

Process Documentation

In this project, for my own learning purpose, I initially started with the idea of building a recommendation system. And because movie datasets are generally more accessible, I chose to build a movie recommendation system. In my project, there are two parts:

1. Code

a. Feature 1: content-based filtering algorithm

Description: Some people might love a movie so much and want to find movies that similar to the one that they have just watched. So here, I want to build a system that can recommend similar movies to what user feeds to the algorithm

- i. **Step 1:** This algorithm returns a set of 10 movies that the algorithm believes are similar to the movie that user feeds to the algorithm (I used “the dark knight” in my notebook for demonstration purpose)

Trade-off: content-based filtering does not take into account popularity, it decides whether or not to recommend a movie based on cosine similarity of features (director, genre, cast, and keyword)

- ii. **Step 2:** therefore, on top of the results returned by step 1, I re-filtered by popularity scores and returned the top 10 results sorted by popularity (average voting).
- iii. A page from google movie search is attached. The results of searching “the dark knight” on google are somewhat similar to my algorithm.
- iv. Detail steps and comments are marked in the notebook file.

b. Feature 2: voting-based algorithm

Description: Some people enter the site and had no idea what to watch, however, they might have a favorite genre (e.g. crime movie). Therefore, I want to add a feature to the system so that my site can recommend movies to people based on a specific genre

- i. **Step 1:** One of the YouTube videos I watched introduced the IMDb weighted rating system, which takes a weighted rating score instead of the pure score. I adopted this approach and let the algorithm calculate a weighted rating score for each movie. Then I filtered by genre and popped the top-10 rated movies.

- ii. Detail steps and comments are marked in the notebook file.

2. Mock-up of a user interface

The idea of this user interface is to allow users to either search by a movie that he/she has seen or search by genre.

Because it was really difficult for me to find a photo database that has all the movie photos that matches the movies in my dataset, I did not code the UI of this recommendation system. Instead, I built a mock-up to show how the system should look like. The general idea is conveyed as follow:

- **<More Genre> Button:** Because there are so many possible genres and they cannot all possibly fit on the landing page, by adding a more genre button, I want users to be able to enter a subpage that allow them to choose from more detailed categories.
- **<Show Genre on Hover>** By hovering on any movie pictures on the landing page, a genre name will be shown. In case users have no idea what the movie on the landing page talks about, show genre on hover notifies users what genre this movie belongs to.
 - From user experience perspective, the cover movie photo of a genre also signals a general theme of this genre
 - e.g. for the crime movie, I used a photo of the dark knight, the color code of that movie well-demonstrates what a crime movie might look like — dark, cold, evil, etc.
- **<Top 10 movies by Genre>** This page shows the top 10 rated movies using the IMDb weighted voting score.
 - Filter by time: In this algorithm, I did not assign weights to time or filter movies by time, therefore, it appears lots of movies shown on the list are pretty old (from 1980s – 1990s). In case a user really wants to search for movies in the recent 2 years, I included a button to allow users to filter movies by time.
 - More: In case the user has watched all 10 movies on the list (or is not interested), this button allows users to see more down the list.
 - Ideally, all movies on the list should be clickable and lead to the corresponding movie detail page
- **<Top 10 movies by user's loved movie>** This page shows the top 10 rated movies that are similar to the user's loved movie.
 - Doesn't like it? Change a batch: this button allows users to switch a batch of recommendations. In the backend, this could just be done by popping the next 10 movies in the array.
 - Ideally, all movies on the list should be clickable and lead to the corresponding movie detail page

A complete version of the mockup is hosted on:

<https://www.figma.com/proto/LuHyG0h47FUhRasBUEhpN4/Untitled?node-id=1%3A2&scaling=scale-down>

The walk-through videos are on google drive (in demo folder):

<https://drive.google.com/open?id=1k3WWuzum3qrCZtWP-aE0OpHhssH7Hf3D>