New User Existing User

Movie Recommendation

Flixify

Let’s begin the show

CONTENTS

Why Recommendation?

01

Huge Content

Overwhelming content that's growing every day

02

Personalized Recommendation

We enjoy personalized recommendations

03

New Content

A lot of content that might not be discovered

04

User Experience

Enhances user experience on platforms

02

ABOUT DATA

Insights and Visualizations

ABOUT DATA

01

Movies

Movie information with Movie ID, Title, Genre

02

Ratings

Rating info with User ID, Movie ID, Rating

03

Tags

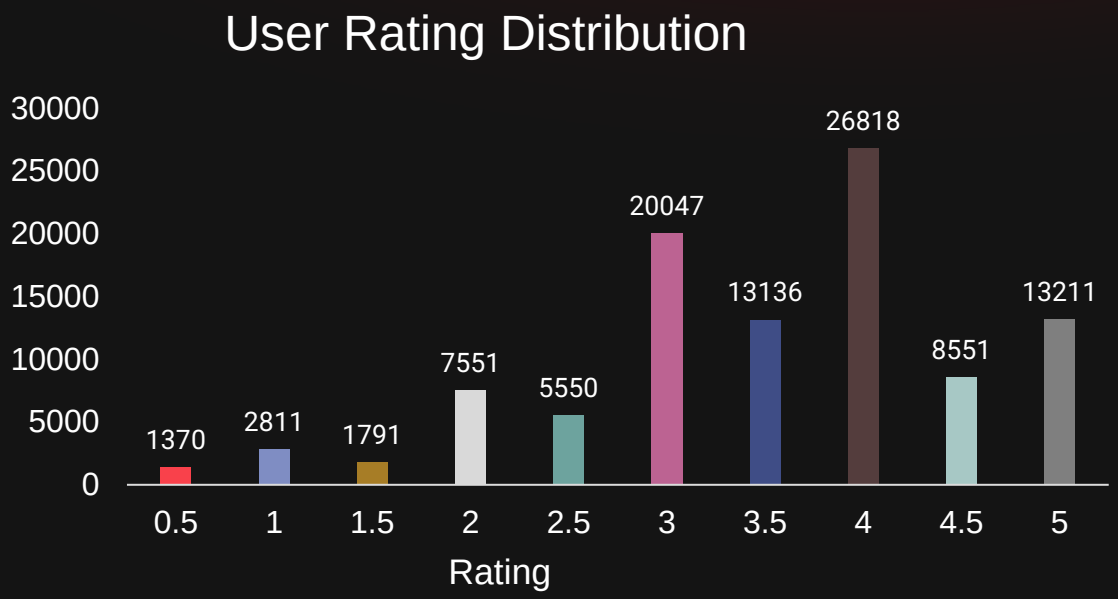
Short one line or one word review with Movie ID

04

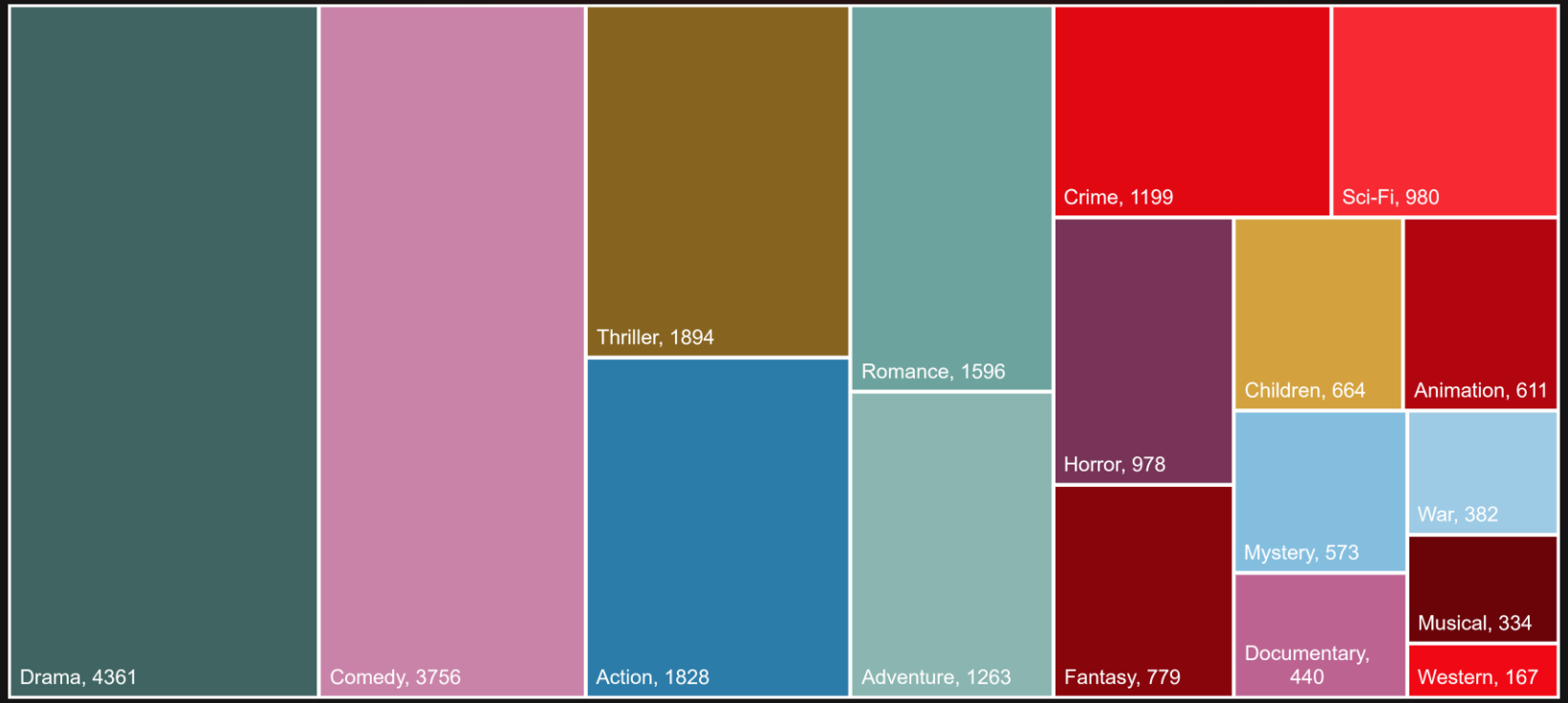
Links

Movie IDs to fetch additional info from TMBD, IMDB

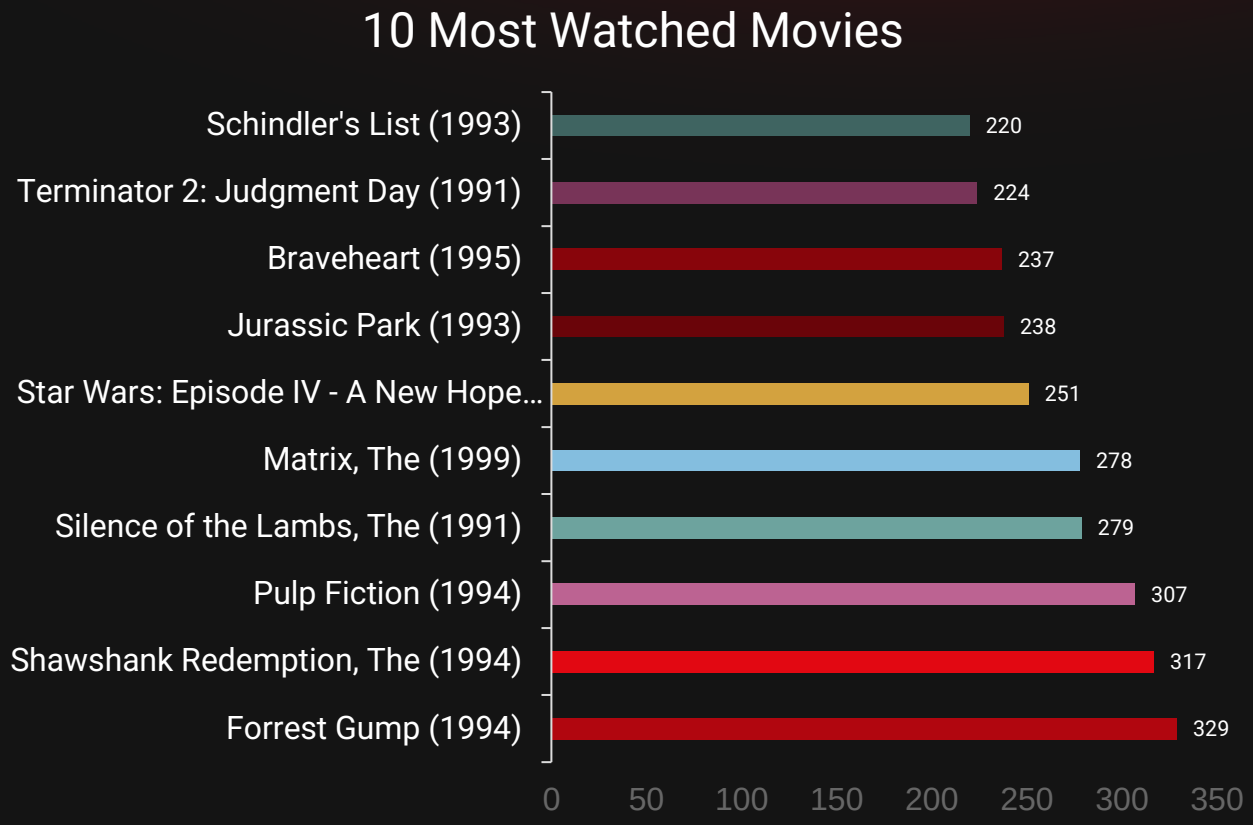
ABOUT DATA


User Rating Distribution
30000
25000
20000
15000
10000
5000
0
1370
2811
1791
7551
5550
20047
13136
26818
8551
13211
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Rating

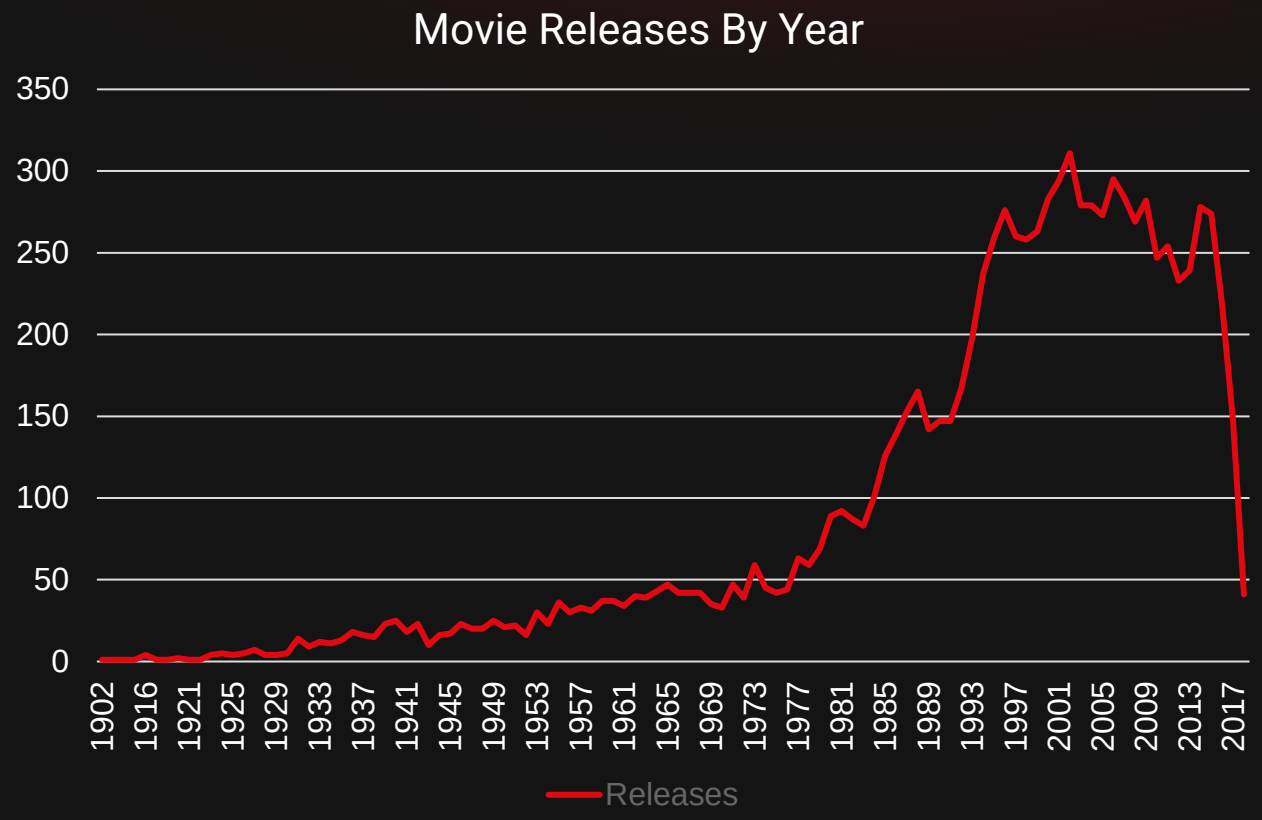
ABOUT DATA


Drama, 4361
Comedy, 3756
Thriller, 1894
Action, 1828
Romance, 1596
Adventure, 1263
Crime, 1199
Sci-Fi, 980
Horror, 978
Fantasy, 779
Children, 664
Mystery, 573
Documentary, 440
Animation, 611
War, 382
Musical, 334
Western, 167

ABOUT DATA


10 Most Watched Movies
Schindler's List (1993)
220
Terminator 2: Judgment Day (1991)
224
Braveheart (1995)
237
Jurassic Park (1993)
238
Star Wars: Episode IV - A New Hope...
251
Matrix, The (1999)
278
Silence of the Lambs, The (1991)
279
Pulp Fiction (1994)
307
Shawshank Redemption, The (1994)
317
Forrest Gump (1994)
329
0
50
100
150
200
250
300
350

ABOUT DATA


Movie Releases By Year
350
300
250
200
150
100
50
0
1902
1916
1921
1925
1929
1933
1937
1941
1945
1949
1953
1957
1961
1965
1969
1973
1977
1981
1985
1989
1993
1997
2001
2005
2009
2013
2017
Releases

03

METHODOLOGIES

Movie Recommendation Techniques

Recommendation Techniques

01

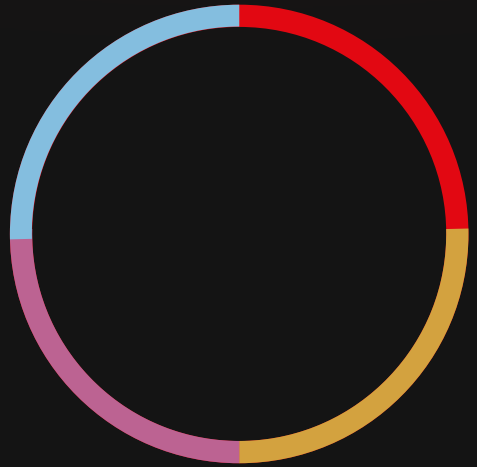
Filtering

Get recommendations by rating and genre

02

Collaborative Filter SVD

Get recommendations from similar user's using singular value decomposition



03

Content Based Filter

Get recommendation by similar movie

04

Collaborative Filtering NN

Get recommendations from similar users using neural networks

Filtering and Similarity

Filtering

- Fetch movie recommendations based on rating and genre.
- Uses SQL queries to fetch recommendations.

Similarity

- Get movie name from user and recommend similar movies.
- Using technique called as cosine similarity

Collaborative Filtering

- A machine learning technique used to make personalized recommendation based on user preferences.
- Recommend movies to a user based on the preferences of other users who have similar taste in movies.

04

ENVIRONMENT

Web application deployment environment

APPLICATION ENVIRONMENT

DATABASE

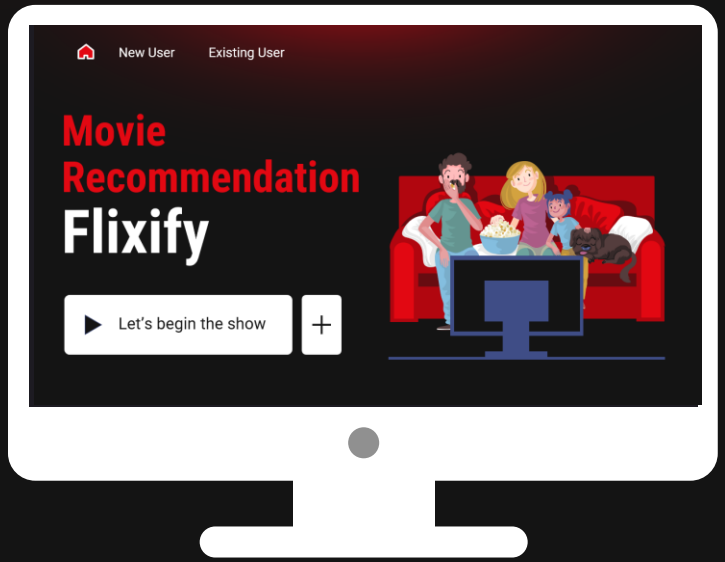
Postgres DB in AWS RDS

REST API

Flask application deployed in heroku

WEB APPLICATION

Angular application deployed using Gooogle Firebase



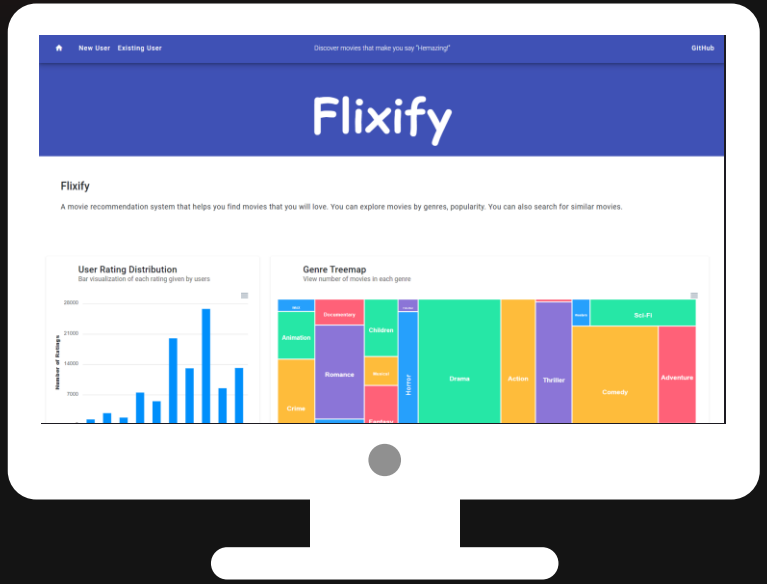
05

APPLICATION FLOW

Walk-Through.

APPLICATION FLOW - DASHBOARD

Visualizations of the data displayed which convey information about dataset and data we are dealing.



APPLICATION FLOW – New User

TOP RATED

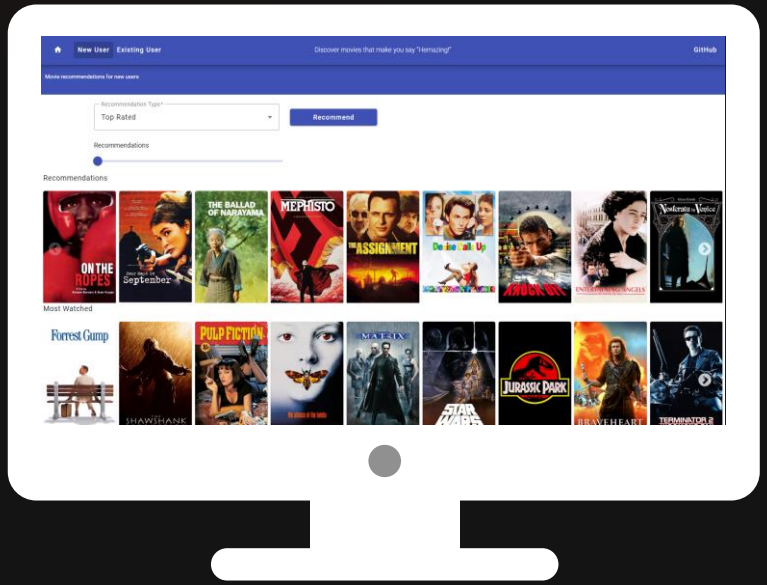
Gets top rated movies from all the genres

TOP RATED BY GENRE

Gets top rated movies by genre

SIMILAR MOVIE

Ask for a movie and recommend similar movie



APPLICATION FLOW – Existing User

COLAB FILTER SVD

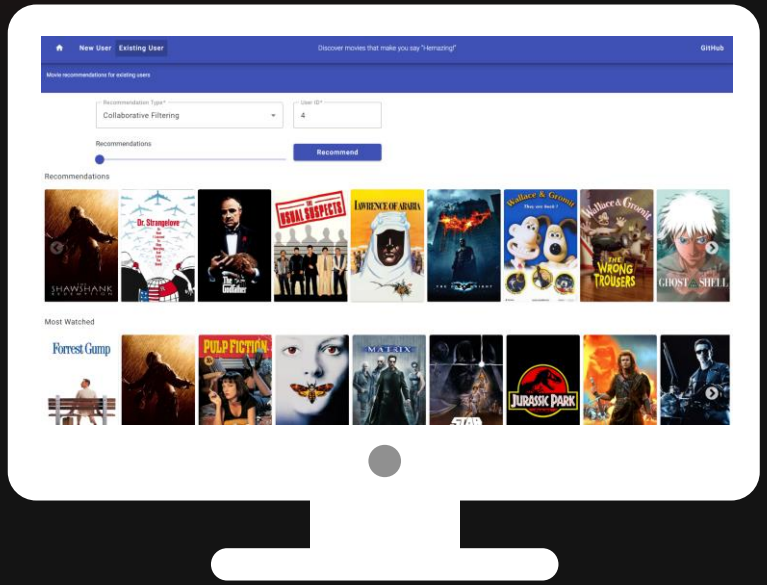
Gets recommendations using SVD Algorithm

COLAB FILTER NN

Gets recommendation using Neural network

SIMILAR MOVIE

Ask for a movie and recommend similar movie



06

CHALLENGES

Problems encountered and future scope

CHALLENGES – Cold Start

- A difficulty in making accurate predictions for new users who limited historical data.
- Active learning can be used to tackle this problem.
- Used similar movie recommendation for new users which is a content-based filtering technique.

CHALLENGES – Other

FLASK APP HTTPS

Application didn't accept response from flask application deployed in AWS with http.

WORKAROUND:

Moved to Heroku.

SLUG FILE LIMIT

Heroku has a deployment package limit of 500 MB but my package was around 2.4 GB.

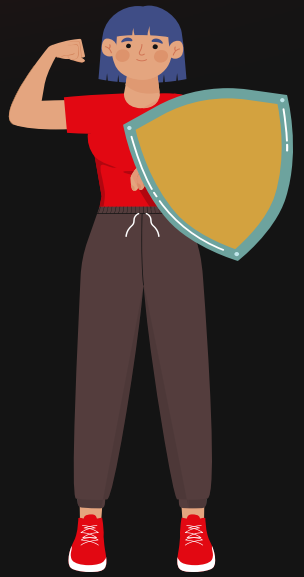
WORKAROUND:

Changed the PyTorch dependency to CPU only dependency.

FUTURE SCOPE – Address Cold Start

Implement Active Learning:

- Ask new users to provide feedback on a subset of data and recommend movies based on user’s feedback



THANK YOU