, we have to use Unicode, for those titles with special characters other than English, we have to decode first. For example, Café must be decoded according to latin1.

First, I created two tables, one for id_rating and one for movie_titles. Then I join the two tables based on the movie ids.

Then we get what we need from the dataset one: movie title, year of release and average Netflix rating.

Data manipulation for IMDB dataset.

For the IMDB dataset, using movie title and release year to send request for the information of each movie, in return we get a json object. After that we can load the json object into a dictionary and extract the genre, country and IMDBRating.

There are some movies from Netflix which we can't find in IMDB, request for them will return empty json object. For those movies, what I did is just put N/A on the fields other than title and year to represent missing values.

What's more, to search by title in the API, first divide the title into words then connect by "+" and place it in the "t=" area in the API request URL. Here we also have to consider the encoding for special characters.

Here we have to make the program to tolerate errors caused by Internet connection and empty response. I used the "try except" sentences.

Join the datasets

Then it's time to join the datasets. I chose to use SQL again.

To use SQL to manipulate the data, we have to use Unicode, for those titles with special characters other than English, we have to decode first. For example, Café must be decoded according to latin1.

First, I created two tables, one for Netflix and one for IMDB. Then I join the two tables based on the movie titles.

Here we get what we want: imdbid, title, genre, country, imdbrating, netflixrating.

One movie can be with more than one genres or countries, so we have multiple lines for one movie with distinct combination of genre and country.

Workflow

Part one: generate average rating for each movie from Netflix

Part two: get movie information using API

Part three: Join datasets, compare and summarize.

Challenges

Unstable Internet connection made program to stop and lose data. I used "try except" to bypass the errors. And sleep 5 seconds to wait for next request if encounter connection error.

Also some of the elements in the datasets are containing "\t" or "\n" sometimes, so that when joining the datasets, they didn't match. I use string.strip() before putting the lists into SQL.

Analysis and Results

For the analysis, I mainly use SQL to group and summarize the combined data set.

By sorting the average ratings from Netflix and IMDB and their difference, I got the results below.

1) Based on the average of Netflix ratings and IMDB ratings, we can find the top 10 rated movies, which are:

Movie	ratings((imrating+2*nfrating)/2)
Band of Brothers	9.31
The Godfather	9.1
Firefly	9.0
Lord of the Rings: The Return of the King	9.0
Star Wars: Episode V: The Empire Strikes Back	8.94
The Beatles Anthology	8.93
Schindler's List	8.91
The Godfather,Part II	8.9
Pride and Prejudice	8.89
Star Wars: Episode IV: A New Hope	8.85

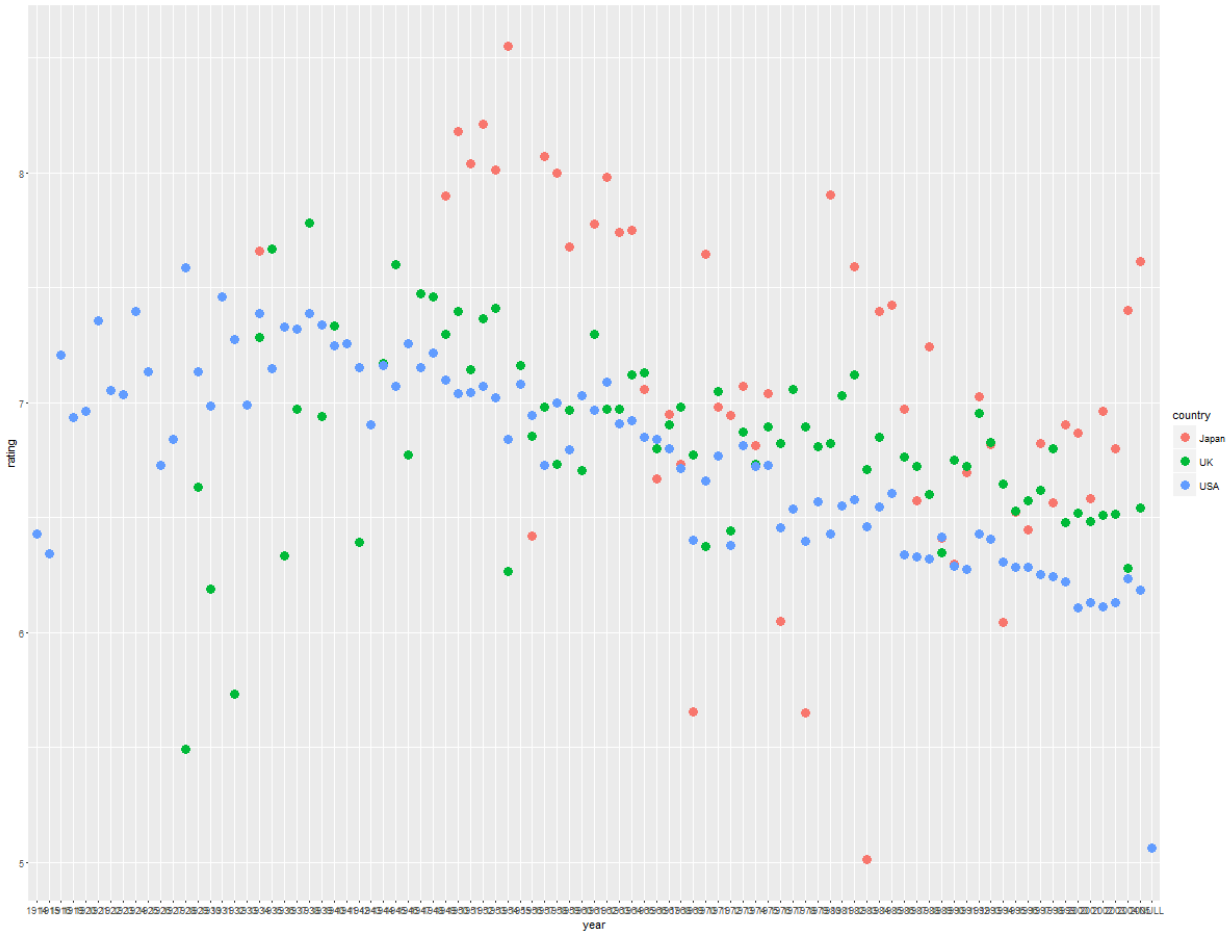
2) And also we find the top 10 most highly rated combination of country and genre (some combination have too few movies to compare, here we choose those combination which have more than 100 movies), which are:

country	genre	number	average rating((imrating+2*nfrating)/2)
UK	Documentary	135	7.29407407407
Japan	Animation	179	7.09508379888
USA	History	231	7.0874025974
UK	Biography	111	7.04162162162
Japan	Drama	201	6.9815920398
USA	War	217	6.96211981567
USA	Documentary	680	6.94752941176
USA	Biography	346	6.93384393064
USA	Music	448	6.93241071429
Japan	Adventure	104	6.90884615385

3) We can also find the movies with largest difference in the two ratings, which are:

Movie	rating_diff(ABS(imrating-2*nfrating)
The Vault	4.62
Freaks & Geeks: The Complete Series	4.3
The Who	4.22
Salsa & Merengue: Cal Pozo's Learn to Dance in Minutes	4.1
Ben & Arthur	3.96
Sasquatch	3.86
The Life	3.86
Superbabies: Baby Geniuses 2	3.8
Meet the Browns	3.76
Fractured Flickers	3.74

4) Finally let us have a look at how ratings for each country change along the years from 1914 to 2005. We pick Japan, UK and USA as example. I used ggplot in R to produce this plot.



There are some missing years for the movie rates for each country, we can see from the plot above.

Interesting findings from the plot

We can see that Japan movies were more highly rated back in the 1950s and the ratings changed a lot along time, while movies from UK and USA gained more stable ratings. As there is a trend of decrease in ratings, maybe people nowadays rate more strictly for movies as there are so many movies produced.