Xiaoli Zhu, Yuchen Zhou, Zhe Chen

SI 206 Final Project Report

April 25, 2022

1. **The goals for your project**

   -Explore the relationship between the age of a film and its popularity in modern society
   -Create visualization based on data calculated

2. **The goals that were achieved**

   We get 4 charts based on the data we collected. It shows the difference between scores for movies in 4 periods from 1925 to 2025 (25 years as a period). We also look into the connection between the top 250 rated movies and the most popular movies. By presenting our result, our group make 5 different forms of visualizations to give a clear picture.

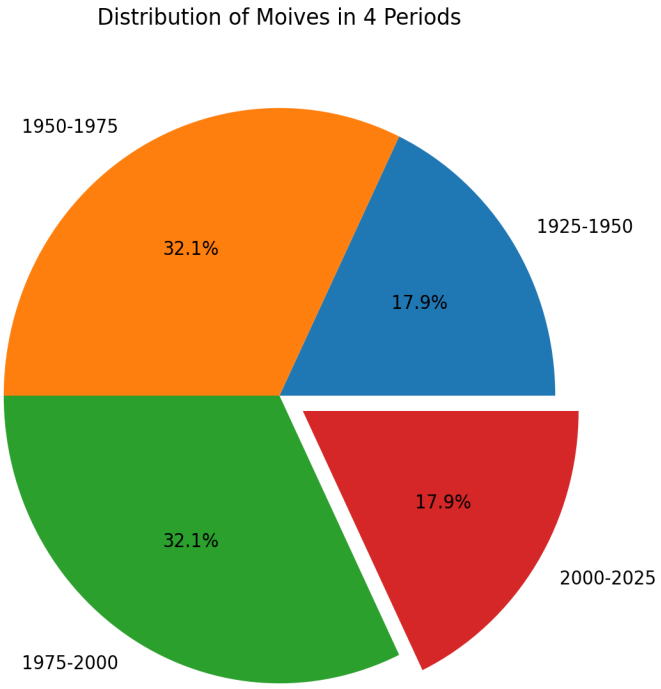
3. **The problems that you faced**

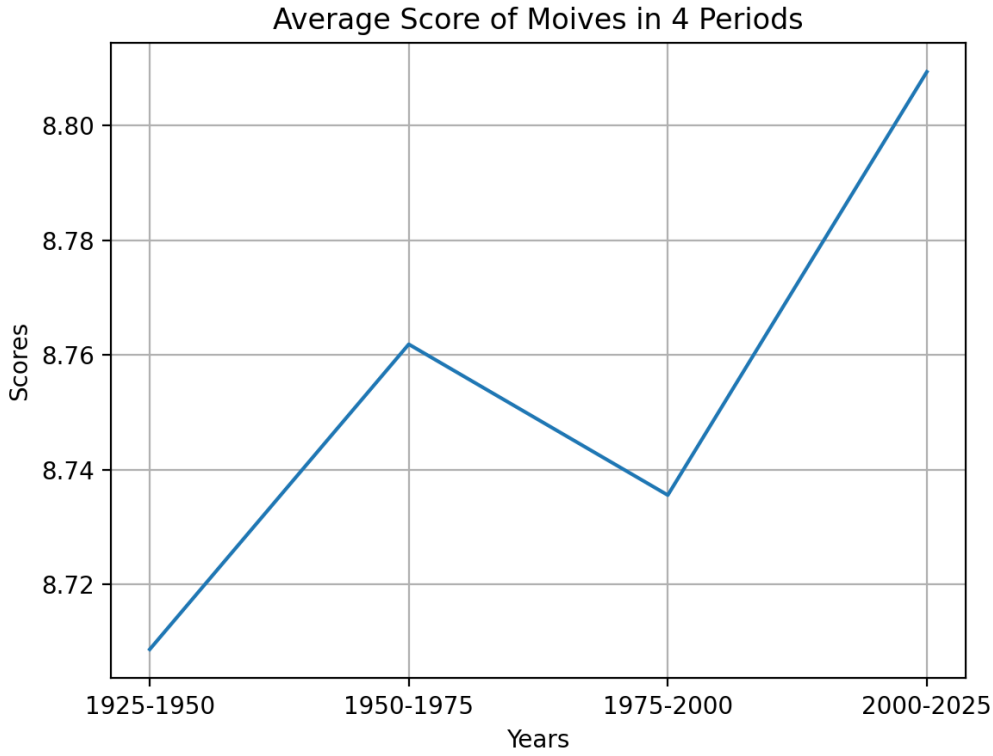
   At first, we don't know what genre we want to analyze. Then, we have trouble accessing the API of websites, such as rottentomato.com. We didn't get any response from the company after we sent out an email asking for API key requests. Therefore, I started to loop up a different one and found TMDB as another resource. In the process of analyzing data, we don't know how to calculate the row average and create a new column to add them using SQL functions. The online source said it would not be possible to do this, so we solved this problem by creating a new table to store the data.

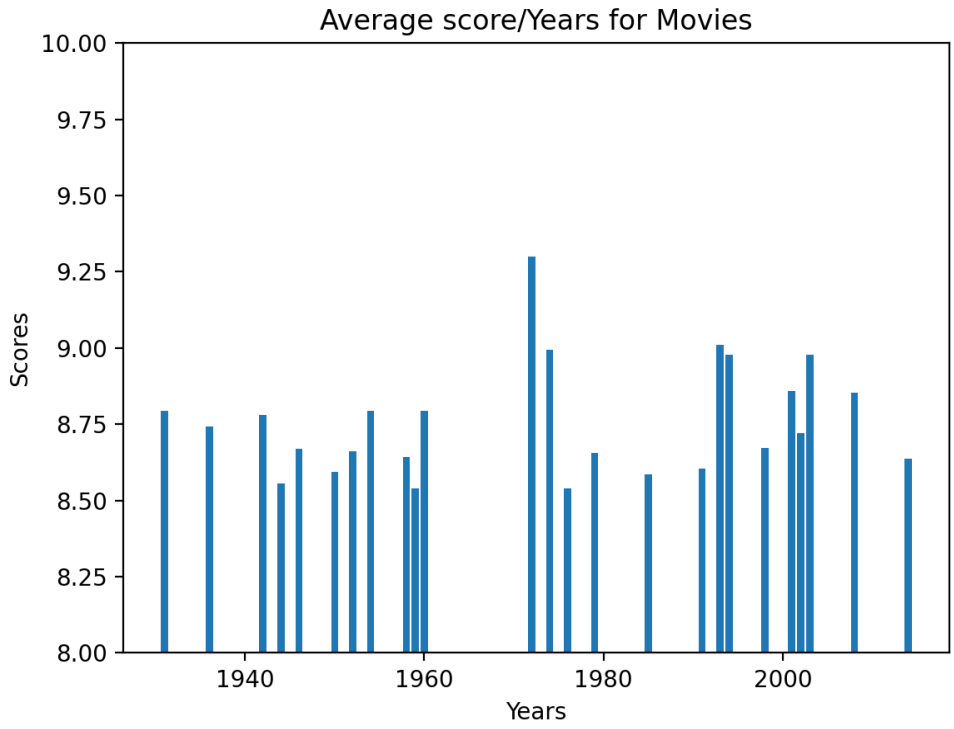
4. **Your file that contains the calculations from the data in the database**

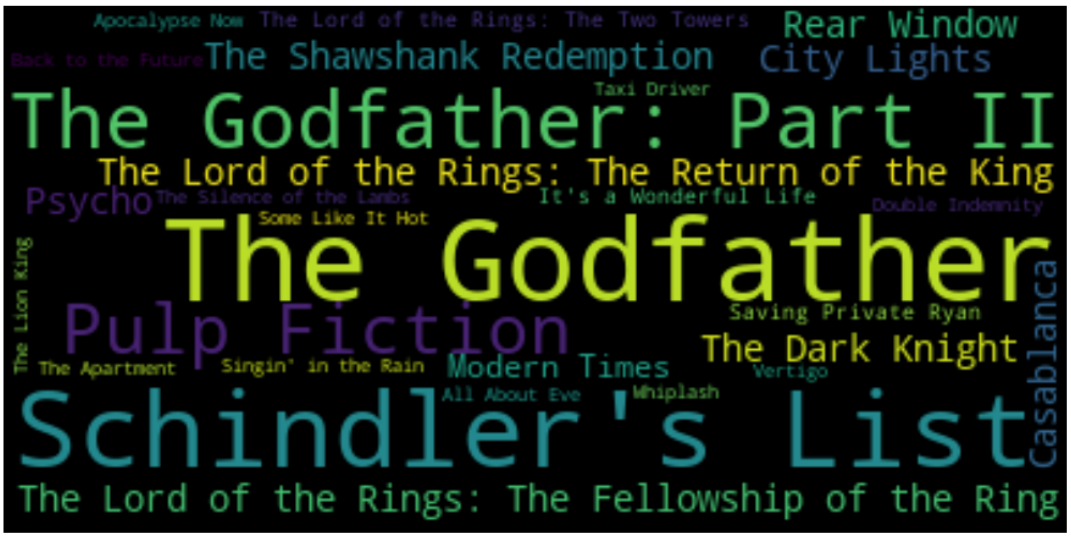
   The following url contains the calculation output file in the form of txt.
   https://drive.google.com/file/d/1UGhzMRVO1V8GZI81s-eaEGt8jPPgj-2t/view?usp=sharing

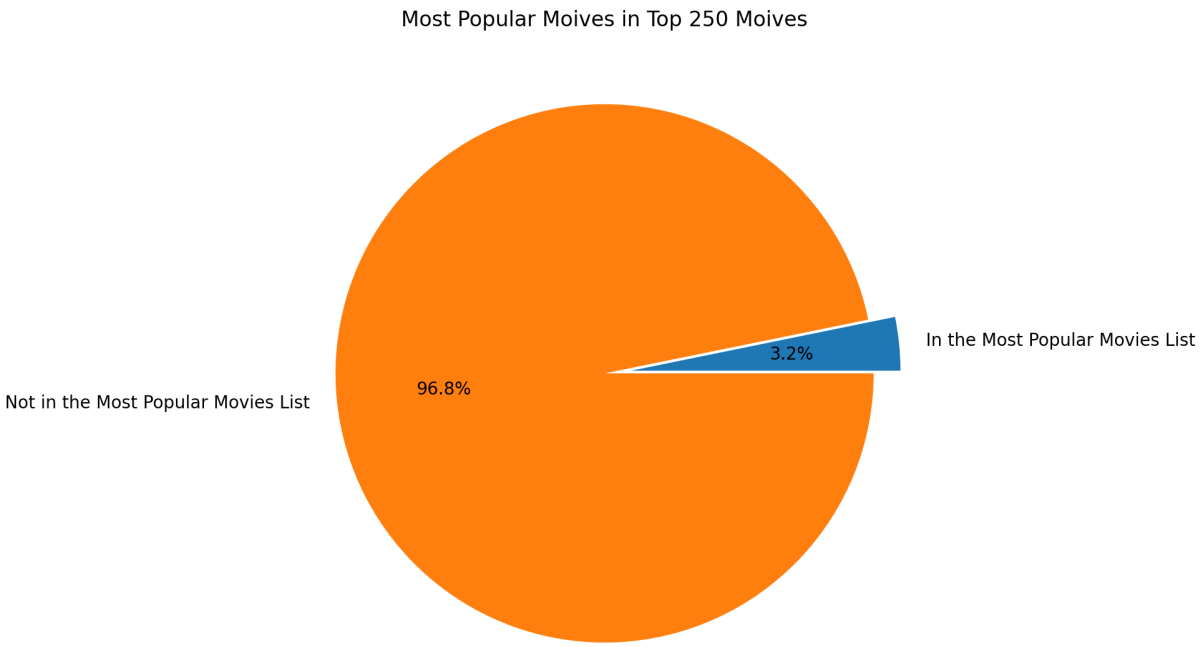
5. **The visualization that you created**






Average score/Years for Movies
10.00
9.75
9.50
9.25
9.00
8.75
8.50
8.25
8.00
Scores
1940
1960
1980
2000
Years


Apocalypse Now
The Lord of the Rings: The Two Towers
Rear Window
Back to the Future
The Shawshank Redemption
City Lights
Taxi Driver
The Godfather: Part II
The Lord of the Rings: The Return of the King
Psycho
The Silence of the Lambs
It's a Wonderful Life
Double Indemnity
Some Like It Hot
The Godfather
The Lion King
Pulp Fiction
Saving Private Ryan
The Dark Knight
The Apartment
Singin' in the Rain
Modern Times
Vertigo
All About Eve
Whiplash
Schindler's List
Casablanca
The Lord of the Rings: The Fellowship of the Ring



6. Instructions for running your code

1) Open local IDE
2) Install the WordCloud package using pip install wordcloud if needed
3) Click “run”
4) Once a graph is shown, close it for the next one to show up.

7. Documentation for each function that you wrote

put_info_into_lists(): This function uses BeautifulSoup to parse the website and get movie titles, scores, and years. Store data in lists Name, Year, and Score.
setUpDatabase(): This function will set up a database
setUpStackerTable(cur, conn): Create a table called stacker and insert data
put_api_1_into_lists(): catch data(titles, years, and scores) from api into database
setUpTMDBTable(cur, conn): Create a table called TMDB and insert data into database
put_api_2_into_lists(): catch data(titles, years, and scores) from api into database
put_api_3_into_lists(): catch data(titles and years) from api into database
setUpIMDBTable(cur, conn): Create a table called IMDB and insert data into database

setUpPopTable(cur, conn): Create a table called PopTable and insert data into database

PopAndTop(cur, conn): JOIN two tables from the same API together and do basic calculation, find connection between top ratings and most populars, and create a pie chart

JoinTables(cur, conn): Create a new table in database by using JOIN based on the same names from three different tables we created

Average(cur, conn): This function uses the three tables we created and creates a new table with the average score of movies and years of movies.

linechar(cur, conn): This uses a processed table as input to output a line chart based on the period of year of the movies to compare their score.

bar_char(cur, conn): This uses a processed table as input to output a bar chart based on the year of the movies to compare their score.

word_cloud(cur, conn): This uses a processed table as input to output a word cloud**.**

8. Documentation for all used resources

Date	Issue Description	Location of Resources	Result
3.25	Can't find an api key for the website	https://developers.themoviedb.org/3/getting-started/authentication	Find the key for TMDB
4.8	Which charts should we create	Course website and slides	Pie chart & line chart
4.12	Incorrect regex expression	https://regex101.com/	Solved
4.23	Not sure about how to join 3 tables	https://learnsql.com/blog/how-to-join-3-tables-or-more-in-sql/	Solved, viewed the example code
4.23	Don't know how to use SQL to get the average of specific columns in each row	https://discuss.codecademy.com/t/how-to-calculate-the-average-value-of-a-row-in-sql/507967	Not solved, the code doesn't work (returned NULL)
4.23	Don't know how to add an empty column into a table	https://www.w3schools.com/sql/sql_alter.asp	Solved, columns added
4.23	Don't know how to set the range of y-axis for line charts	https://pythonguides.com/matplotlib-set-axis-range/	Solved

4.23	Implementation of pie chart	https://www.w3resource.com/graphics/matplotlib/piechart/matplotlib-piechart-exercise-2.php	Solved

9. **Github Repository Link**
   https://github.com/czhe0603/Movie-ratings-.git