



Project Presentation

Bianca Zlavog
Florenca Marcaccio
Mansi Rathod
Sanjana Gupta



Outline

- Background
- Data
- Use Cases
- Demo
- Design
- Project Structure
- Lessons Learned
- Future Work



Background

With so many streaming platforms available, it can be overwhelming to browse every catalog to find a movie that you'll like.

We aim to provide a personalized streaming experience based on the user's preferences with a movie recommendation tool.



Data

Source	Files	Features	Highlights
<u>Netflix Prize Data (Kaggle Dataset)</u>	Movie Ratings: combined_data_1.txt, combined_data_2.txt, combined_data_3.txt, combined_data_4.txt	MovieID, UserID, Rating, Date of Rating	Contains 100,480,507 ratings that 480,189 users gave to 17,770 movies.
	Movie Titles: movie_titles.csv	MovieID, Title, Year Released	Contains the movie title and released year for the 17,770 movies in the dataset.
<u>IMDb Datasets</u>	Basic information of each title: title.basics.tsv.gz	Identifier, Title, Content Type, Genres, Start and End Year, Minutes	Contains 6,842,632 titles, but only 736,380 are movies/tv series.
	Rating: title.ratings.tsv.gz	Identifier, Average Rating, Number of Votes	Only 305,357 of the previously filtered titles appear on this file.



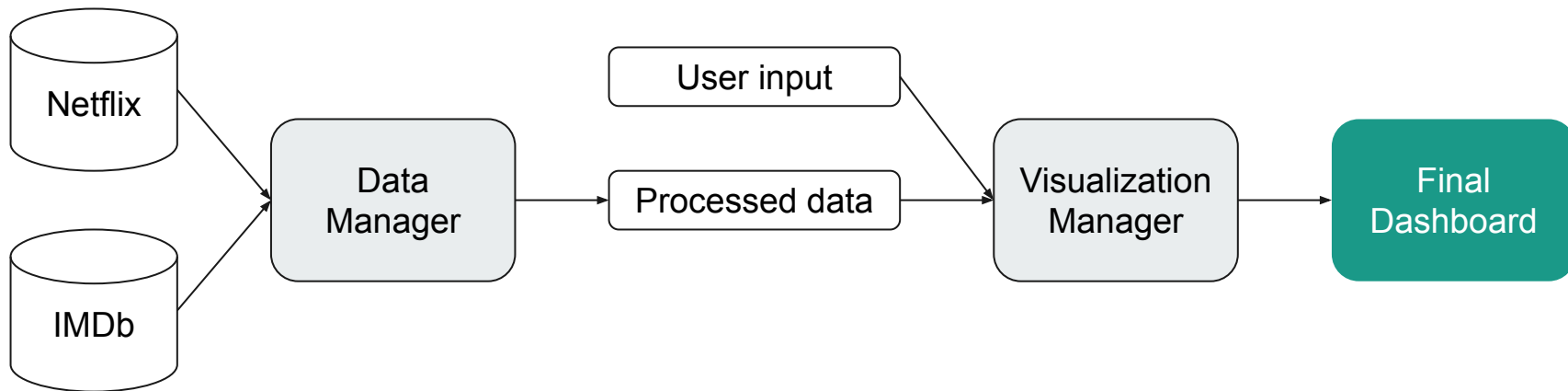
Use Cases

Use Case	The user wants to watch a movie similar to a movie of their choice.	The user wants to see the top 10 movies of a particular year.	The user wants to see the top 10 movies from a particular genre.
User	Inputs the movie they liked.	Inputs a particular year.	Inputs a genre.
Tool	Displays a list of the most similar movies to the one the user input.	Displays a list of the top 10 movies from that year, based on average rating and number of votes.	Displays a list of the top 10 movies from that genre, based on average rating and number of votes.



Demo

Design





Project Structure

```
— LICENSE
— README.md
— CODE_OF_CONDUCT.md
— environment.yml
— .travis.yml
— requirements.txt
— setup.py
— docs/
  — Component_Specification.pdf
  — Final_Presentation.pdf
  — Functional_specs.pdf
  — TechnologyReview.pptx
— examples/
— bingewatch/
  — main.py
  — tab1.py
  — tab2.py
  — imdb.py
  — netflix.py
  — __init__.py
  — data/
    — data_manager.py
    — helper_functions.py
    — processed/
      — dict_recommendations.pkl, imdb_df.csv,
        movie_titles.csv, set_genres.pkl
  — tests/
    — tests.py
```




Lessons Learned

- Difficulty working between languages (Shiny in R vs Python Dash)
- Building dashboards using Dash app
- It would have been easier to start with the CI and testing from the beginning



Future Work

- Add more filtering variables. For example, filter by decade instead of year, by director or actors, by MPAA content rating, etc.
- Add data source in order to include TV Shows in the Recommendation System based on a title chosen by the user.
- Continue improving the recommendation system itself.



Questions?





Thank you!