MOVIE RECOMMENDATION ON ML

A PROJECT REPORT

Submitted by

Manan Sharma	**21BCE10210**
P.Harshit Kumar	**21BCE10709**
Arnav Sinha	**21BCE10168**
Shivang Choudhary	**21BCE10188**
Arvind Garg	**21BCE10184**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE
MADHYA PRADESH - 466114

September 2022

VIT BHOPAL UNIVERSITY,KOTHRIKALAN, SEHORE MADHYA PRADESH – 466114

BONAFIDE CERTIFICATE

Certified that this project report titled **Movie Recommender System** is the bonafide work of **Manan Sharma (21BCE10210), Arnav Sinha (21BCE10168), P Harshit Kumar(21BCE10709), Arvind Garg (21BCE10184), Shivang Choudhary(21BCE10188)** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here does not form part of any other project / research work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR	**PROJECT GUIDE**
Dr. Preetam Suman	Dr. Anand Motwani
School of Computer Science and Engineering	School of Computer Science and Engineering

The Project Exhibition I Examination is held on 30 Sept 2022.

ACKNOWLEDGEMENT

First and foremost, I would like to thank GOD for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to Dr Preetam Suman, Head of the Department, School of Computer Science and Engineering for much of his valuable support encouragement in carrying out this work.

I would like to thank my internal guide Dr. Anand Motwani, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science and Engineering, who extended directly or indirectly all support.

Last, but not the least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

ABSTRACT

The aim of the project is to build a responsive online website – Movie Recommendation System. Our website helps user to get them to know which movie they want to watch according to their preference. Our website takes the user data and recommends the movie. TMDB API has been integrated into the website to fetch the movies. Hence in order to address these functionalities, deep research has been made on the earlier works for automating the machine learning model which does the recommendation. This Machine Learning Model was developed using the Jupiter Notebook.

HTML and CSS were used to develop the responsive frontend along with Django Framework which made the website user-friendly.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.1 Recommender systems are information filtering tools that aspire to predict the rating for users and items, predominantly from big data to recommend their likes. Movie recommendation systems provide a mechanism to assist users in classifying users with similar interests. This makes recommender systems essentially a central part of websites and e-commerce applications. This article focuses on the movie recommendation systems whose primary objective is to suggest a recommender system through data clustering and computational intelligence. In this research article, a novel recommender system has been discussed which makes use of k-means clustering by adopting cuckoo search optimization algorithm applied on the Movielens dataset.

1.2 A movie recommendation system provides a level of comfort and personalization that helps the user interact better with the system and watch movies that cater to his needs. Providing this level of comfort to the user was our primary motivation in opting for movie recommendation system as our Project.

1.3 Many other web platforms all use recommendation engines to predict their users' preferences and boost their business but They recommend movies based on past activity of user not based on sentiments or reviews of users.

Diffrent types of recommendation engines

The most common types of recommendation systems are content-based and collaborative filtering recommender systems. In collaborative filtering, the behavior of a group of users is used to make recommendations to other users. The recommendation is based on the preference of other users. A simple example would be recommending a movie to a user based on the fact that their friend liked the movie. There are two types of collaborative models Memory-based methods and Model-based methods. The advantage of memory-based techniques is that they are simple to implement and the resulting recommendations are often easy to explain. They are divided into two:

- User-based collaborative filtering: In this model, products are recommended to a user based on the fact that the products have been liked by users similar to the user. For example, if Derrick and Dennis like the same movies and a new movie come out that Derick like, then we can recommend that movie to Dennis because Derrick and Dennis seem to like the same movies.

- Item-based collaborative filtering: These systems identify similar items based on users' previous ratings. For example, if users A, B, and C gave a 5-star rating to books X and Y then when a user D buys book Y they also get a recommendation to purchase book X because the system identifies book X and Y as similar based on the ratings of users A, B, and C.

Model-based methods are based on Matrix Factorization and are better at dealing with sparsity. They are developed using data mining, machine learning algorithms to predict users' rating of unrated items. In this approach techniques such as dimensionality reduction are used to improve accuracy. Examples of such model-based methods include Decision trees, Rule-based Model, Bayesian Model, and latent factor models.

Content-based systems use metadata such as genre, producer, actor, musician to recommend items say movies or music. Such a recommendation would be for instance recommending Infinity War that featured Vin Diesel because someone watched and liked The Fate of the Furious. Similarly, you can get music recommendations from certain artists because you liked their music. Content-based systems are based on the idea that if you liked a certain item you are most likely to like something that is similar to it.

CHAPTER 2: LITERATURE REVIEWS

The main use of Recommendation Systems is to help users to have personalized results according to their preferences. Recommendation Systems can also be used as a filtering technique to take out the best result from a set of predicted results by use of some Machine Learning Algorithm target according to the viewer search. Movie suggestions for users are dependent on web-based models. As Movies can be segregated on basis of genres like thriller, animation, comedy, action, drama etc. Another way to categorize movies are on the basis of some metadata such as cast, year of release, language or director. Nowadays, most of online video-streaming platforms provide a number of similar types of tv shows and movies to the user by the help of utilizing users previous search keywords and previous watch history of the user. These Movie Recommendation Systems helps the user to search the movies or shows of their choice and hence reduce their time to think about choosing what to watch. The main goal while building a Movie Recommendation System is to make it reliable and efficient in order to provide the suggestions to users accurately what they are looking for. Basically, Recommendation Systems are divided into two different types Collaborative Filtering (CF) and Content Based Filtering (CBF).

Many Recommendation Systems have been introduced till date following different approaches for the computation like CBF, CF and hybrid models for recommendation. Sentiment Analysis is also used to improve the recommendation efficiency.

A. <u>Content Based (CB), Collaborative Filtering (CF) and Hybrid Filtering</u> technique showed better results. The most widely used and researched recommendation technique is Content-based Filtering. Description of the item and a profile of the users' preferences are used in Content-based filtering. Contador et al. proposed the use of user and item profiles by describing them in terms of weighted lists of social tags to provide music recommendations. Meteran et al. introduced a Personalized

Recommender System (PRES) to give suggestions of articles for home improvement by the use of similarity between the user profile vector and a document was formed by using the combination of TF-IDF and the cosine similarity, whereas Goossen et al. introduced a new method for news recommendation based on TF-IDF and a domain characteristic which was called CF-IDF. Results of CF-IDF as compared to TF-IDF approaches were better on several measures such as accuracy, recall, and the F1-Score when tested and implemented on the Athena framework. The latest studies have shown that the hybrid methods are better than traditional CF and CBF methods.

B. Sentiment Analysis

Sentiment analysis is commonly used to retrieve user reviews and reactions over a particular item or topic. In, rating of products for online software services is depicted by use of sentiment analysis. Its research enhances both CBF and CF algorithms by the use of review techniques such as sentiment analysis and subjective logic. This technique can also be used to calculate the polarity and confidence of review sentences.

CHAPTER 3:PROJECT PROCEDURE

- **DATA SET:**

  A dataset in machine learning is, quite simply, a collection of data pieces that can be treated by a computer as a single unit for analytic and prediction purposes. This means that the data collected should be made uniform and understandable for a machine that doesn't see data the same way as humans do. For this, after collecting the data, it's important to preprocess it by cleaning and completing it, as well as annotate the data by adding meaningful tags readable by a computer. Moreover, a good dataset should correspond to certain quality and quantity standards. For smooth and fast training, you should make sure your dataset is relevant and well-balanced. Try to use live data whenever possible and consult with experienced professionals about the volume of the data and the source to collect it from.

  In our Recommendation System, All the movie data we used is from TMDB . In the end we get 5000 movies and related information. For the perspective of recommender system, a movie can be described by a collection of features, which can be genres, actors, directors and so on.

  Attributes in our datasets are:

  MOVIE_ID: UNIQUE NUMBER TO FETCH DATA FROM API

  TITLE : MOVIES TITLES (STRING)

  CAST : ACTOR’S NAME

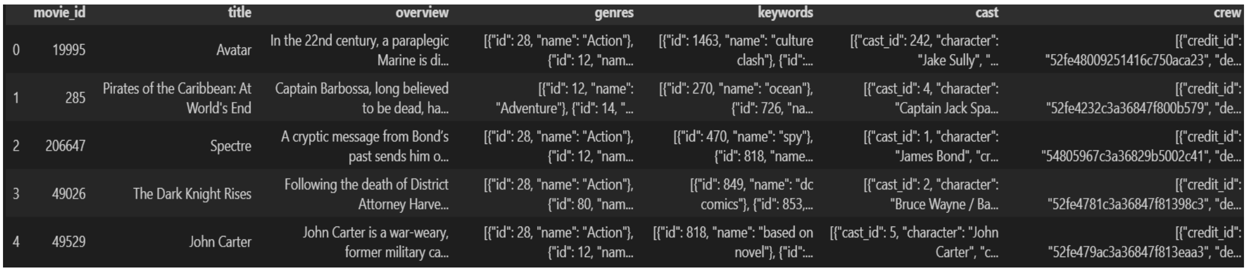
  OVERVIEW : DESCRIPTION OF THE MOVIE

  GENRE : GENRE

  KEYWORDS : IMPORTANT TAGS IN MOVIES

  CREW : DIRECTOR’S NAME

  PRODUCTION HOUSE: NAME OF PRODUCTION COMPANIES

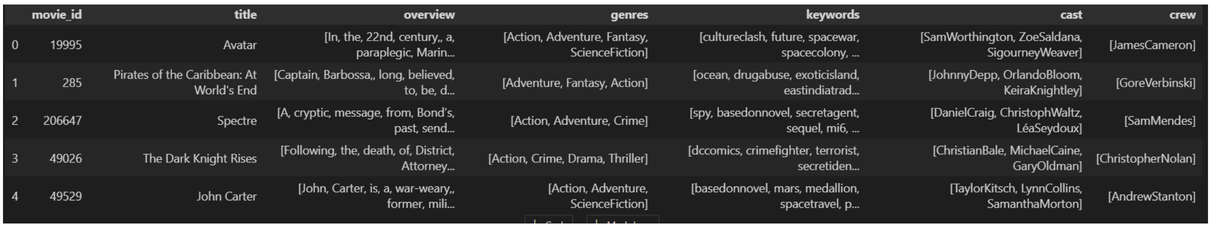


	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 1463, "name": "culture clash"}, {"id":...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
2	206647	Spectre	A cryptic message from Bond’s past sends him o...	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 470, "name": "spy"}, {"id": 818, "name...	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	[{"id": 849, "name": "dc comics"}, {"id": 853,...	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
4	49529	John Carter	John Carter is a war-weary, former military ca...	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 818, "name": "based on novel"}, {"id":...	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...

- **DATA PROCESSING:**

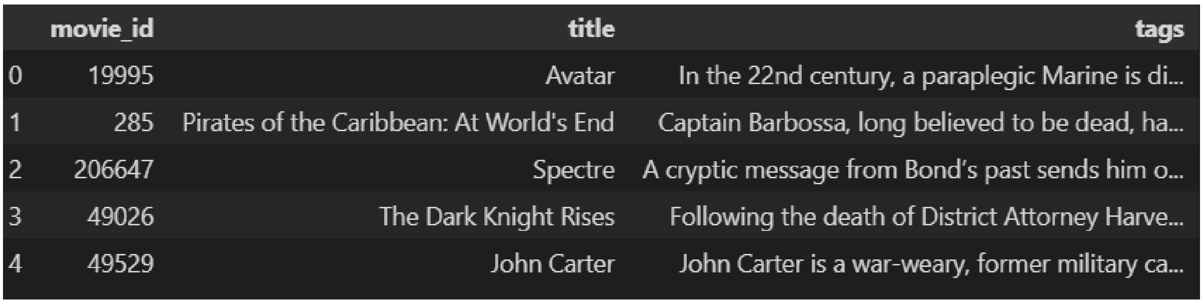
  All the data is available in different forms , but to use it we need to make one big column with all data in form of string separated by space .This was achieved with the help of pandas.

  ★ Data in form usable form



	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCameron]
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbinski]
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMendes]
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman]	[ChristopherNolan]
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[AndrewStanton]

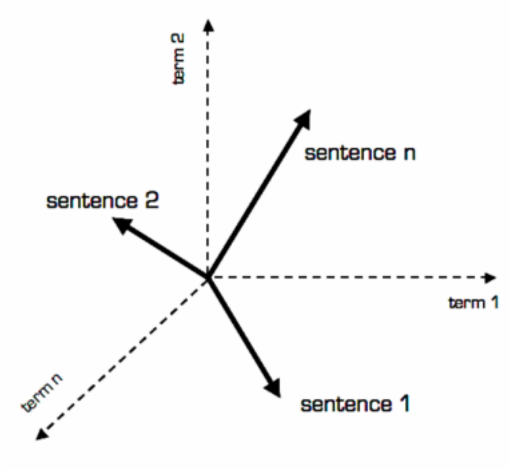
  ★ Making one big column



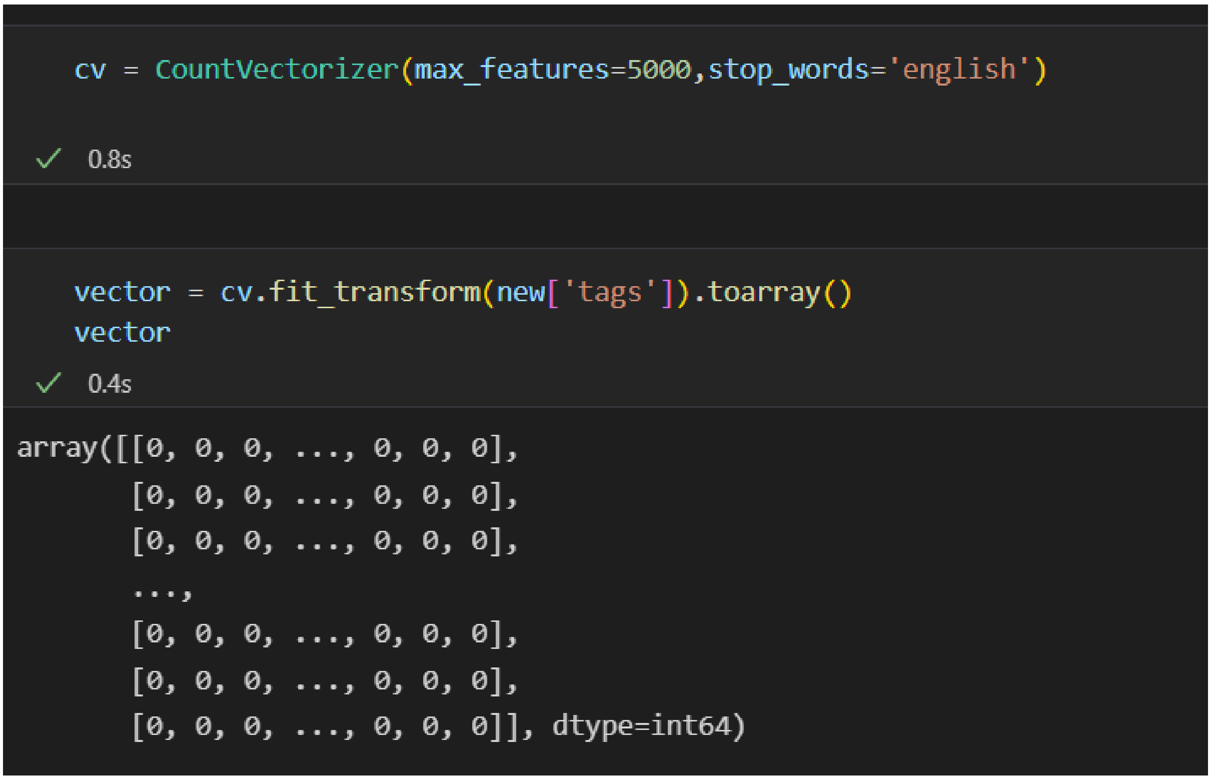
	movie_id	title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...
2	206647	Spectre	A cryptic message from Bond's past sends him o...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...
4	49529	John Carter	John Carter is a war-weary, former military ca...

- **VECTOR SPACE MODEL:**

  It is very important to get the idea of document representation. Assume there are several documents and each document consists of one single sentence. Then we can represent the document as a model in Figure, which is called Vector Space Model. Consider each feature of a movie as a term, then a feature can represented by this model. It is obviously that some features are more important than others and the importance is the weight in the similarity calculation.



Here we converted tags into vector with the help of countvectorizer function of sklearn library



```
cv = CountVectorizer(max_features=5000,stop_words='english')
```

✓ 0.8s

```
vector = cv.fit_transform(new['tags']).toarray()
vector
```

✓ 0.4s

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

- **SIMILARITY:**

  Calculate the similarity between movies is the objective of content-based recommender systems. The content can be anything such as text, video and image. In our project, each movie is represented by a feature vector. I will introduce the cosine similarity algorithm then. Cosine Similarity is the most popular measurement for document similarity. In order to calculate the similarity between two features, we can calculate the cosine of the angle between the feature vector using Equation.

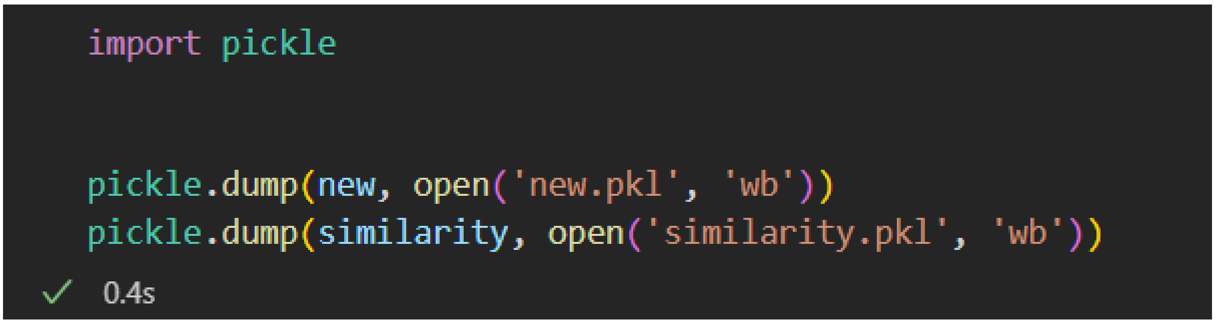
$$similarity = \cos(\theta) = \frac{A \cdot B}{||A||||B||} = \frac{\sum_{n}^{i=1} A_i \times B_i}{\sqrt{\sum_{n}^{i=1} (A_i)^2} \times \sqrt{\sum_{n}^{i=1} (B_i)^2}}$$

Implementation in code-

bl

- **<u>Dumping model using pickle:</u>**

To save the model all we need to do is pass the `model` object into the `dump()` function of Pickle. This will serialize the object and convert it into a "byte stream" that we can save as a file called `model.pkl`. You can then store this model and run it on unseen test data without the need to re-train the model again from scratch.



```
import pickle


pickle.dump(new, open('new.pkl', 'wb'))
pickle.dump(similarity, open('similarity.pkl', 'wb'))
```

✓ 0.4s

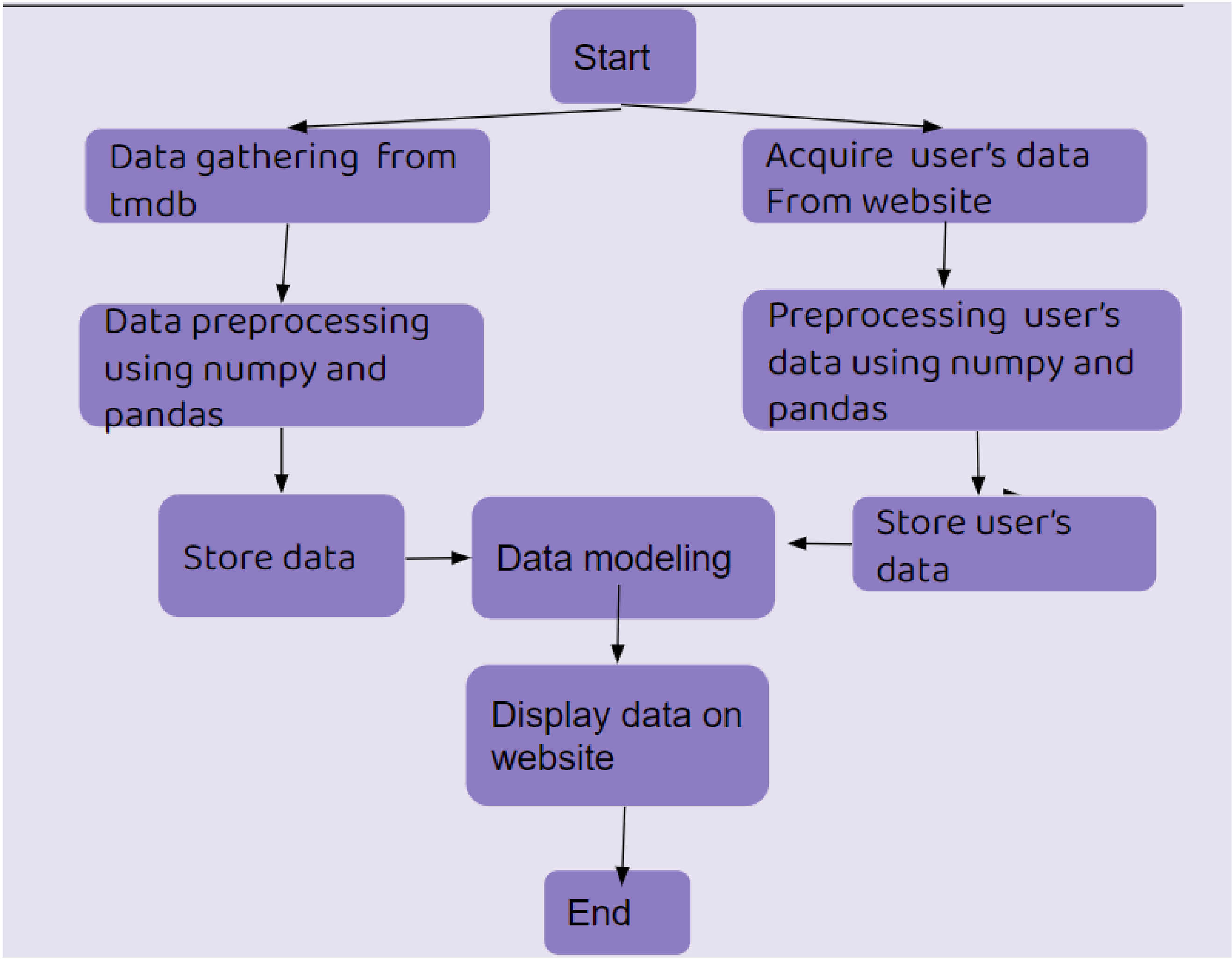
Content based implementation

Content based implementation Data Scrapping. For this algorithm, we needed to obtain movie metadata. While some basic movie metadata, such as release year, tags and genre were provided in the set, we decided to collect further information about the movie, with hopes of incorporating them into our system. We decided to obtain the required information from the TMDB website. The pages content were accessed using the Beautiful Soup2 python library. Beautiful Soup is a Python library designed for quick turnaround projects like screen-scraping. It allows us to easily search and navigate the pages content. Once the structure of the website was identified, we wrote a script to automatically extract the relevant information about a web-page and create a dump. In python3.

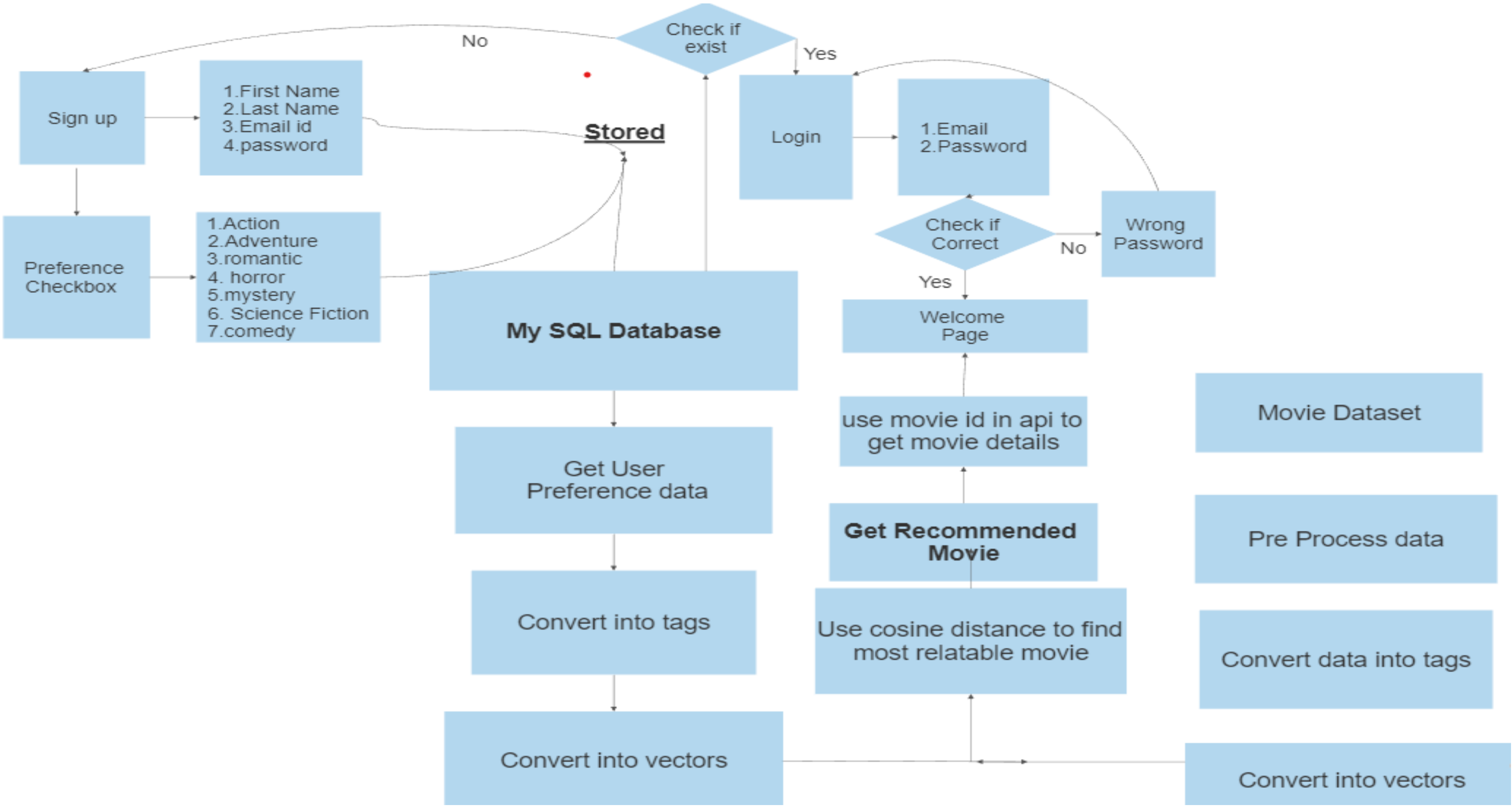
Movie Plot keyword search

We also implemented an additional feature that provides a distributed indexer on the movie synopsis as extracted from the TMDB website to allow searching for movies based on keyword searches. It is built on a MapReduce framework to build an inverted index. The framework for the indexer is the same as the one developed earlier in the semester for the assignments. It has been modified to allow searching on the movie information dataset. First, the dataset is generated by scrapping data from the IMDB pages of the respective movies. The IMDB movie id is provided along with the movie lens dataset. After accessing the webpage, the metadata from the movie is collected and then pickled and stored in a dump. This same dump can also be used to implement and extend the content based model. In the current model, the synopsis refers to the first of the provided plot summaries. The summaries tend to be much shorter than the synopsis and can be extracted much more easily. Also, the size of the dump increased by a factor of almost 100 when using synopses instead of plot summaries. One more reason to select plot summaries is that detailed synopsis are not available for a large portion of the movies, which would result in a bias in the search.

Data Flow


Start
Data gathering from tmdb
Acquire user's data From website
Data preprocessing using numpy and pandas
Preprocessing user's data using numpy and pandas
Store data
Data modeling
Store user's data
Display data on website
End

Graphical Explanation.



CHAPTER 6: RESULT AND CONCLUSION

Recommender system has developed for many years, which ever entered a low point. In the past few years, the development of machine learning, large-scale network and high performance computing is promoting new development in this field. We will consider the following aspects in future work.

- Use collaborative filtering recommendation. After getting enough user data, collaborative filtering recommendation will be introduced. Collaborative filtering is based on the social information of users, which will be analyzed in the future research

- Introduce more precise and proper features of movie. Typical collaborative filtering recommendation use the rating instead of object features. In the future we should extract features such as color and subtitle from movie which can provide a more accurate description for movie.
- Introduce user dislike movie list. The user data is always useful in recommender systems. In the future we will collect more user data and add user dislike movie list. We will input dislike movie list into the recommender system as well and generate scores that will be added to previous result. By this way we can improve the result of recommender system.
- Introduce machine learning. For future study, dynamic parameters will be introduced into recommender system, we will use machine learning to adjust the weight of each feature automatically and find the most suitable weights.
- Make the recommender system as an internal service. In the future, the recommender system is no longer a external website that will be just for testing. We will make it as an internal APIs for developers to invoke. Some movie lists in the website will be sorted by recommendation.

REFERENCES

[1] A Survey of Collaborative Filtering Techniques; Su et al; https://www.hindawi.com/journals/aai/2009/421425/

[2] Google News Personalization: Scalable Online Collaborative Filtering; Das et al; https://www2007.org/papers/paper570.pdf

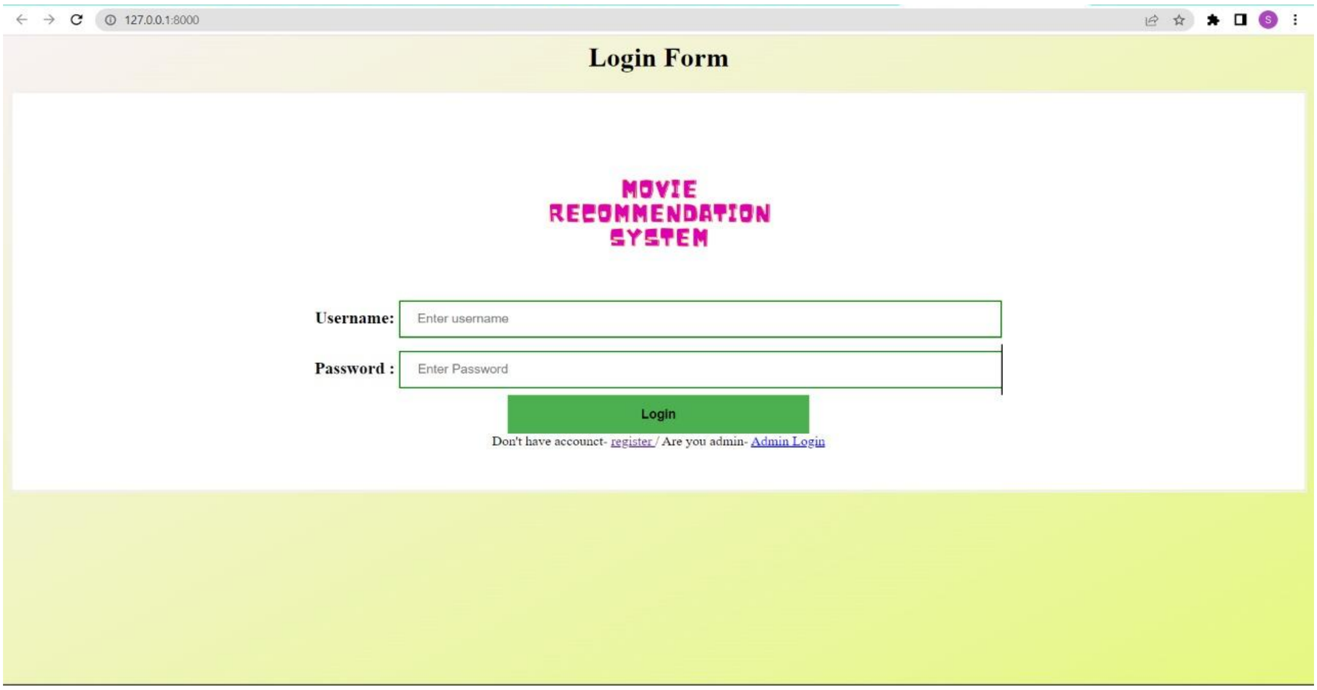
[3] Intro to Recommender Systems: Collaborative Filtering; http://blog.ethanrosenthal.com/2015/11/02/intro-to-collaborative -filtering/

[4] Collaborative Filtering Recommender Systems; Stanford Student project; http://cs229.stanford.edu/proj2014/Rahul %20Makhijani,%20Saleh%20Samaneh,%20Megh%20Mehta,%20 Collaborative%20Filtering%20Recommender%20Systems.pdf [

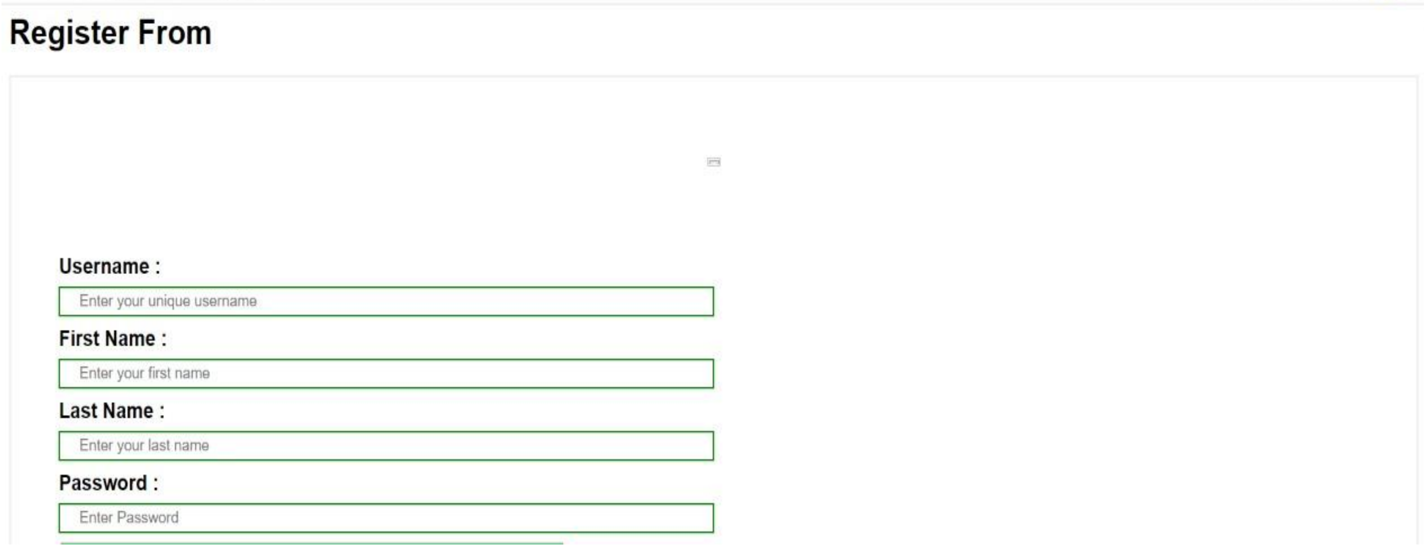
5] MMDS course slides; Jeffrey Ullman; http://infolab.stanford.edu/ ullman/mmds/ch9.pdf

[6] Recommending items to more than a billion people; Kabiljo et al; https://code.facebook.com/posts/861999383875667/ recommending-items-to-more-than-a-billion-people/

1.LOGIN PAGE



Don't have account- register / Are you admin- Admin Login



Register From

Username :

Enter your unique username

First Name :

Enter your first name

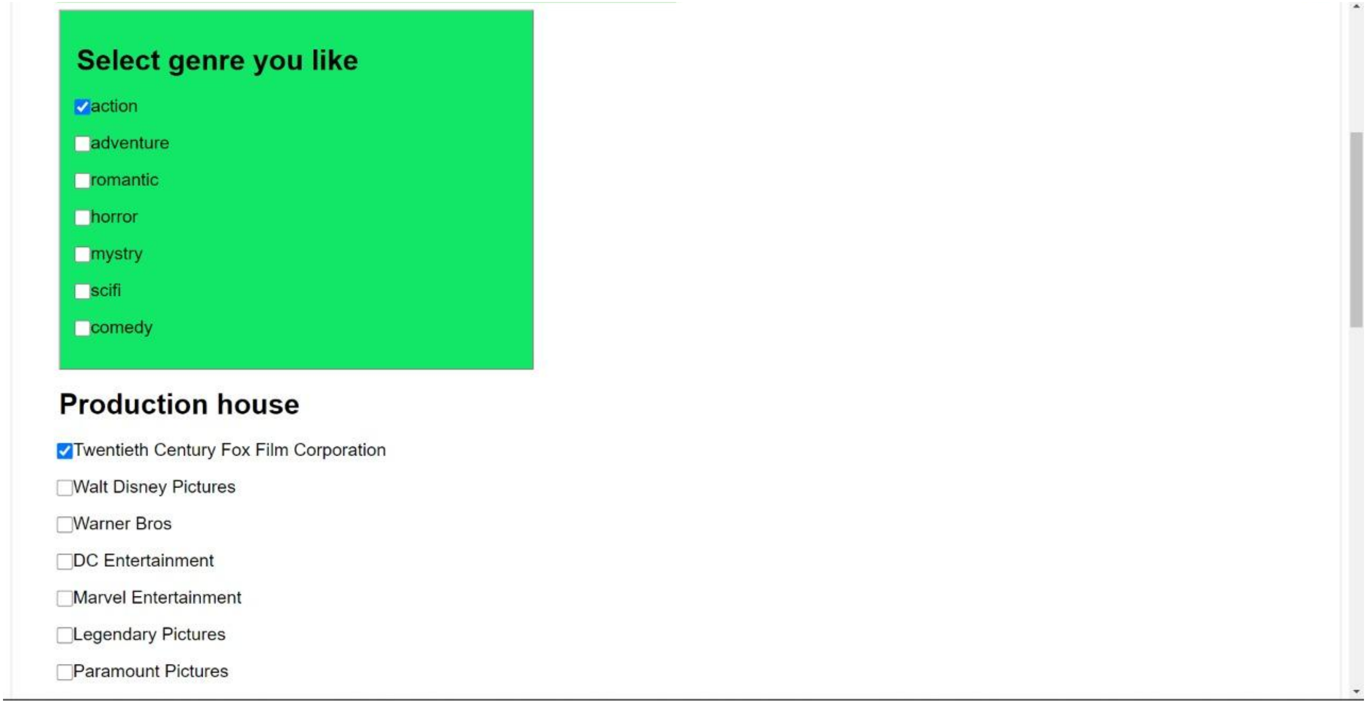
Last Name :

Enter your last name

Password :

Enter Password

3.TAKING USER INPUT



Select genre you like

☑action

☐adventure

☐romantic

☐horror

☐mystry

☐scifi

☐comedy

Production house

☑Twentieth Century Fox Film Corporation

☐Walt Disney Pictures

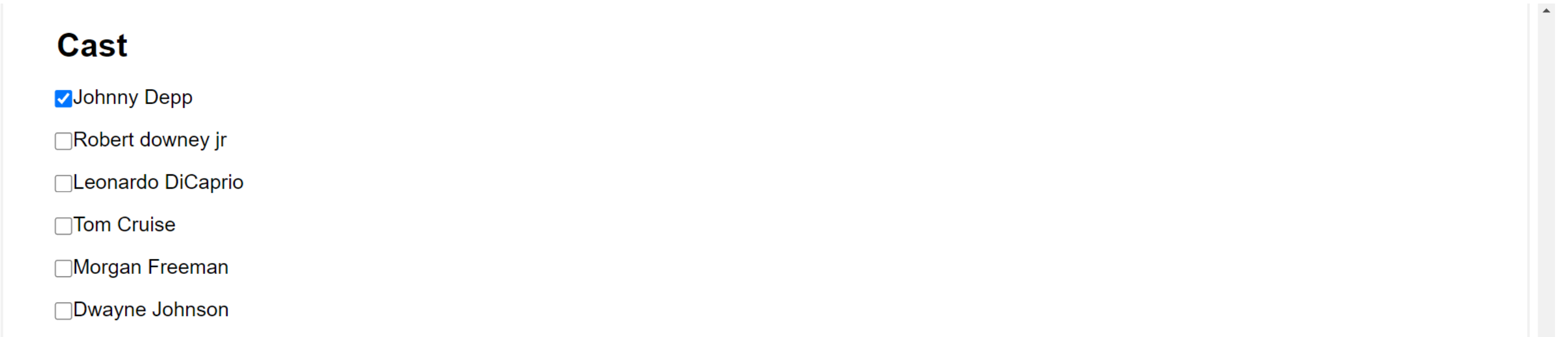
☐Warner Bros

☐DC Entertainment

☐Marvel Entertainment

☐Legendary Pictures

☐Paramount Pictures



Cast

☑Johnny Depp

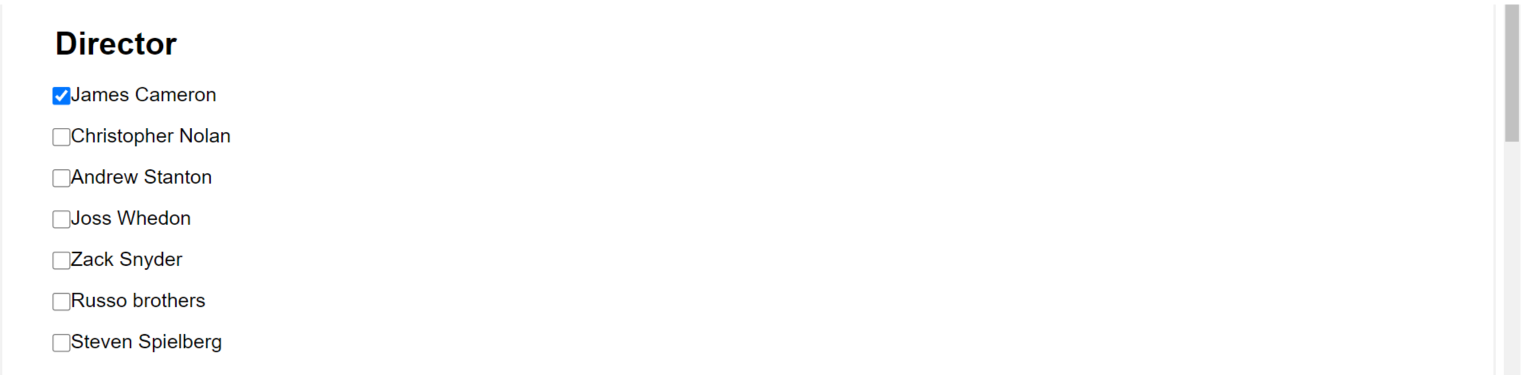
☐Robert downey jr

☐Leonardo DiCaprio

☐Tom Cruise

☐Morgan Freeman

☐Dwayne Johnson



Director

☑ James Cameron

□ Christopher Nolan

□ Andrew Stanton

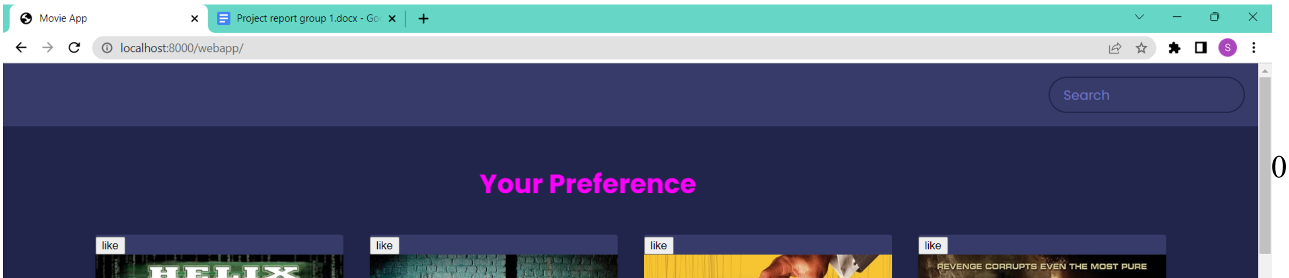
□ Joss Whedon

□ Zack Snyder

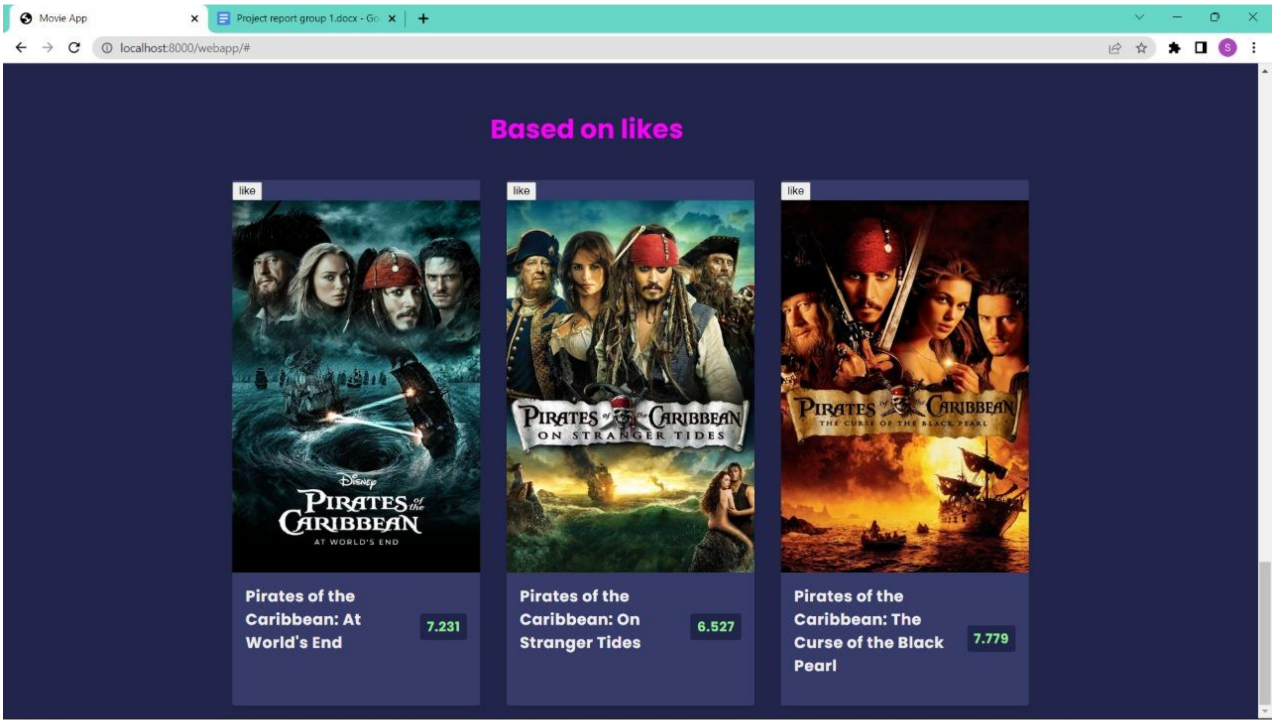
□ Russo brothers

□ Steven Spielberg

4.RECOMMENDATION BASED ON REFERENCE



5.RECOMMENDATION BASED ON LIKES



Based on likes

Pirates of the Caribbean: At World's End 7.231

Pirates of the Caribbean: On Stranger Tides 6.527

Pirates of the Caribbean: The Curse of the Black Pearl 7.779