

A Project Report on

## **Movie Recommendation System**

Submitted in partial fulfillment of the requirements for the award  
of the degree of

**Bachelor of Engineering**

in

**Computer Science**

by

**Mirza Zulfiqar Ali Jaffer Ali(18102043)  
Vandan Jain (18102031)  
Jeel Jain (18102041)  
Yashi Rathod(18102039)**

Under the Guidance of

**Dr.Rahul Ambekar**



**Department of Computer Science  
NBA Accredited**

A.P. Shah Institute of Technology  
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615  
UNIVERSITY OF MUMBAI  
**Academic Year 2020-2021**

## **Approval Sheet**

This Project Report entitled "**Movie Recommendation System**" Submitted by **"Zulfiqar Ali Mirza "(18102043), "Vandan Jain"(18102031), "Jeel Jain"(18102041)**,  
**,"Yashi Rathod"(18102039)**is approved for the partial fulfillment of the requirement  
for the award of the degree of **Bachelor of Engineering** in **Computer Science** from  
**University of Mumbai**.

(Dr.Rahul Ambekar)  
Guide

Prof. Sachin Malave  
Head Department of Computer Science

Place:A.P.Shah Institute of Technology, Thane  
Date:

## **CERTIFICATE**

This is to certify that the project entitled "***Movie Recommendation System***" submitted by "***Mirza Zulfiqar Ali Jaffer Ali*** (18102043), "***vandan Jain*** (18102031), "***Jeel jain*** (18102041), "***Yashi rathod*** (18102039) for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Science***, to the University of Mumbai, is a bonafide work carried out during academic year 2020-2021.

(Dr.Rahul Ambekar)  
Guide

Prof. Sachin Malave  
Head Department of Computer Science

Dr. Uttam D.Kolekar  
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane

Date:

## **Acknowledgement**

We have great pleasure in presenting the report on **Movie Recommendation System**. We take this opportunity to express our sincere thanks towards our guide **Guide Dr.Rahul Ambekar** Department of CS, APSIT thane for providing the technical guide-lines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin Malave** Head of Department,CS, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol Kalugade** BE project co-ordinator, Department of CS, APSIT for beinhttps://www.overleaf.com/project/61879aa480dc35253506d624g encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

**Student Name1:Mirza Zulfiqar Ali Jaffer Ali**

**Student ID1:18102043**

**Student Name2:Vandan Jain**

**Student ID2:18102031**

**Student Name3:Jeel Jain**

**Student ID3:18102041**

**Student Name4:Yashi Rathod**

**Student ID4:18102039**

## **Declaration**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

---

(Signature)

(Mirza Zulfiqar Ali Jaffer Ali 18102043)  
(Vandan Jain 18102031)  
(Jeel Jain 18102041)  
(Yashi Rathod 18102039)

Date:

# Contents

<b>1 Project Concept</b>	<b>1</b>
1.1 Abstract . . . . .	1
1.2 Introduction . . . . .	2
1.3 Objectives . . . . .	3
1.4 Literature Review . . . . .	4
1.5 Problem Definition . . . . .	6
1.6 Scope . . . . .	7
1.7 Technology Stack . . . . .	8
1.8 Benefits for Environment And Society .....	11
<b>2 Project Design</b>	<b>12</b>
2.1 Proposed System.....	12
2.2 Design (Flow of modules) .....	13
2.3 Description of use case diagram.....	14
2.4 Class Diagram .....	15
2.5 Modules.....	16
2.5.1 Module 1:-Data set gathering and analysis.....	16
2.5.2 Module 2:-We will conduct exploratory data analysis (EDA) on the data in order to get the necessary information for autocompletion .....	17
2.5.3 Module 3:-We will explore tmdb api for api calls, to get data of the movies , posters genre, reviews .....	18
2.5.4 Module 4:-Web Scrapping And fetch Reviews .....	19
2.5.5 Module 5:-Build a web site for deployment of the model and integrate the same using flask so that it can interact with the model .....	20
2.5.6 Module 6:- Deployment of the System on AWS Beanstalk .....	21
<b>3 Implementation</b>	<b>22</b>
3.1 Proposed System-Implementation .....	22
3.1.1 Algorithm.....	23
3.1.2 Pseudo code .....	24
3.1.3 Platforms For Execution .....	25
<b>4 Results</b>	<b>26</b>
4.0.1 Expected Output .....	26
<b>5 Conclusion</b>	<b>32</b>
5.1 Future Scope.....	33

<b>6 References</b>	<b>34</b>
<b>7 Bibliography</b>	<b>36</b>

# **Chapter 1**

## **Project Concept**

### **1.1 Abstract**

A recommendation system is a system that provides suggestions to users for certain resources like books, movies, songs, etc., based on some data set. Movie recommendation systems usually predict what movies a user will like based on the attributes present in previously liked movies. Finding useful information is becoming increasingly difficult as the Internet grows at a quicker rate. Information or content can take any form that users find appealing, such as music, video, photos, or text. As a result, recommendation algorithms come into play. Recommendation systems are a type of information filtering system that assists people in locating products, accurate information, and even other people. Books, news, articles, music, videos, and movies are among the most prominent domains where recommender systems are used. A lot of factors can be considered while designing a movie recommendation system like the genre of the movie, actors present in it or even the director of the movie. The systems can recommend movies based on one or a combination of two or more attributes. In our project the recommendation system has been built on these various types of attribute that the user might prefer to watch. We suggested a movie recommendation system based on a content-based filtering strategy that takes the information provided by users, analyses it, and then suggests the movies that are most suited to the user at that time.

## **1.2 Introduction**

Recommendation systems are crucial in information filtering systems because they determine how best to recommend things or information that are relevant to the user. A recommendation system, also known as a recommender system, can be used in a variety of situations. Recommender Systems make recommendations, which the user can accept or reject, as well as providing implicit or explicit feedback, either immediately or afterwards. 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. User's actions and feedback can be saved in the recommender database, which can then be utilized to generate new recommendations in subsequent user-system interactions. The user experience is enhanced with high-quality tailored recommendations. Recently, web-based personalized recommendation systems have been used to deliver numerous forms of personalized information to their users. These are the systems in place. These systems can be applied in various types of applications and are very common now a day.

We can classify the recommender systems in two broad categories :-

- a) Collaborative filtering approach
- b) Content-based filtering approach

## **1.3 Objectives**

The main objective of this project is to make a Movie Recommendation System which generally provides movie recommendations on the basis of content based filtering

1. Content Based Filtering:- Content-based filtering is a type of recommender system that attempts to guess what a user may like based on that user's activity. Content-based filtering makes recommendations by using keywords and attributes assigned to objects in a database. It Selects items based on the correlation between the content of the items and the user's preferences
2. Sentiment Analysis:- Existing recommendation algorithms usually do not consider sentiment factors and only consider quantitative scoring data, but this type of data is sparse when there are a large number of products and users. In various systems that apply recommendation technology, user reviews construct a component that cannot be ignored. Quantifying user reviews through sentiment analysis technology makes up for the biggest shortcoming of existing recommendation algorithms.

Hence, We develop a recommendation system on both Content based filtering and Sentiment Analysis techniques in python which generally takes inputs from the user and recommends top 10 movies as recommendation to the user or client.

## **1.4 Literature Review**

Paper[1]:-

In this study, we present a review of the relevant literature that employs a variety of methodologies to make various recommendations systems. Collaborative filtering is classified in this study utilizing several methodologies such as matrix factorization, user-based recommendation, and item-based recommendation. This survey also provides a roadmap for future research in this field. Several strategies for making movie suggestions have been intensively researched during the last few decades. A recommendation system based on the ALS algorithm, a recommendation based on the weighting technique, and collaborative filtering based on item similarity are some examples. These strategies necessitate prior knowledge of the user-generated movie ratings. For the most part, movie lens datasets are used to evaluate these approaches. These systems, however, are not entirely accurate, and research is ongoing to improve them.

Paper[2]:-

The system recommends an item to the user for the decision making. Then separate the movie data set into an unrated and rated sample set with the help of the KNN model. It can recommend the movies to the unseen users via user registration information, and it can create new and not popular movie recommendations according to the film's history and score. The database in this approach is the MYSQL database.

Paper[3]:-

CBRS (content-based recommendation system) suggests things to the user based on their previous purchases. In order to give relevant information to the user or buyer, a user profile with traits and features of the objects must be developed utilizing online usage mining or information retrieval methods. The similarity of the materials the user is interested in is used to filter things in the content recommendation system. It suggests or analyses goods based on high-rated items that are comparable to the user's preferences. Documents with textual information, websites, and other applications of content-based recommendation

#### Paper[4]:-

People rely on knowledge to decide their interests therefore the recommendation system is ingrained in their daily lives. The collaborative filtering methodology uses information from a user's previous actions (for example, previously purchased things or numerical ratings given to them) as well as similar judgments made by other users. Following that, several models are employed to predict which products (or ratings for objects) the user might be interested in. Despite the fact that there have been numerous approaches created in the past. Search, on the other hand, continues to exist since it is widely employed in many applications that customize recommendations and cope with inaccuracy. These requirements present considerable difficulties.

#### Paper[5]:-

K-Means Algorithm-Based Research Paper:- We choose K initial centroids in the K-means clustering algorithm, where K is the desired number of clusters. The cluster with the nearest mean, i.e. the cluster's centroid, is then assigned to each point. The centroid of each cluster is then updated based on the points allocated to the cluster. We continue the operation till the cluster centre does not change (centroid). Finally, the goal of this technique is to minimize an objective function, in this case, a squared error function.

#### Paper[6]:-

A utility-based system is a function that expresses the degree to which someone is happy. It calculates the usefulness of each attribute of the user to recommend things and determines the users contentment. The user profile is the system's derived utility function for the user, and the system uses constraint satisfaction techniques to find the best match. The advantage of utility-based recommendation is that it may incorporate non-product qualities like vendor reliability and product availability into the utility calculation, allowing users to trade off price versus delivery schedule for example. The disadvantages of the new user problem, data sparsity, are alleviated using utility-based techniques. It also determines the users satisfaction and makes computation based on utility of each attribute of the user to recommend items. This technique can be applied to applications where short term goals need to be achieved.

## **1.5 Problem Definition**

This problem introduces a Movie Recommendation System that provides related content out of relevant and irrelevant collection of items to the users. Its main aim is to recommend movie to users based on the attributes that will be provided by the users. Our user will enter name of the movie they have watched or like to get information for that movie. Our web app will display all information about the movie along with the poster, reviews, names of the top casts also we will get some user reviews and perform sentiment analysis on that. At last we will have the recommended list of movie based on the movie entered by the user based on the reviews about a particular movie, similar content based and sentiment analysis based movies are recommended.

## **1.6 Scope**

A movie recommendation system , based on collaborative filtering approach makes use of the information provided by users, analyzes them and then recommends the movie that is best suited to the user at that time also by Using Sentiment Analysis.Recommender systems open new opportunities of retrieving personalized information on the Internet. It also helps to alleviate the problem of information overload which is a very common phenomenon with information retrieval systems and enables users to have access to products and services which are not readily available to users on the system. We come up with a strategy that focuses on dealing with user's personal interests and based on his previous reviews, movies are recommended to users. This strategy helps in improving accuracy of the recommendations.

## **1.7 Technology Stack**

**Rest API:-**

Representational State Transfer (REST) is an architectural style that defines a set of constraints to be used for creating web services. REST API is a way of accessing web services in a simple and flexible way without having any processing. REST technology is generally preferred to the more robust Simple Object Access Protocol (SOAP) technology because REST uses less bandwidth, simple and flexible making it more suitable for internet usage. It's used to fetch or give some information from a web service. All communication done via REST API uses only HTTP request. An API is a set of definitions and protocols for building and integrating application software. It's sometimes referred to as a contract between an information provider and an information user—establishing the content required from the consumer (the call) and the content required by the producer (the response). For example, the API design for a weather service could specify that the user supply a zip code and that the producer reply with a 2-part answer, the first being the high temperature, and the second being the low.

**TMDB:-**

Our API is available for everyone to use. A TMDB user account is required to request an API key. Professional users are approved on a per application basis. As always, you must attribute TMDB as the source of your data. The API service is for those of you interested in using our movie, TV show or actor images and/or data in your application. Our API is a system we provide for you and your team to programmatically fetch and use our data and/or images. The API provides a fast, consistent and reliable way to get third party data.

**Jupyter Notebook:-**

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality. The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience.

**HTML:-**

HTML stands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page. HTML consists of a series of elements. HTML elements tell the browser how to display the content. HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

### **CSS:-**

CSS is the language we use to style an HTML document. CSS describes how HTML elements should be displayed. This tutorial will teach you CSS from basic to advanced. Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media. CSS is used for defining the styles for web pages. It describes the look and formatting of a document which is written in a markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces.

### **JavaScript:-**

JavaScript is a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc. — you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area. It is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Okay, not everything, but it is amazing what you can achieve with a few lines of JavaScript code.)

### **Python:-**

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems. This versatility, along with its beginner-friendliness, has made it one of the most-used programming languages today. A survey conducted by industry analyst firm RedMonk found that it was the second-most popular programming language among developers in 2021

### **Flask:-**

Flask is considered more Pythonic than the Django web framework because in common situations the equivalent Flask web application is more explicit. Flask is also easy to get started with as a beginner because there is little boilerplate code for getting a simple app up and running. An API of Python used to build web-applications(front - end) Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

### Pandas:-

Python library for data manipulation and analysis. Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. As one of the most popular data wrangling packages, Pandas works well with many other data science modules inside the Python ecosystem, and is typically included in every Python distribution, from those that come with your operating system to commercial vendor distributions like ActiveState's ActivePython.

### AJAX:-

Ajax stands for Asynchronous Javascript And Xml. Ajax is just a means of loading data from the server and selectively updating parts of a web page without reloading the whole page. Basically, what Ajax does is make use of the browser's built-in XMLHttpRequest (XHR) object to send and receive information to and from a web server asynchronously, in the background, without blocking the page or interfering with the user's experience. Ajax has become so popular that you hardly find an application that doesn't use Ajax to some extent. The example of some large-scale Ajax-driven online applications are: Gmail, Google Maps, Google Docs, YouTube, Facebook, Flickr, and so many other applications.

### NLTK:-

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. Natural Language Processing with Python provides a practical introduction to programming for language processing. Written by the creators of NLTK, it guides the reader through the fundamentals of writing Python programs, working with corpora, categorizing text, analyzing linguistic structure, and more. The online version of the book has been updated for Python 3 and NLTK 3.

### Scikit-Learn:-

Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. Think of any supervised machine learning algorithm you might have heard about and there is a very high chance that it is part of scikit-learn. Starting from Generalized linear models (e.g Linear Regression), Support Vector Machines (SVM), Decision Trees to Bayesian methods – all of them are part of scikit-learn toolbox.

## **1.8 Benefits for Environment And Society**

Increased sales/conversion:-

There are very few ways to achieve increased sales without increased marketing effort. Once you setup an automated recommendation system, you get recurring additional sales without any effort.

Increased user satisfaction:-

Shortest path to a sale is great both for you and your user reducing their effort. Recommendation systems allow you to reduce your user's path to a sale by recommending them an appropriate option sometimes even before they search for it.

Reduced churn:-

Recommendation system powered emails are one of the best ways to re-engage customers. Discounts or coupons are other effective yet costly ways of re-engaging customers and they can be coupled with recommendations to increase customer's probability of conversion.

# **Chapter 2**

## **Project Design**

### **2.1 Proposed System**

Our user will enter name of the movie they have watched or like to get information for that movie. Our web app will display all information about the movie along with the poster, reviews, names of the top casts also we will get some user reviews and perform sentiment analysis on that. At last we will have the recommended list of movie based on the movie entered by the user. User can click on the list of casts to get some basic details on them, also user can click on the recommended movies to get more info on that movie which again will give us list of recommended movies for the movie user clicked We are going to Perform Exploratory Data Analysis (EDA) on the data Gathering the appropriate data for the model. Build the Movie Recommendation System Testing the model on different parameters. Get recommendations for the movies. Build a web site for deployment of the model and integrate the same using flask. Lastly Testing and improvement will be done.

## **2.2 Design (Flow of modules)**

- 1) Data set gathering and analysis:- We will Gather the datasets from Tmdb and do the analysis and churn the data for the further process
- 2) Exploratory Data Analysis:- We will conduct exploratory data analysis (EDA) on the data in order to get the necessary information for autocompletion.
- 3) Dataset:- We will explore tmdb api for api calls, to get data of the movies , posters genre, reviews.
- 4) Web Scrapping And fetch Reviews:- We used beautifulsoup for web scrapping and fetching the data.
- 5) Deployment on Aws :- Build a web site for deployment of the model and integrate the same using flask so that it can interact with the model.

## 2.3 Description of use case diagram

The purpose of a recommendation system basically is to search for content that would be interesting to an individual. Moreover, it involves a number of factors to create personalised lists of useful and interesting content specific to each user/individual. We get am Imdb Dataset for evaluating it for the further process.Then we pre-process and web scrapping of the model is done. After the succesful execution we use the user's review to train the data for sentiment analysis.By using flask we design the webpage and develop the backend as well.We use the REST API for firing queries at TMDB API.After all the process is done, we recommend a list of movies on basis of sentiment analysis using TMDB API.

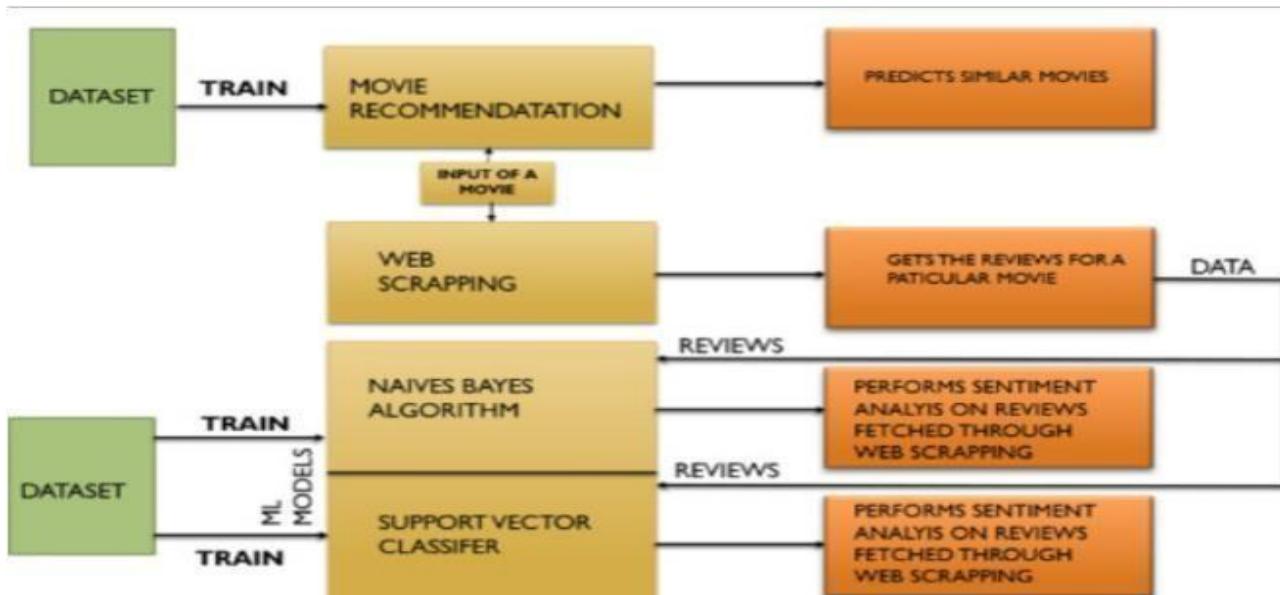


Fig 2.3 Use Case Diagram

## 2.4 Class Diagram

The class diagram provides the detailed insight of how the project is implemented.

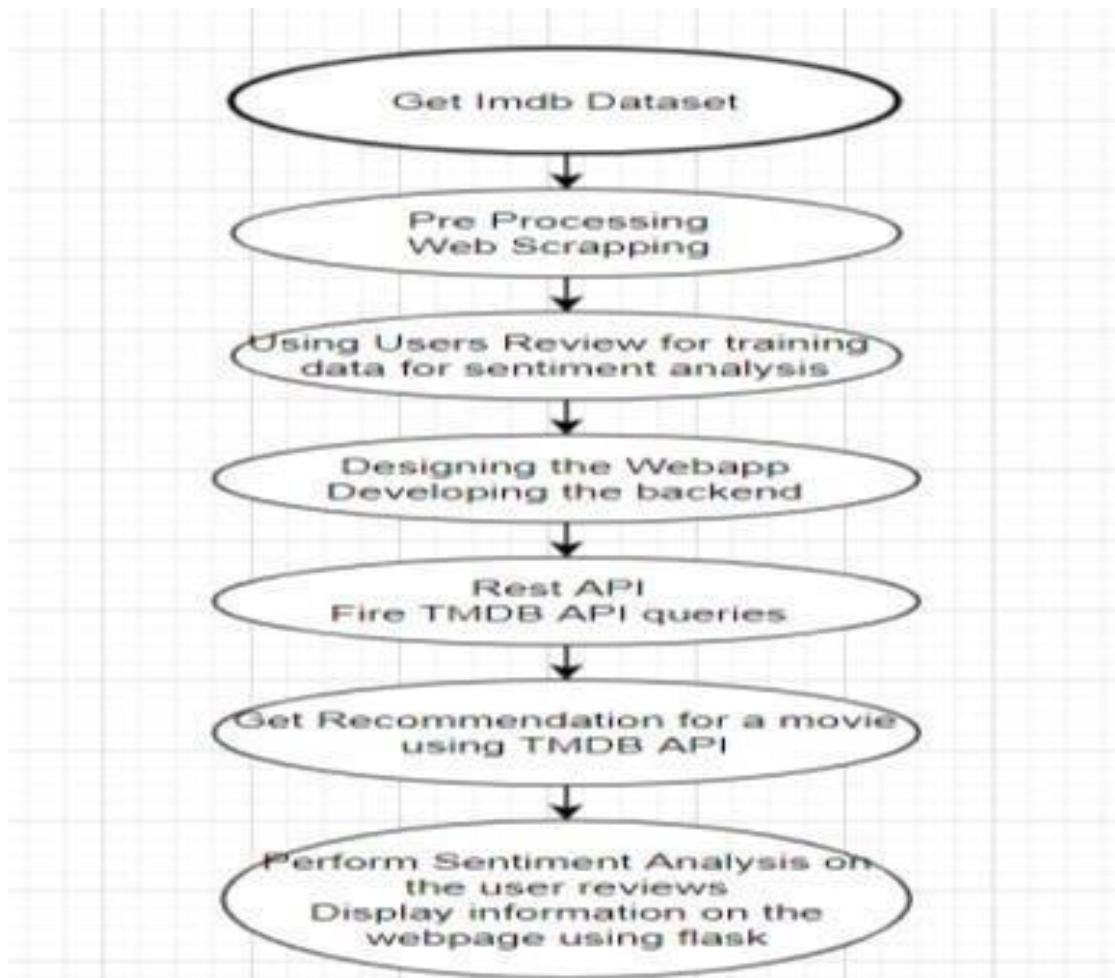


Fig2.4ClassDiagram

## **2.5 Modules**

### **2.5.1 Module 1:-Data set gathering and analysis.**

Prediction is done through multiple servers. Servers have some available memory but are considered stateless from a user data perspective. This means that users may interact with different machines during a session and user data should be available regardless of which machine the user has landed on. All metadata attached to articles and recommended items (such as classification, article text etc.) is available both online and offline. However, fetching this data requires a db call. Some user activity needs to be available for inference fairly quickly while other activity may be available few hours after it happened. For instance, after a user clicks on a recommendation, we would like to make this information available as soon as possible. On the other hand, it's ok to have longer term browsing behavior data available for inference hours after the user has browsed that content. Data for all users is too large to hold in memory while training.

## **2.5.2 Module 2:-We will conduct exploratory data analysis (EDA) on the data in order to get the necessary information for autocompletion.**

Exploratory data analysis is an approach of analyzing data sets to summarize their main characteristics, often using statistical graphics and other data visualization methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling and thereby contrasts traditional hypothesis testing. Exploratory data analysis has been promoted by John Tukey since 1970 to encourage statisticians to explore the data, and possibly formulate hypotheses that could lead to new data collection and experiments. EDA is different from initial data analysis (IDA),[1][2] which focuses more narrowly on checking assumptions required for model fitting and hypothesis testing, and handling missing values and making transformations of variables as needed. EDA encompasses IDA.

### **2.5.3 Module 3:-We will explore tmdb api for api calls, to get data of the movies , posters genre, reviews.**

The Movie,TV Show ,TV season ,episode and detail methods support a query parameter called append to response. This makes it possible to make sub requests within the same namespace in a single HTTP request. Each request will get appended to the response as a new JSON object. The best part of this is that these requests only count as one request against the rate limits so you can really speed up your experience. Each method will still respond to whatever query parameters are supported by each individual request. This is worth pointing out for images since your language parameter will filter images. This is where the include image language parameter can be useful as outlined in the image language page.

#### **2.5.4 Module 4:-Web Scrapping And fetch Reviews**

Web scraping allows scope for customization that ranges from data extraction process to frequency, format, structure by changing your crawler's user agent. Now, this flexibility is not possible with a website's API. There will be either limited or no customization since the consumer does not have any control over it.

## **2.5.5 Module 5:-Build a web site for deployment of the model and integrate the same using flask so that it can interact with the model.**

Each framework has a different way to put together its routes, models, views, database interaction, and overall application configuration. Flask is considered more Pythonic than the Django web framework because in common situations the equivalent Flask web application is more explicit. Flask is also easy to get started with as a beginner because there is little boilerplate code for getting a simple app up and running. the Flask framework became wildly popular as an alternative to Django projects with their monolithic structure and dependencies.

## **2.5.6 Module 6:- Deployment of the System on AWS Beanstalk**

WS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.

There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

# **Chapter 3**

## **Implementation**

### **3.1 Proposed System-Implementation**

The user will first Think for his preferred movie.

user will then enter name of the movie they have watched or like to get information for that movie.

Our web app will display all information about the movie.

On scrolling upwards we will get to see the Top casts of the movie that the user have found.

then the user will further see reviews of the others users who has watched the movie and perform Sentiment analysis on that .

At last we will have the recommended list of movie based on the movie entered by the user.

User can click on the list of casts to get some basic details on them, also user can click on the recommended movies to get more info on that movie which again will give us list of recommended movies

## I. PROPOSED SYSTEMA BLOCK DIAGRAM

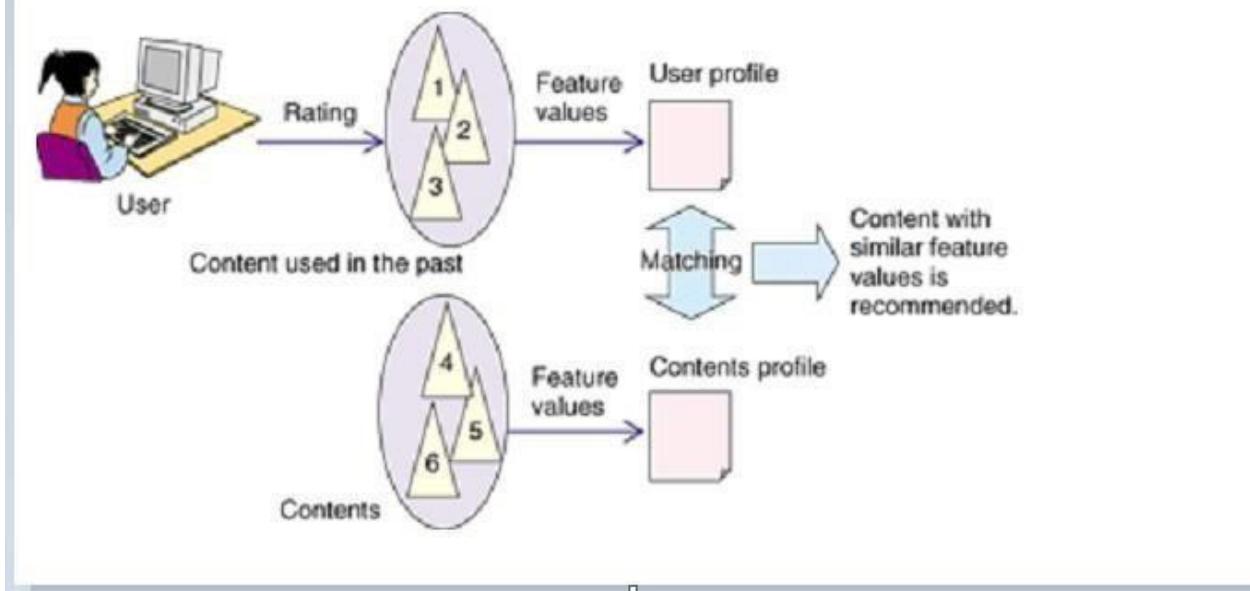


Figure 3.1:Proposed system Block Diagram

### 3.1.1 Algorithm

Nave Bayes algorithm is used to do sentiment analysis (NB). The movie recommendation system now takes a movie title as an input and suggests similar films. The movie's reviews are scraped from the IMDB website and fed into the Sentiment Analysis model, which categorizes them as positive or negative.

The NB Algorithm uses conditional probability to classify the given data set. Bayes theorem is used for the computation and used class levels represented as feature values or vectors of predictors for classification.

A movie title is used as an input to the movie recommendation system, which predicts comparable films. The movie's reviews are acquired via web scraping from the IMDB site and passed to the Sentiment Analysis model for classification as positive or negative. The NB Algorithm uses conditional probability to classify the given data set. Bayes theorem is used for the computation and used class levels represented as feature values or vectors of predictors for classification.

Types of models under NB algorithm

Gaussian

Multinomial

Bernoulli

The multinomial NB model is used in the proposed system to predict the badge of a text, such as an email or a newspaper article. For a given sample, the likelihood of each badge is determined, and the badge with the highest probability is produced. This algorithm was an excellent choice for sentiment analysis of movie reviews because it is primarily utilised for natural language processing and text data analysis.

### **3.1.2 Pseudo code**

- 1) Load the imdb dataset
- 2)Preprocessing of dataset to fetch movie id and movie name for tmdb api and auto-completion respectivielly
- 3)Web scrapping for user's reviews for movies
- 4)Using users review for training data, train the Sentiment analysis model
- 5)Save the model for further use
- 6)Designing of web application
- 7)Developing the backend using Flask
- 8)Using REST API, fire tmdb API queries for fetching details of a movie like Name, Genre, Release Date, Cast, etc.
- 9)Get the recommendation for a movie using tmdb api
- 10)Perform Sentiment analysis on the users reviews.
- 11)Display all the information on the web page using Flask

### **3.1.3 Platforms For Execution**

- a. Python
- b. Jupyter notebook
- c. Visual Studio code
- d. Postman

# **Chapter 4**

## **Results**

### **4.0.1 Expected Output**

- a. Expected output would be the input image
- b. with the help of different methods and algorithms the model will provide the recommendations of movie.

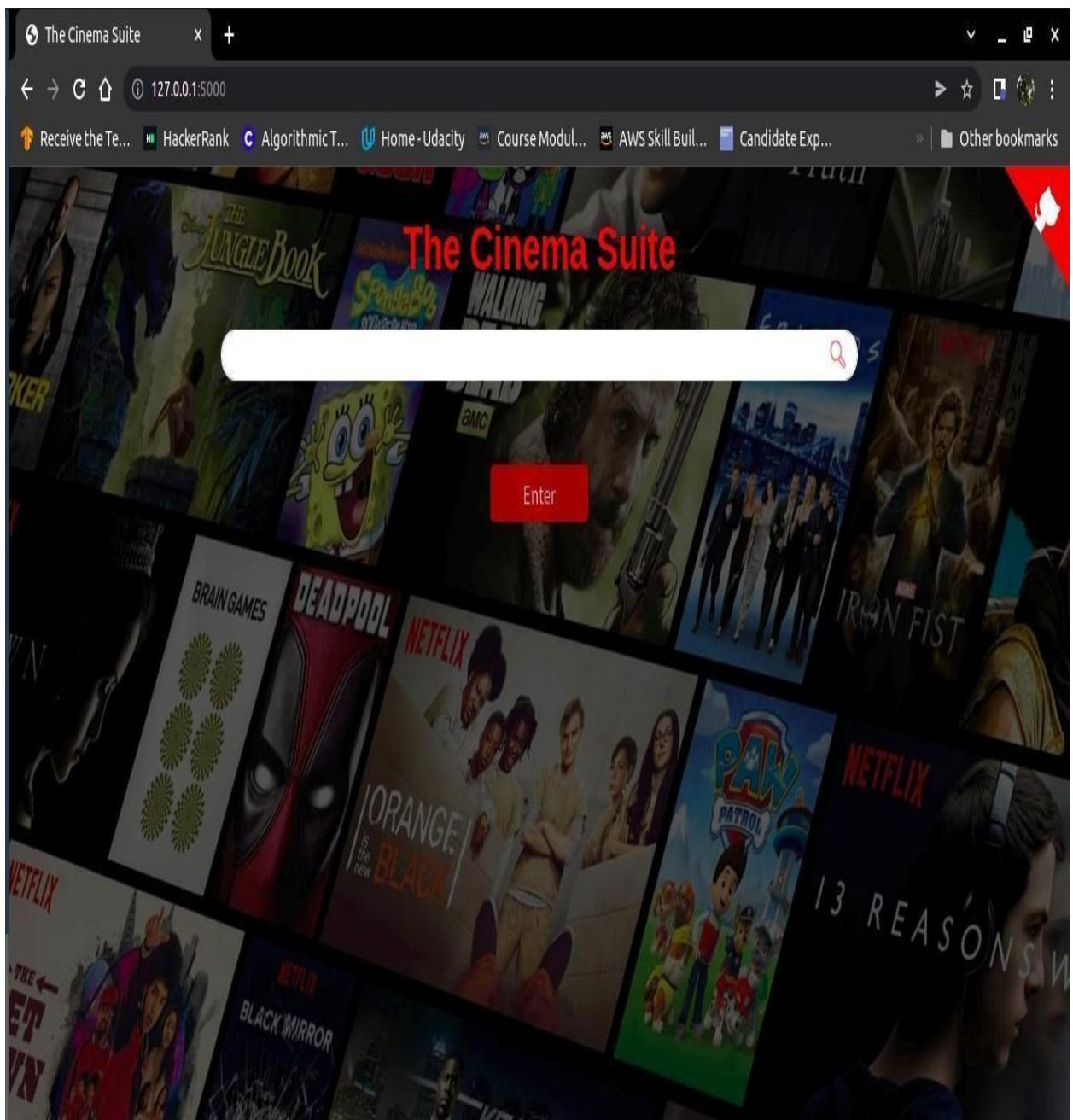


Figure 4.1: input image

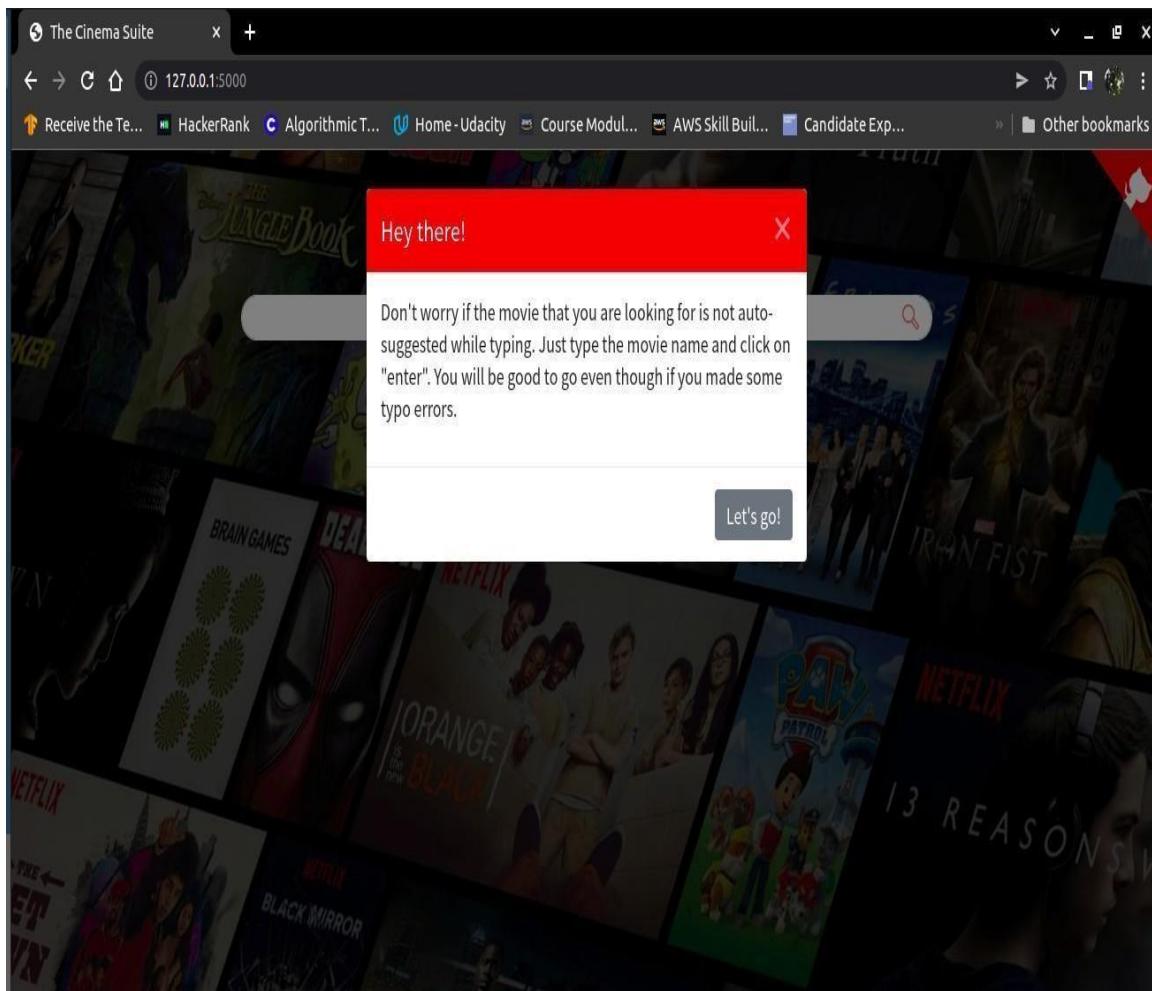


Figure 4.2: input alert box

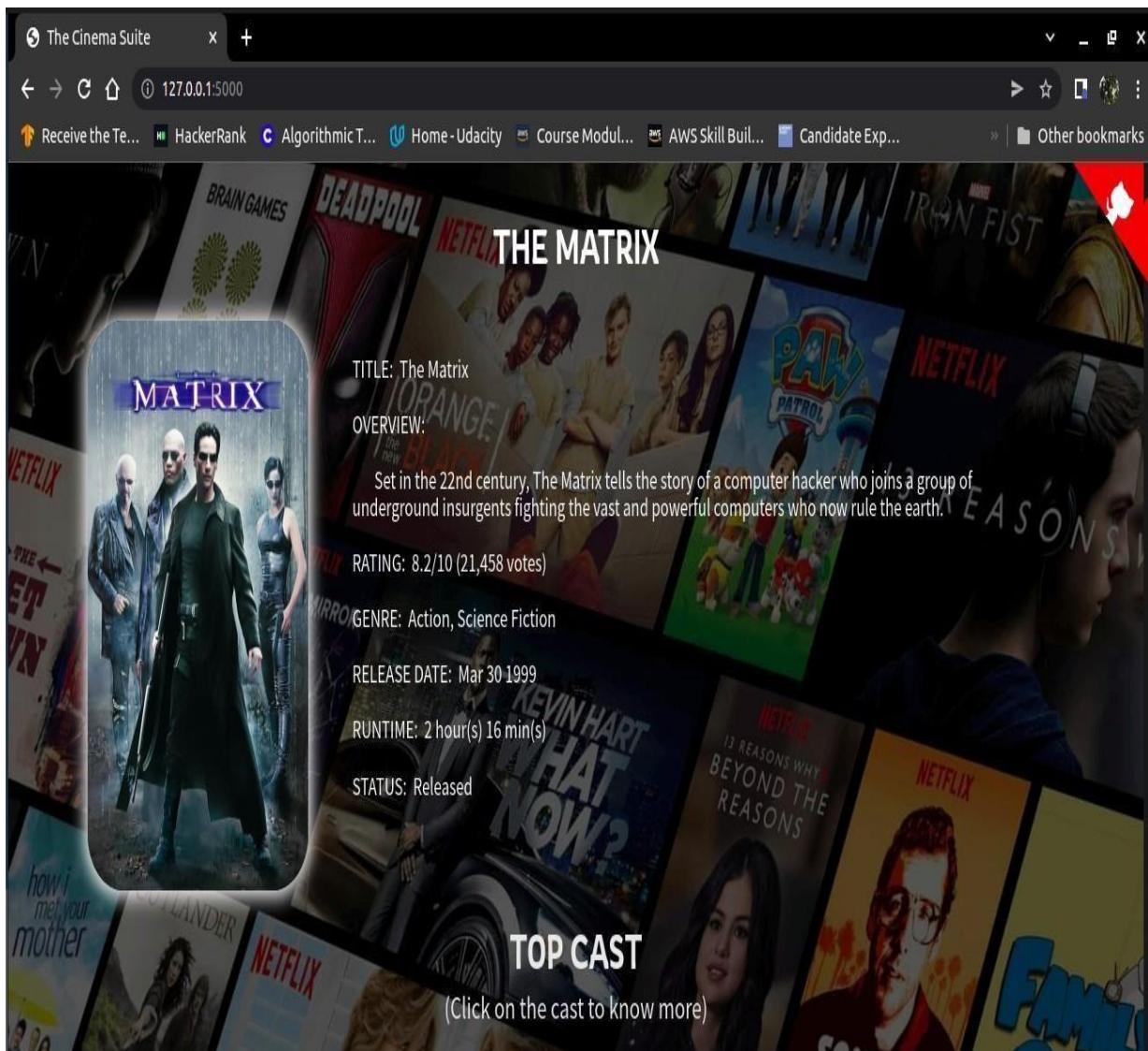


Figure 4.3: About the movie

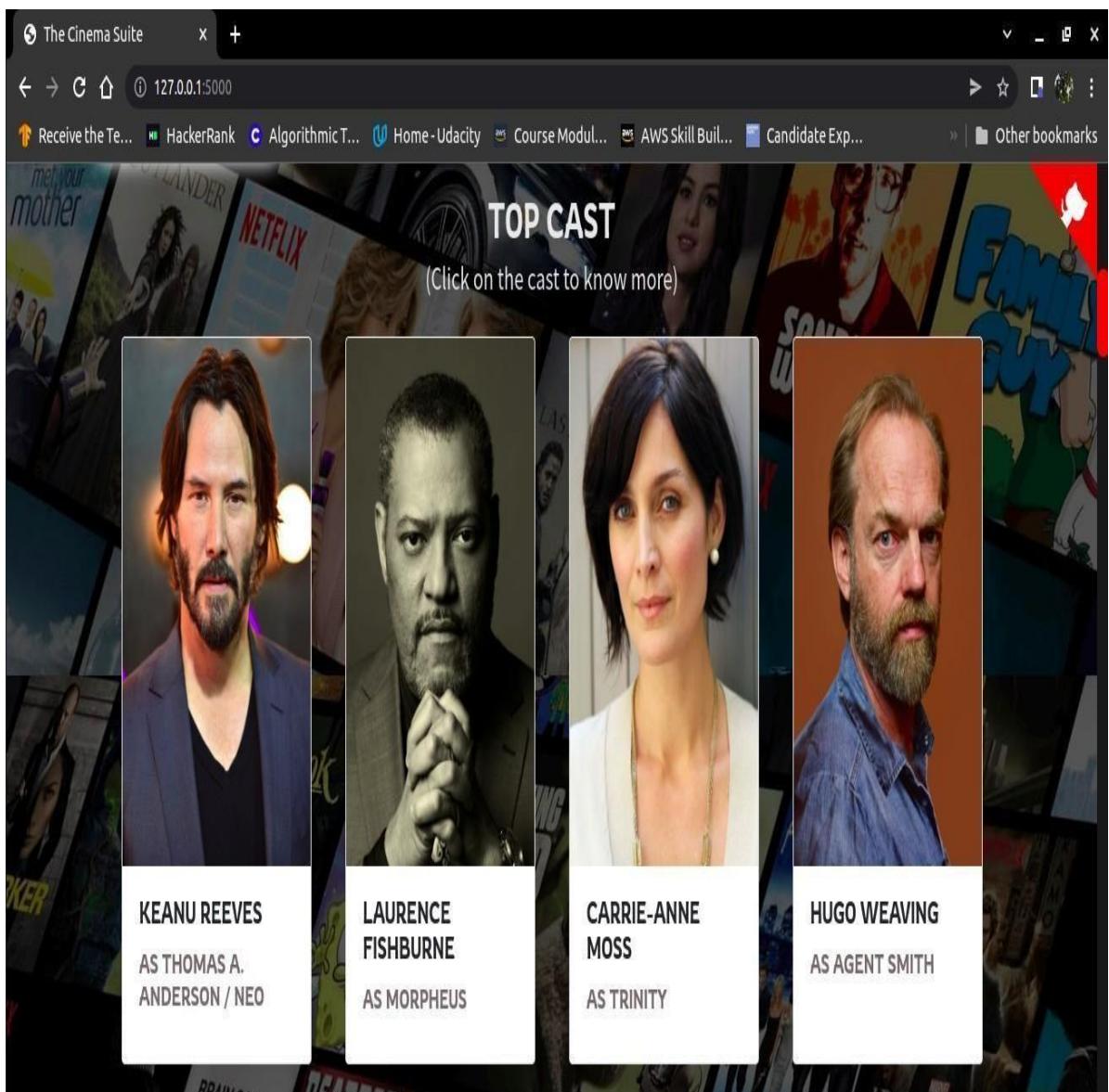


Figure 4.4: Cast of the movie

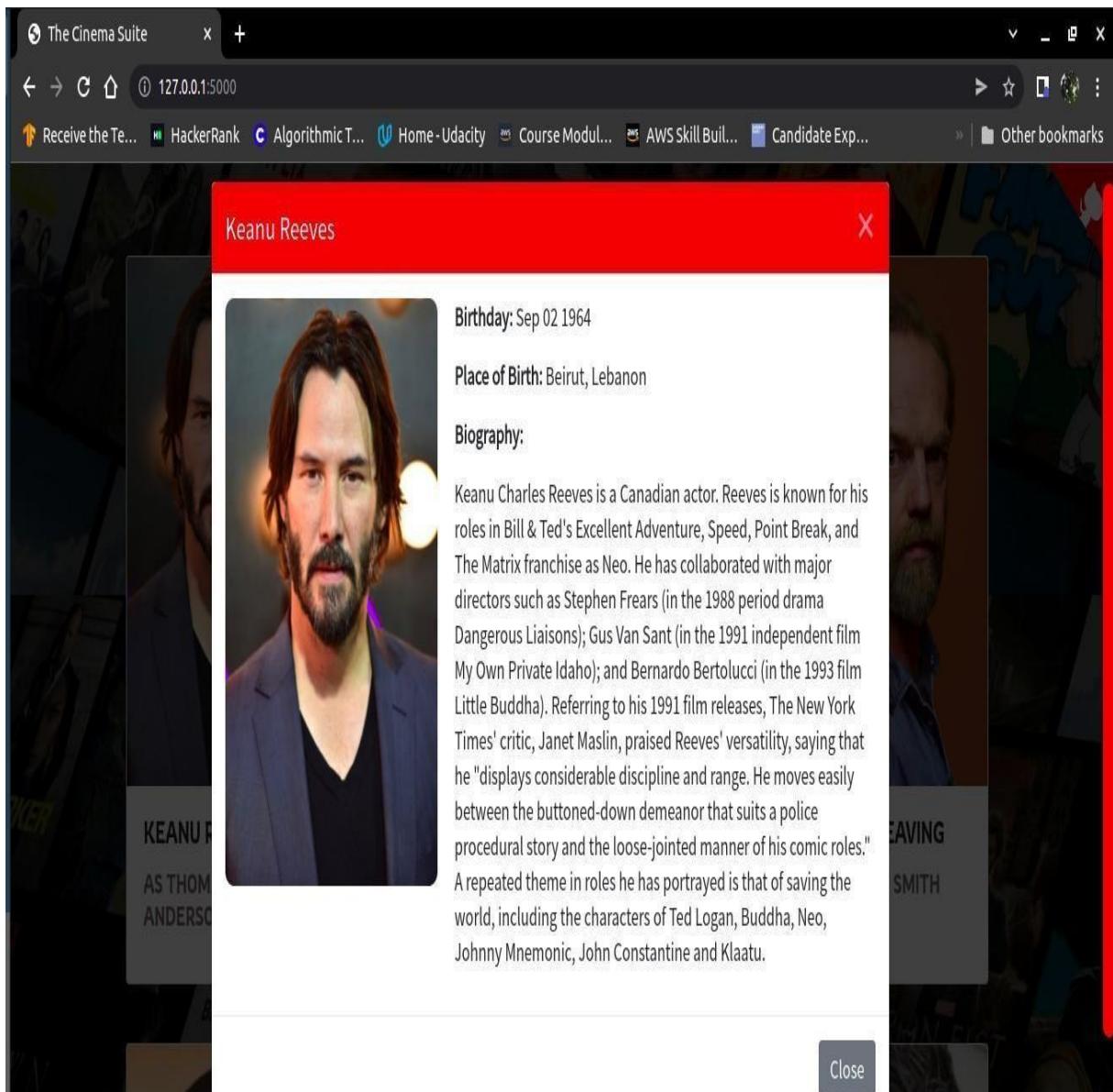


Figure 4.5: Information of the leading actors

The Cinema Suite

127.0.0.1:5000

Receive the Te... HackerRank Algorithmic T... Home - Udacity Course Modular... AWS Skill Buil... Candidate Exp... Other bookmarks

Comments	Sentiments
When this came out, I was living with a roommate. He went out and saw it, came home and said, "Dude, you have to go see The Matrix." So we left and he sat through it a second time. This movie is splendidly done. The mystery about what the Matrix is, unravels and you see a dystopian future unlike any we as a race would want. I have watched this over and over and never tire of it. Everyone does a great job acting in this, the special effects are above par and the story is engaging.	Positive : 😊
This film doesn't age, it will be contemporary even in 2030 or 2040. Wachowski's best one, by far.	Positive : 😊
The film is as well crafted as the matrix itself! On another level entirely to any other science fiction film from the last 20 years . Getting lost in another world, is interly what Cinema is made for. This one takes you into a whole new universe interly.	Positive : 😊
and this is all. because each explanation sounds wrong. sure, the acting,, the plot, the fight/action scenes are great. maybe unique. but the essence, for me, remains the basic idea . a fake reality against the pure truth. and the magic as clothes of each character. because it is more than a film. it is a revolution in the way to see a film and to discover reality. you are Neo. and the mythological mix of symbols and cultural references and the simple story who seems have more and more levels are more than fascinating - it is real. sure, Matrix has a lot of reviews and around it is very easy to say hypothesis, verdicts, opinions or, maybe, critics. it is enough to say it is a masterpiece. or the perfect fascinating fairy tale.	Positive : 😊
This is one of those movies that "everyone" has seen, except me. I tried to watch it with some friends about fifteen years ago, but they had two kids under five years old and I was only able to ingest it in five minute increments. So a friend lent me the video and I finally got a chance to see it. It's a pretty	Positive : 😊

Figure 4.6: Reviews

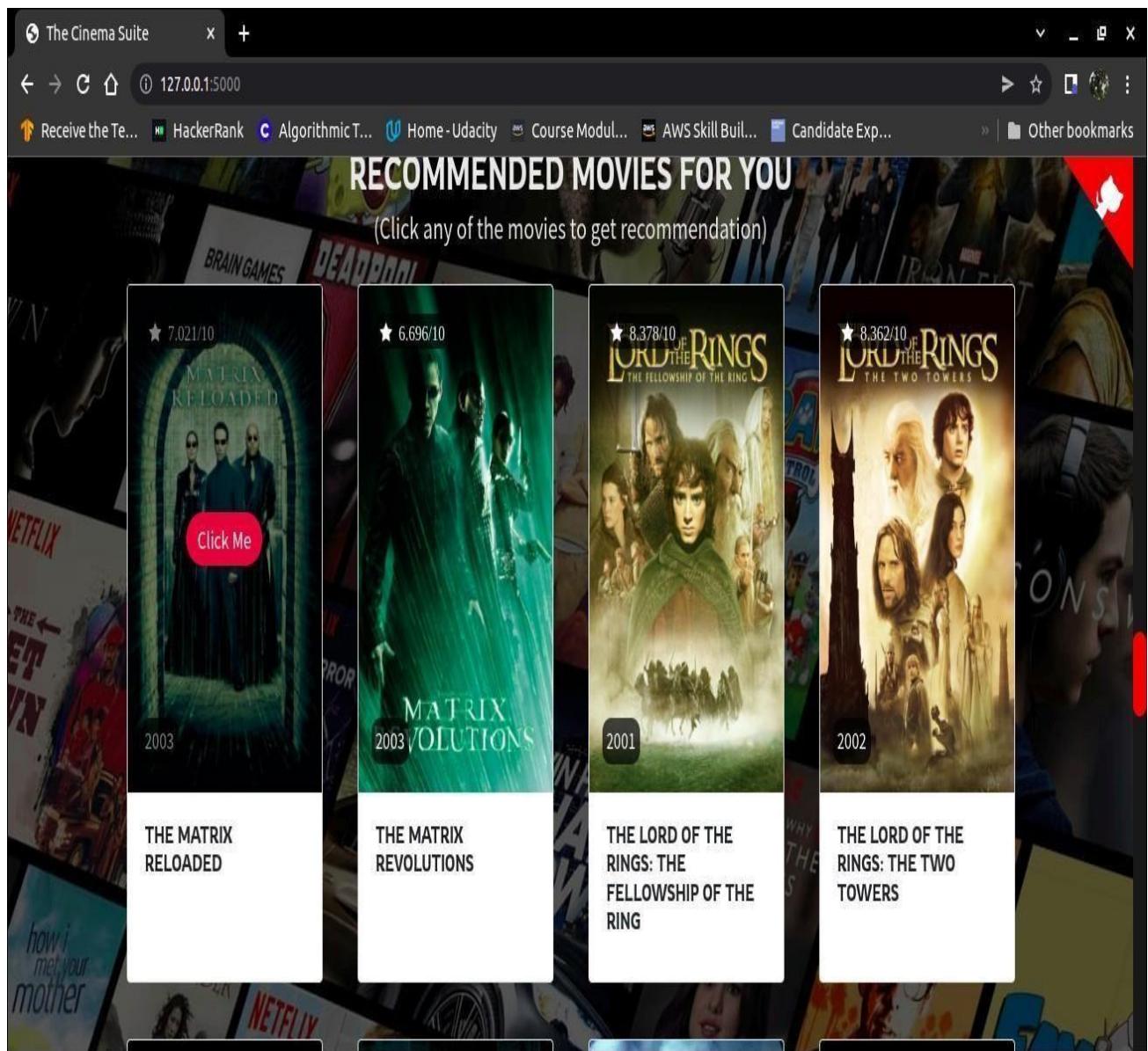


Figure 4.7: Recommendation of movies

The screenshot shows the AWS Elastic Beanstalk console interface. The left sidebar lists environments, applications, and change history. Under the 'movrec' application, the 'Movrec-env' environment is selected. The main dashboard for 'Movrec-env' displays the following information:

- Health:** Shows a green circle with a checkmark and the status "Ok".
- Running version:** movrec-source, with a "Upload and deploy" button.
- Platform:** Python 3.8 running on 64bit Amazon Linux 2/3.3.12, with a "Change" button.
- Recent events:** A table with columns Time, Type, and Details, showing one entry: 2022-04-20 15:40:46.

At the bottom, there are links for Feedback, Privacy, Terms, and Cookie preferences.

Figure 4.8: Deployment of movie recommendation on Elastic Beanstalk

The screenshot shows the AWS Elastic Beanstalk Environments page. At the top, there is a blue banner with the text "AWS Graviton now supported" and a message about its performance benefits. Below the banner, the left sidebar has links for "Environments", "Applications", and "Change history". Under "Recent environments", there is a single entry for "Movrec-env". The main content area is titled "All environments" and contains a table with one row. The table columns are: Environment name, Health, Application name, Date created, Last modified, URL, and Running version. The data for the single environment is as follows:

Environment name	Health	Application name	Date created	Last modified	URL	Running version
Movrec-env	Ok	movrec	2022-04-20 01:34:15 UTC+0530	2022-04-20 15:15:03 UTC+0530	Movrec-env.eba-mp8mpuq7.us-east-1.elasticbeanstalk.com	movrec source

At the bottom of the page, there are links for "Feedback", "© 2022, Amazon Internet Services Private Ltd. or its affiliates.", "Privacy", "Terms", and "Cookie preferences".

Figure 4.9:All environments

The screenshot shows the AWS Elastic Beanstalk console interface. At the top, there's a navigation bar with tabs for 'Environments' and 'The Cinema Suite'. The URL in the address bar is [us-east-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-1#/application/overview?applicationName=movrec](https://us-east-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-1#/application/overview?applicationName=movrec). Below the address bar, there are several bookmarks: 'Receive the Te...', 'HackerRank', 'Algorithmic T...', 'Home - Udacity', 'Course Modul...', 'AWS Skill Buil...', 'Candidate Exp...', 'Other bookmarks', and a user profile 'zulfqar100'.

The main content area is titled 'Elastic Beanstalk' and shows a message about AWS Graviton now supported. It includes a breadcrumb trail: 'Elastic Beanstalk > Applications > movrec'. On the right, there's a 'Actions' dropdown menu. The central part of the screen is titled 'Application 'movrec' environments' and features a search bar with the placeholder 'Filter results matching the display values'. A table lists the environments for the 'movrec' application:

Environment name	Health	Date created	Last modified	URL	Running versions	Platform	Pla sta
Movrec-env	Ok	2022-04-20 01:34:15 UTC+0530	2022-04-20 15:15:03 UTC+0530	Movrec-env.eba-mp8mpuq7.us-east-1.elasticbeanstalk.com	movrec-source	Python 3.8 running on 64bit Amazon Linux 2	Su

At the bottom of the page, there are links for 'Feedback', '© 2022, Amazon Internet Services Private Ltd. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

Figure 4.10: Application of movrec environment

# **Chapter 5**

## **Conclusion**

This paper is broken into two primary sections. One focuses on a movie recommendation system, while the other is concerned with sentiment analysis. The research examines both systems in depth and draws some crucial conclusions. The Cosine Similarity algorithm has been used in the Movie Recommendation System to recommend the best movies that are relevant to the movie submitted by the user based on multiple characteristics such as the movie's genre, overview, cast, and ratings. Even after multiple testing, Cosine Similarity has shown reasonable results and has been pretty accurate in selecting movies. In this study, sentiment analysis is also significant. Its primary goal is to categorize evaluations as good or negative. While conducting sentimental analysis, one difficulty is the linguistic barrier. Only reviews written in English may currently be examined. If the reviews are sarcastic or humorous, the Sentimental analysis incorrectly classifies them.

## **5.1 Future Scope**

Collaborative filtering, Content-based filtering, Hybrid recommender systems, and Personality-based recommender systems are the four types of recommender systems. Each filtering algorithm is utilised in accordance with the application or product's specific requirements. However, one thing that each of them needs to concentrate on is breaking free from the 'Filter Bubble' in order to provide a fresh viewpoint to the idea of assisting users in establishing, discovering, and understanding their individual preferences. Integrating self-actualization to do justice to serendipity while suggesting will also support rather than replace human decision-making by knowing preferences in the future of recommender systems.

# **Chapter 6**

## **References**

- [1] International Journal of Computer Applications (0975 – 8887) Volume124 – No.3, August 2015
- [2] Nymphia Pereira et al, / (IJCSIT) International Journal of ComputerScience and Information Technologies, Vol. 7 (1) , 2016, 281-284
- [3] Linnaeus University, School of Computer Science, Physics andMathematics, Kalmar, Sweden
- [4] International Journal Of Scientific Technology Research Volume 8, Issue 12, December 2019
- [5] Human-centric Computing and Information Sciences volume 8,Article number: 38 (2018)
- [6] 2020 International Conference on Emerging Trends in InformationTechnology and Engineering (ic-ETITE)
- [7] Prateek Sappada, YashSadhani, PranitArora. Movie Recommender System Search Engine Architecture, Spring 2017, NYU Couran.
- [8] Nagamanjula R, A. Pethalakshmi A Novel Scheme for Movie Recommendation System using User Similarity and Opinion Mining, International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-8 Issue-4S2 March, 2019

- [9] Nupur Kalra, Deepak Yadav, Gaurav Bathla. Movie Recommender System using Collaborative Filtering, International Journal on Future Revolution in Computer Science Communication Engineering, ISSN: 2454-4248 Volume: 4 Issue: 12, December 2018.
- [10] Marwa Hussien Mohamed, Mohamed Helmy Khafagy Recommender Systems Challenges and Solutions Survey, " 2019 International Conference on Innovative Trends in Computer Engineering (ITCE'2019), Aswan, Egypt, 2-4 February 2019.
- [11] Rajatha Prabhu, Pooja Shetty, Shilpa, Shwetha D R, Ramakrishna Hegde, "A review: Recommender System using Collaborative Filtering and Gray Sheep Problem," ©IJEDR 2018 — Volume 6, Issue 2 — ISSN: 2321-9939.
- [12] Dr. Sarika Jain, Anjali Grover, Praveen Singh Thakur, Sourabh Kumar Choudhary. Trends, Problems and Solutions of Recommender System, International Conference on Computing, Communication and Automation (ICCCA2015).
- [13] Goral Godhani, Maulik Dhamecha. A Study on Movie Recommendation System Using Parallel MapReduce Technology, © 2017 IJEDR — Volume 5, Issue 1 — ISSN: 23219939.
- [14] Arpan V Dev, Anuraj Mohan. Recommendation System For Big Data Applications Based On Set Similarity Of User Preferences, 2016 International Conference on Next Generation Intelligent Systems (ICNGIS), 2016.
- [15] Shreya Agrawal, Pooja Jain, "An Improved Approach for Movie Recommendation System," International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), (I-SMAC 2017).
- [16] N Pavitha , Vithika Pungliya , Ankur Raut , Roshita Bhonsle ,Atharva Purohit , Aayushi Patel , R Shashidhar Movie Recommendation and Senti-ment Analysis Using Machine Learning, Global Transitions Proceedings(2022),doi<https://doi.org/10.1016/j.glt.2022.03.012>

## **Publications:**

International Journal For Research in Engineering Applications & Management (IJREAM)

DOI: 10.35291/2454-9150.2022.0130

# **Chapter 7**

## **Bibliography**

<https://www.analyticsvidhya.com/blog/2020/11/create-your-own-movie-movie-recommendation-system/>  
<https://www.sciencedirect.com/science/article/pii/S2666285X22000176>  
<https://data-flair.training/blogs/data-science-r-movie-recommendation/>  
<https://towardsdatascience.com/how-to-build-a-movie-recommendation-system-67e321339109>  
<https://ijcsit.com/docs/Volume>  
file:///C:/Users/Admin/Downloads/ResearchPaperRecommenderSystems–ALiteratureSurvey–preprint.pdf  
file:///C:/Users/Admin/Downloads/AMovieRecommenderSystemMOVREC.pdf  
<http://www.ijstr.org/final-print/dec2019/A-Review-Paper-On-Collaborative-Filtering-Based-Moive-Recommened-System-.pdf>  
file:///C:/Users/Admin/Downloads/Moviesent.pdf  
<https://ieeexplore.ieee.org/document/9077803>

## The Editorial Board of

INTERNATIONAL JOURNAL FOR RESEARCH IN ENGINEERING APPLICATION & MANAGEMENT  
ISO 3297:2007 CERTIFIED JOURNAL | ISSN : 2454-9150

## Certificate of Publication

This is to Certify that Paper ID : **IJREAMV08I0185023**

# *Movie Recommendation System Using Rest API And Sentiment Analysis*

## Authored By

# Mirza Zulfiqar Ali Jaffer Ali

has been published in Volume 08, Issue 01, Apr 2022.

The mentioned paper has gone through Peer Review Process & measured upto the required standard.

Editor, IJREAM



*Certificate No. 245491502200289*

*\*Corresponding Author*

DOI : 10.35291/2454-9150.2022.0130



IJREAM  Crossref  
International Journal for Research  
in Engineering Application & Management  
<http://www.ijream.org>

The Editorial Board of

INTERNATIONAL JOURNAL FOR RESEARCH IN ENGINEERING APPLICATION & MANAGEMENT  
ISO 3297:2007 CERTIFIED JOURNAL | ISSN : 2454-9150

# Certificate of Publication

This is to Certify that Paper ID : IJREAMV08I0185023

*Movie Recommendation System Using  
Rest API And Sentiment Analysis*

Authored By

*Jain Vandana Vinit*

has been published in **Volume 08, Issue 01, Apr 2022.**

The mentioned paper has gone through Peer Review Process & measured upto the required standard.



Editor, IJREAM



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INTERNATIONAL CENTRE



Google  
scholar



THOMSON REUTERS



\*Corresponding Author

DOI : 10.35291/2454-9150.2022.0130



IJREAM

ADVANCED SCIENCES INDEX

Crossref

International Journal for Research  
in Engineering Application & Management

<http://www.ijream.org>

Certificate No. 245491502200290



The Editorial Board of

INTERNATIONAL JOURNAL FOR RESEARCH IN ENGINEERING APPLICATION & MANAGEMENT  
ISO 3297:2007 CERTIFIED JOURNAL | ISSN : 2454-9150

# Certificate of Publication

This is to Certify that Paper ID : IJREAMV08I0185023

*Movie Recommendation System Using  
Rest API And Sentiment Analysis*

Authored By

*Jain Jeel Jitendra*

has been published in **Volume 08, Issue 01, Apr 2022.**

The mentioned paper has gone through Peer Review Process & measured upto the required standard.



Editor, IJREAM



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INTERNATIONAL CENTRE



Google  
scholar



THOMSON REUTERS



7.427



आग्ने विद्यालय  
UGC  
University Grants Commission

Certificate No. 245491502200291



IJREAM

ADVANCED SCIENCES INDEX

International Journal for Research

Crossref

in Engineering Application & Management

<http://www.ijream.org>

\*Corresponding Author

DOI : 10.35291/2454-9150.2022.0130



QUALITY  
CERTIFICATION

ISO  
3297:2007

The Editorial Board of

INTERNATIONAL JOURNAL FOR RESEARCH IN ENGINEERING APPLICATION & MANAGEMENT  
ISO 3297:2007 CERTIFIED JOURNAL | ISSN : 2454-9150

# Certificate of Publication

This is to Certify that Paper ID : IJREAMV08I0185023

*Movie Recommendation System Using  
Rest API And Sentiment Analysis*

Authored By

*Rathod Yashi Sandeep*

has been published in **Volume 08, Issue 01, Apr 2022.**

The mentioned paper has gone through Peer Review Process & measured upto the required standard.



Editor, IJREAM



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INTERNATIONAL CENTRE



Google  
scholar



University Grants Commission



Certificate No. 245491502200292

\*Corresponding Author

DOI : 10.35291/2454-9150.2022.0130



IJREAM



International Journal for Research  
in Engineering Application & Management

<http://www.ijream.org>





Om Ganeshay Namah  
**A. P. SHAHI INSTITUTE OF TECHNOLOGY**  
(Approved by AICTE New Delhi & Govt. of Maharashtra, Affiliated to University of Mumbai)  
(Religious Jain Minority)

**Department Of Computer Engineering**

**Academic Year: 20<sup>21</sup> to 20<sup>22</sup>**

**Year (SE / TE / BE) :** BE      **SEM :** VIII

**Project Title:** MOVIE RECOMMENDATION SYSTEM

**Team Member 1: (Team Leader)**

Name: Mirza Zulfikar Ali Jaffer Ali

College ID: 18102043 Email: zulfikarmirza92@gmail.com

Mobile No: 9967031338

**Team Member 2:**

Name: Jeev Jitendra Jain

College ID: 18102041 Email: jeevjain09@gmail.com

Mobile No: 8108863863

**Team Member 3:**

Name: Vandan Vinit Jain

College ID: 18102031 Email: vandanjain78@gmail.com

Mobile No: 7506739413

**Team Member 4:**

Name: Yashvi Sandeep Rathod

College ID: 18102039 Email: yashvirathod66@gmail.com

Mobile No: 9167861440

*Future*

**Project Guide**  
Dr. Rakesh K. Ambekar  
Guide Name:

**Project Coordinator**

**Head of Department**



## Department Of Computer Engineering

Academic Year: 2021 to 2022

Year (SE / TE / BE ): BE SEM : VIII WEEK No.: 01

Date: From 17<sup>th</sup> Jan to 21<sup>st</sup> Jan

Progress Planned	Progress Achieved
Planned for the topic of the project and searched for datasets.	Topic decided r.e Movie recommendation system dataset:- IMDB dataset .

Guides Review: The Guide will be more about the topic. Identify objectives to be achieved

Signature

Team Member 1: Tu

Team Member 2: Jain

Team Member 3: GJ

Team Member 4: YK

Project guide: Dr. Rahul K. Ambekar

Signature:

Date: 22/04/22



## Department Of Computer Engineering

Academic Year: 2021 to 2022

Year (SE / TE / BE): BE SEM : VIII WEEK No.: 02

Date: From 24<sup>th</sup> Jan to 28<sup>th</sup> Jan

Progress Planned	Progress Achieved
To develop a machine learning model to recommend appropriate movies according to content based filtering	Developed the machine learning model with 92% accuracy and predicted/recommended set of 10 movies.

Guides Review: Analyse existing literature to determine the applicability of machine learning model.

Signature

Team Member 1: Jai

Team Member 2: Jain

Team Member 3: Jai

Team Member 4: Jai

Project guide: Dr. Rahul K. Ambekar

Signature: Fer

Date: 22/04/22



## Department Of Computer Engineering

Academic Year: 2021 to 2022

Year (SE / TE / BE) : BE SEM : VII WEEK No.: 03  
Date: From 31<sup>st</sup> Jan to 4<sup>th</sup> Feb

Progress Planned	Progress Achieved
To apply some hyper parameter tuning to the machine learning model to increase the accuracy.	i) Found a limitation in ML model. It would only recommend movies present in dataset. ii) Tried to find the sol'

Guides Review: Discussing the limitations of ML model, prepared note on API's

Signature

Team Member 1: Zu... -

Team Member 2: Jain

Team Member 3: DJ

Team Member 4: YB

Project guide: Dr. Rahul K. Andher

Signature:

Date: 22/04/22



## Department Of Computer Engineering

Academic Year: 2021 to 2022

Year (SE / TE / BE ): BE SEM : VIII WEEK No.: 04

Date: From 7<sup>th</sup> feb to 11<sup>th</sup> feb

Progress Planned	Progress Achieved
We discovered a limitation so to solve that we found a way:- i) To use Rest API ii) Rest API - TMDB	The solution for limitation was achieved. We explored the TMDB API & fired some base API calls using postman.

Guides Review: Start working on Backend. Demonstrate prototype working in next meeting.

Signature

Team Member 1: Shiv

Team Member 2: Jain

Team Member 3: Dhaval

Team Member 4: YK

Project guide:

Dr. Rahul K. Ambekar

Signature:

Rahul K. Ambekar

Date:

22/04/22



## Department Of Computer Engineering

Academic Year: 20<sup>21</sup> to 20<sup>22</sup>

Year (SE / TE / BE ): BE SEM : VIII WEEK No.: 05

Date: From 21<sup>st</sup> feb to 25<sup>th</sup> feb

Progress Planned	Progress Achieved
i) To complete the base of the project & to get all the data from the API including poster & reviews.	i) The base of the project was completed.
ii) To start developing backend scripts using flask.	ii) Frontend & backend developed.
iii) Start developing frontend.	iii) Research paper drafted

Guides Review: Research paper progress to be shown in next meeting

Signature

Team Member 1: Rahul

Team Member 2: Jain

Team Member 3: Dinesh

Team Member 4: Yash

Project guide:

Dr. Rahul K. Ambani

Signature:

for

Date: 22/04/22



## Department Of Computer Engineering

Academic Year: 20<sup>21</sup> to 20<sup>22</sup>

Year (SE / TE / BE) : BE SEM : VII WEEK No.: 06

Date: From 4<sup>th</sup> April to 8<sup>th</sup> April

Progress Planned	Progress Achieved
To apply sentiment analysis on user's review & to apply a chain reaction on the recommended movies.	i) The project was successfully executed. ii) The app <sup>n</sup> was successfully deployed on AWS cloud.
To get the recommended movies for the clicked ones. To add cast details.	iii) The research paper is accepted & published @ ijreset, ijrcm, ijream.
To deploy whole app <sup>n</sup> on AWS.	

Guides Review: Prepare Power Point presentation and demostrate morning application in the next meeting

Signature

Team Member 1: Ru

Team Member 2: Fatin

Team Member 3: Dj

Team Member 4: Yh

Project guide: Dr. Rahul K. Ambekar

Signature:

Date: 22/04/22