

# BASIC RECOMMENDATION SYSTEM

Ujjawal Varshney  
Email: [ujjawalvarshney20@gmail.com](mailto:ujjawalvarshney20@gmail.com)

## Abstract

A recommendation engine filters the data using different algorithms and recommends the most relevant items to users. It first captures the past behavior of a customer and based on that, recommends products which the users might be likely to buy. If a completely new user visits an e-commerce site, that site will not have any past history of that user. So how does the site go about recommending products to the user in such a scenario? One possible solution could be to recommend the best selling products, i.e. the products which are high in demand. Another possible solution could be to recommend the products which would bring the maximum profit to the business

## METHODOLOGY

### THE DATASET:

The Movie Database, also known as TMdb is a database that contains detailed information on over 500,000 movies. The Movies dataframe has links to the movie posters in it, but the vast majority of them are outdated and no longer work. In order to find the poster links, the tmdbsimple library, which acts as a Python wrapper for the TMDB API was used. The ID of the movie is all that it takes to obtain that movie's updated information. The TMDB website has a simple structure to its URLs, therefore getting the link to each movie's page involves taking the base site link and adding the id of the movie, a dash, and the movie title.

## DATA PREPARATION:

A recommendation system is a type of information filtering system which attempts to predict the preferences of a user, and make suggests based on these preferences. There are a wide variety of applications for recommendation systems. These have become increasingly popular over the last few years and are now utilized in most online platforms that we use. The content of such platforms varies from movies, music, books and videos, to friends and stories on social media platforms, to products on e-commerce websites, to people on professional and dating websites, to search results returned on Google. Often, these systems are able to collect information about a users choices, and can use this information to improve

their suggestions in the future. For example, Facebook can monitor your interaction with various stories on your feed in order to learn what types of stories appeal to you. Sometimes, the recommender systems can make improvements based on the activities of a large number of people

### Data Pre-processing:

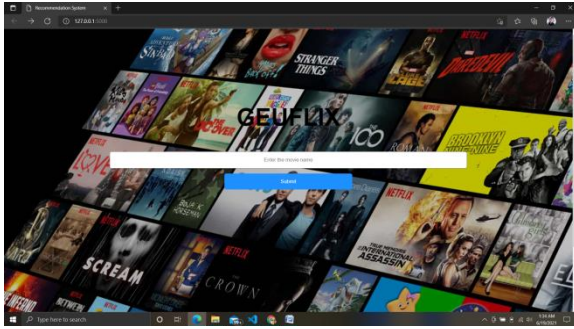
1. Python -: Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

2. **Visual Code : Visual Studio Code** (famously known as **VS Code**) is a free open source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times. VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even creating new extensions including code linters, debuggers, and cloud and web development support.

The VS Code user interface allows for a lot of interaction compared to other text editors. To simplify user experience, VS Code is divided into five main regions:

- The activity bar
- The side bar
- Editor groups
- The panel
- The status bar

## EXAMPLE-:



## CONTENT-BASED FILTERING

In content-based filtering, items are recommended based on comparisons between item profile and user profile. A user profile is content that is found to be relevant to the user in form of keywords (or features). A user profile might be seen as a set of assigned keywords (terms, features) collected by algorithm from items found relevant (or interesting) by the user. A set of keywords (or features) of an item is the Item profile. For example, consider a scenario in which a person goes to buy his favorite cake 'X' to a pastry. Unfortunately, cake 'X' has been sold out and as a result of this the shopkeeper recommends the person to buy cake 'Y' which is made up of ingredients similar to cake 'X'. This is an instance of content-based filtering.

## COSINE SIMILARITY SCORE

Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis.

A document can be represented by thousands of attributes, each recording the frequency of a particular word (such as a keyword) or phrase in the document. Thus, each document is an object represented by what is called a *term-frequency vector*. For example, in Table 2.5, we see that *Document1* contains five instances of the word *team*, while *hockey* occurs three times. The word *coach* is absent from the entire document, as indicated by a count value of 0. Such data can be highly asymmetric.

## ADVANTAGES

One advantage of cosine similarity is its low-complexity, especially for sparse vectors: only the non-zero dimensions need to be considered. Other names of cosine similarity are Orchini similarity and the Tucker coefficient of congruence; Ochiai similarity is cosine similarity applied to binary data

## FLASK

Flask is a lightweight [WSGI](#) web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around [Werkzeug](#) and [Jinja](#) and has become one of the most popular Python web application frameworks.

Flask offers suggestions, but doesn't enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easy.

```
# save this as app.py
from flask import Flask, escape, request

app = Flask(__name__)

@app.route('/')
def hello():
    name = request.args.get("name", "World")
    return f'Hello, {escape(name)}!'

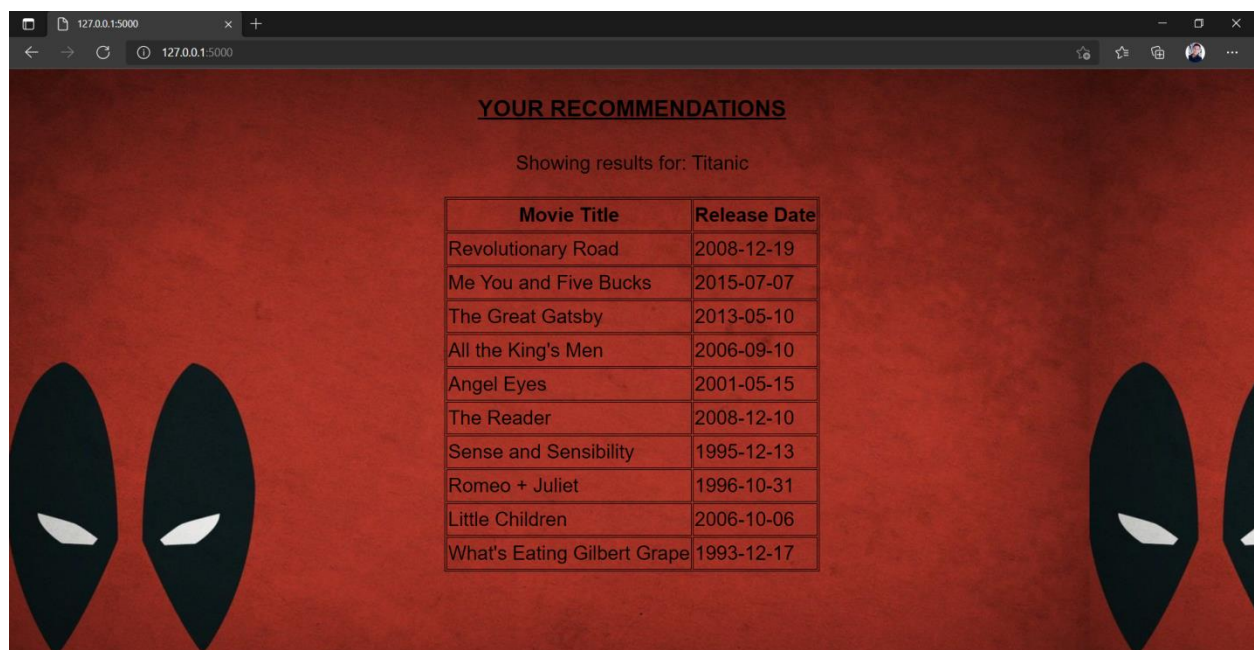
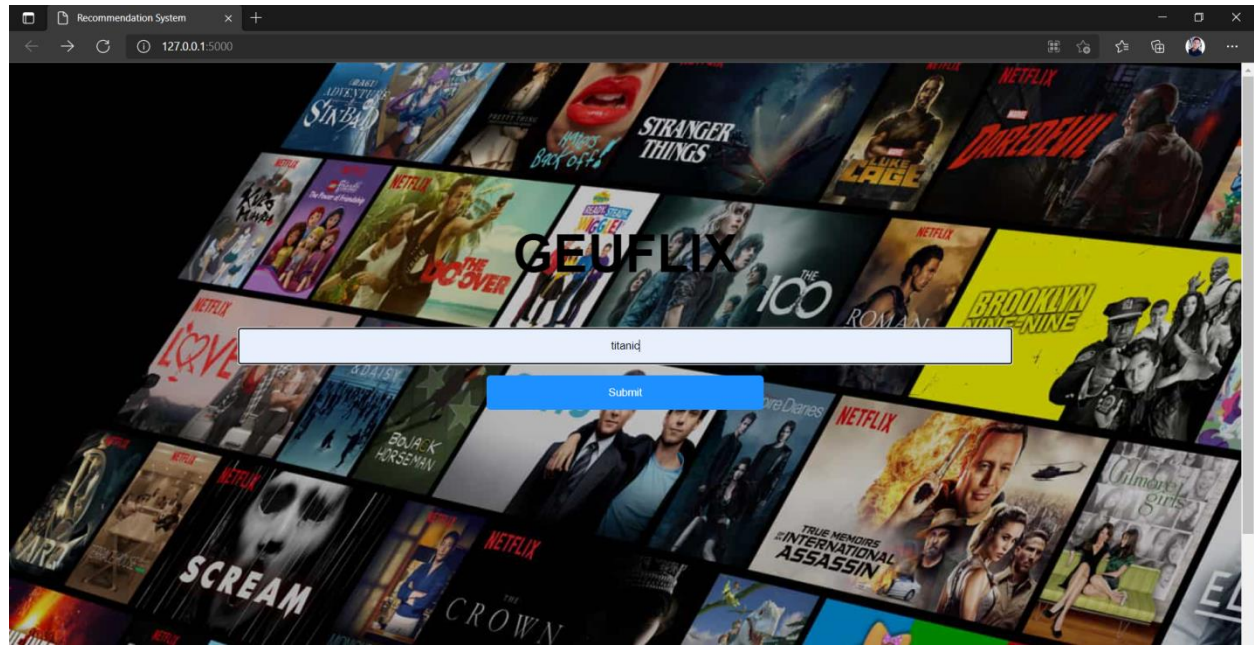
$ flask run
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

## WEB PAGE

Integrating the model with a webapp to run on local host using Flask.

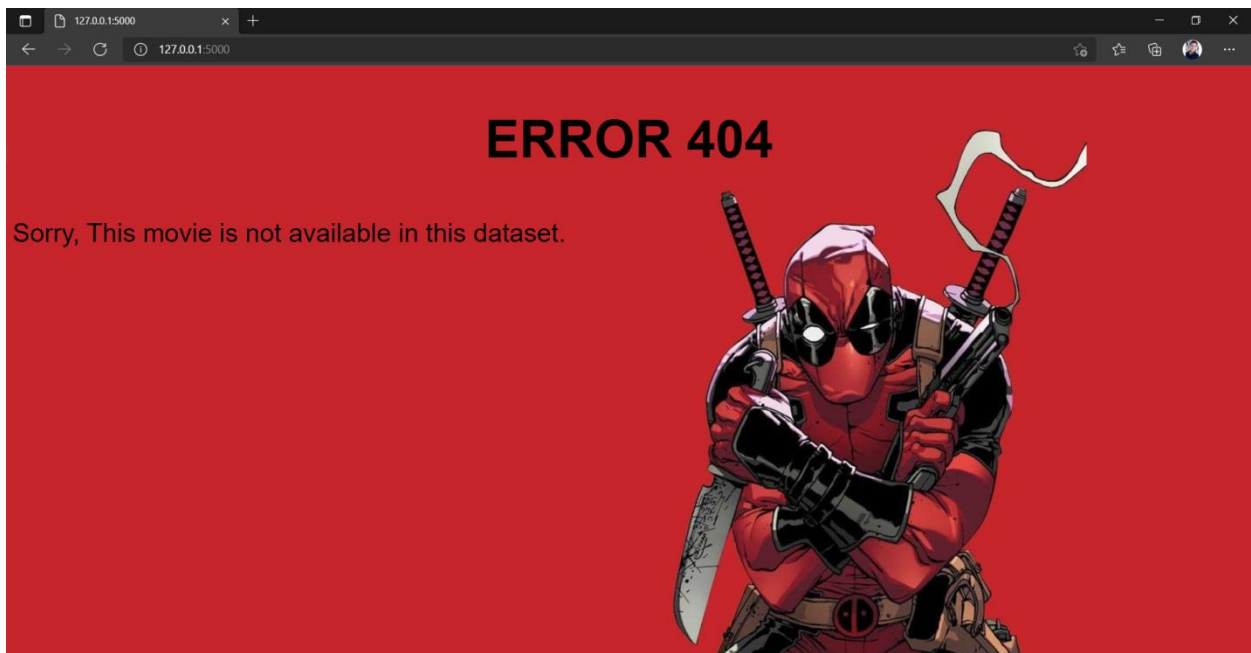
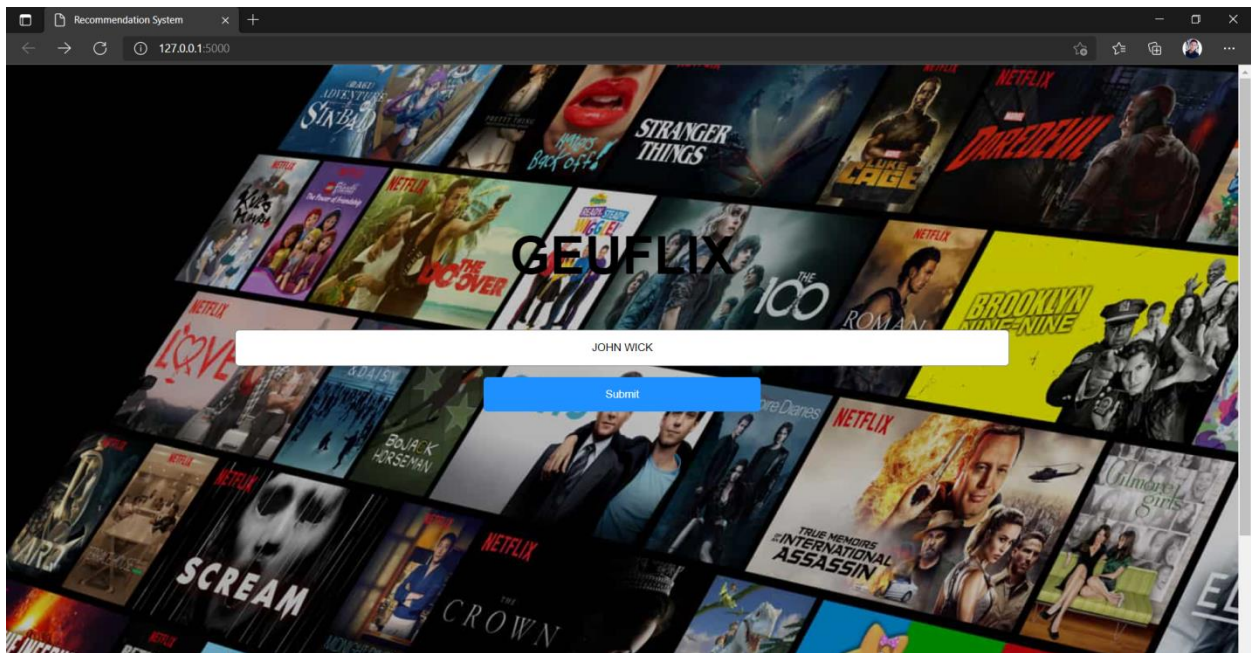
LOCAL IP:- 127.0.0.1:5000

EX:- POSITIVE SAMPLE





EX-: NEGATIVE SAMPLE



## CONCLUSIONS

Movie recommendation systems which are existing have poor efficiency due to which movies are suggested in view of aspects for example - movie rated & evaluated by the User. They have almost same viewing tastes, by means of data mining and insisting movies based on juncture of the three methods mentioned above that is - User Based Collaborative filtering, Content-based algorithm & data mining because of which the user will not only be recommended movies but this scheme also delivers the user with additionally advanced and sophisticated endorsements as movies which have a poor rating score in any of the Movie features.

## REFERENCES

- [1] <https://youtu.be/UbCWoMf80PY>
- [2] [https://youtu.be/lj4I\\_CvBnt0](https://youtu.be/lj4I_CvBnt0)
- [3] <https://ieeexplore.ieee.org/document/6382910>
- [4] <https://www.mygreatlearning.com/blog/masterclass-on-movie-recommendation-system/>
- [5] <https://www.geeksforgeeks.org/movie-recommendation-based-emotion-python/>