Microsoft-Engage 2022

MyTV (MOVIE APP)
- *Movie Recommendation System*

What is a Recommendation system?
A recommender system, or a recommendation system (sometimes replacing 'system' with a synonym such as platform or engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item.

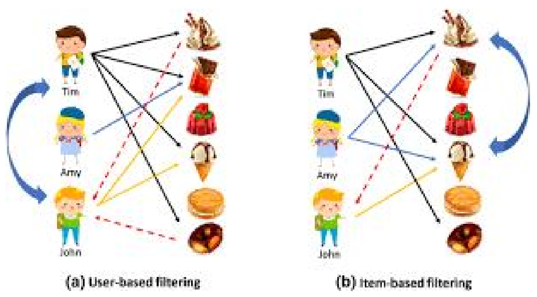
Recommender systems are used in a variety of areas, with commonly recognized examples taking the form of playlist generators for video and music services, product recommenders for online stores, or content recommenders for social media platforms and open web content recommenders. These systems can operate using a single input, like music, or multiple inputs within and across platforms like news, books and search queries. There are also popular recommender systems for specific topics like restaurants and online dating. Recommender systems have also been developed to explore research articles and experts, collaborators, and financial services.

Recommender systems usually make use of either or both collaborative filtering and content-based filtering (also known as the personality-based approach), as well as other systems such as knowledge-based systems.

- **Collaborative filtering** approaches build a model from a user's past behavior as well as similar decisions made by other users.
- **Content-based filtering** approaches utilize a series of discrete, pre-tagged characteristics of an item in order to recommend additional items with similar properties.

Various approaches used in recommendation systems can be described as;

***Collaborative Filtering*,** as mentioned previously this approach is widely used as it is based on the assumption that people who agree in the past will agree in the future and will like similar kind of items as they liked before.
These filtering is classified as memory-based and model-based. A well-known example of memory-based approaches is the user-based algorithm, while that of model-based approaches is the Kernel-Mapping Recommender.



Most importantly, the advantage which we get while using collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself.

Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity.

- **Cold start**: For a new user or item, there isn't enough data to make accurate recommendations. Note: one commonly implemented solution to this problem is the Multi-armed bandit algorithm.

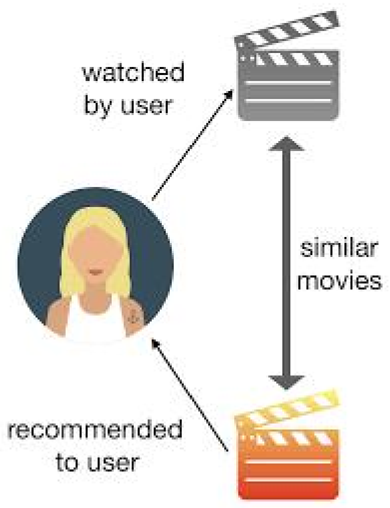
- **Scalability**: There are millions of users and products in many of the environments in which these systems make recommendations. Thus, a large amount of computation power is often necessary to calculate recommendations.
- **Sparsity**: The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

One of the most famous examples of collaborative filtering is item-to-item collaborative filtering, an algorithm popularized by Amazon.com's recommender system.

Another approach we can think of is ***Content-Based Filtering*,** as we saw previously. It is based on a description of the item and a profile of the user's preferences. These methods are best suited to situations where there is known data on an item but not on the user.

Content-based recommenders treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on an item's features.

In this system, keywords are used to describe the items, and a user profile is built to indicate the type of item this user likes. In other words, these algorithms try to recommend items similar to those that a user liked in the past or is examining in the present.



A widely used algorithm is the tf-idf representation (also called vector space representation).The system creates a content-based profile of users based on a weighted vector of item features.

Simple approaches use the average values of the rated item vector while other sophisticated methods use machine learning techniques such as Bayesian Classifiers, cluster analysis, decision trees, and artificial neural networks in order to estimate the probability that the user is going to like the item.

A key issue with content-based filtering is whether the system can learn user preferences from users' actions regarding one content source and use them across other content types.

(*This part is optional if you want to know more*)

This can be used as a third method of approach,

Hybrid recommender systems: Most recommender system ow a days uses hybrid approach as it is the combination of both Content- based and Collaborative-based filtering. This becomes more efficient on comparing with the previous two methods of approaches.

Some hybridization techniques include:

- **Weighted**: Combining the score of different recommendation components numerically.
- **Switching**: Choosing among recommendation components and applying the selected one.
- **Mixed**: Recommendations from different recommenders are presented together to give the recommendation.
- **Feature Combination**: Features derived from different knowledge sources are combined together and given to a single recommendation algorithm.
- **Feature Augmentation**: Computing a feature or set of features, which is then part of the input to the next technique.
- **Cascade**: Recommenders are given strict priority, with the lower priority ones breaking ties in the scoring of the higher ones.
- **Meta-level**: One recommendation technique is applied and produces some sort of model, which is then the input used by the next technique.

More approaches which can be included are;

- ✧ **Session-based recommender systems**
- ✧ **Reinforcement learning for recommender systems**
- ✧ **Multi-criteria recommender systems**
- ✧ **Risk-aware recommender systems**
- ✧ **Mobile recommender systems**

These are some more approaches which we do not require now as we are making a simple recommendation system.

<u>MyTV!!</u>

Now coming to this system, which is been designed by keeping content-based filtering in mind. It is a simple recommender system that gives you some similar movie suggestions.

Technologies and Tools used:

1. VS code
2. TMDB API
3. HTML
4. CSS
5. JavaScript (Vanilla)
6. Git-hub

MyTV contains all the American TV shows and movies which you may like. The API used fetches all the data from the parent website i.e, TMDB. Data like genre, tags, tagline, ratings, reviews, summary etc, are used efficiently for getting your favorite movie to watch. CSS and HTML is used for the designing the website and JavaScript is the main part of for this whole MyTV website. Git-hub is used for hosting our project website.

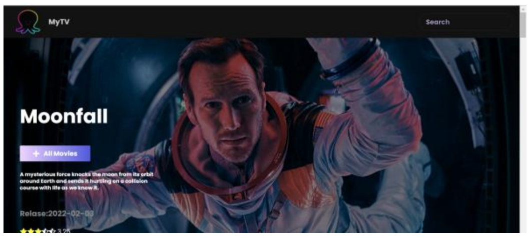
Hosting it on your system.

For hosting it on your system you have to download this zip file from git-hub where you will get all the source code provided already. You don’t need to install any of the packages for this project.

After downloading the zip file, site it to a folder on your desktop and open the index.html on your browser.

If not, the MyTV website link is provided in the git-hub repository you can directly click on the link that will redirect you to the right place.

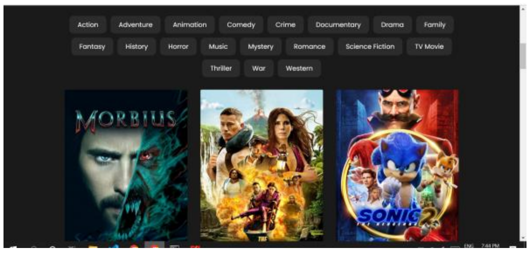
The website will look something like this:



A mysterious force knocks the moon from its orbit around Earth and sends it hurtling on a collision course with life as we know it.

Relase:2022-02-03

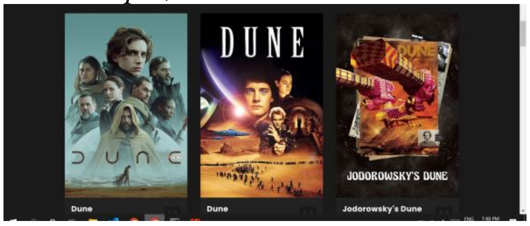
While scrolling down, you will get to see this:



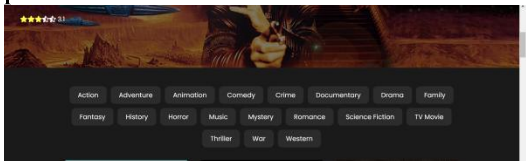
Searching a movie:

To watch your favorite movie, type the name in the search box provided on the top right-hand side and hit enter. This will provide you your movie and some more related to it, which is done accordingly with the help of tags, title, genre, cast and directors.

For example,



You can even sort and discover some more movies with the help of categories provided below the banner.



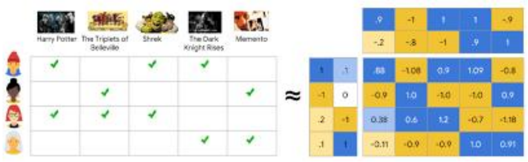
Movie from these categories will be displayed down.

Further describing this website, any user who is new can use it, as it provides suggestions on the basis of tags and directors, you won't see any cold start problem.

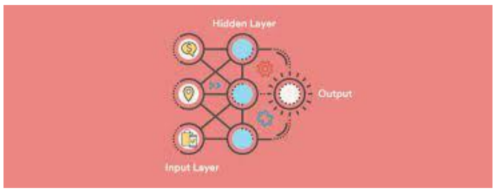
Improvements

This website is very simple and compact you can use it on your mobile too. Moreover we can modify it to use it more efficiently .

For getting the best result out of this you can implement collaborative filtering especially the Matrix factorization method which provides accurate results for your inputs.



Another improvement is using the Deep-Learning method, this is more advanced method used in any recommendation system. Hence, the best way.



Thanks for Reading <3