G H Patel College of Engineering and Technology

(A Constituent College of CVM University)

DEPARTMENT OF COMPUTER ENGINEERING

Report on

" BOOK RECOMMENDATION SYSTEM "

Under subject of -

Mini Project

(102040601)

B.E semester – VI
Academic year 2022-23

Submitted by –

Ashish Patel (12002040501004)

Dharmik Parmar (12002040501013)

Guided by -
Dr. Maulika Patel

CVM UNIVERSITY

GCET
G H PATEL COLLEGE OF ENGINEERING & TECHNOLOGY

CERTIFICATE

This is to certify that the Mini Project Report submitted entitled **“BOOK RECOMMENDATION SYSTEM”** has been carried out by **Ashish Patel** (12002040501004) under guidance in partial fulfillment for the Degree of Bachelor of Engineering in Computer Engineering, 6th Semester of G.H. Patel College of Engineering and Technology, CVM University, Vallabh Vidyanagar during the academic year2022-23.

Dr. Maulika Patel	Dr. Maulika Patel
Internal Guide	Head of Department

CVM UNIVERSITY

GCET
G H PATEL COLLEGE OF
ENGINEERING & TECHNOLOGY

CERTIFICATE

This is to certify that the Mini Project Report submitted entitled **“BOOK RECOMMENDATION SYSTEM”** has been carried out by **Dharmik Parmar** (12002040501013) under guidance in partial fulfillment for the Degree of Bachelor of Engineering in Computer Engineering, 6th Semester of G.H. Patel College of Engineering and Technology, CVM University, Vallabh Vidyanagar during the academic year 2022-23.

Dr. Maulika Patel	Dr. Maulika Patel
Internal Guide	Head of Department

Acknowledgement

I would like to express my sincere gratitude to all those who have contributed to the successfulcompletion of my college project.

First and foremost, I would like to thank my project supervisor , Dr.Maulika Patel for their guidance, support, and valuable feedback throughout the project. Their expertise and insights have been invaluable in shaping the direction and focus of my research.

I would also like to extend my thanks to the faculty members including lab-teacher, Dr. MaulikaPatel, who have provided me with their valuable time, insights, and resources. Their willingness to share their expertise and knowledge has been instrumental in shaping my understanding of the subject matter.

I would like to express my appreciation to my friends and family who have provided me with the emotional support and encouragement I needed to complete this project. Their unwaveringsupport has been a source of motivation and inspiration for me.

Finally, I would like to acknowledge the participants who have taken part in my study. Their time, effort, and cooperation have been vital in collecting the necessary data and information needed for this project.

Once again, thank you to everyone who has contributed to the successful completion of my college project.

Abstract

There are millions of books in the entire world wide and people need some instructionsto find the appropriate book. Making decision and scrolling for the right book from millionsof books can be hard and a complete waste of time. To compete and keep up in the market, some viewers may just rate the book in unbiased form. The use of such overstated descriptions is misleading the viewers to unpromising products or books. Resulting in inaccuracy of recommendation of books to the user. In this project, this problem domain has been addressed and has been tried to solve with a better approachusing a collaborative filtering method. This project offers a quick and intuitive book recommendation system that enables readers to navigate and find the appropriate book to read next. The end goal of this project is to use collaborative filtering method to providea better recommendation.

.

Index

Figures

1. Introduction

1. Introduction : Book Recommendation System

A “book recommendation system” is a type of recommendation system where we have to recommend similar type of books to the reader based on his interest. The books recommendation system is used by online websites which provide ebooks like google playbooks, open library, good Read’s, etc.

1.2 Aims and Objectives:

Aims:

- To develop a system to recommend books to the user.
- Research on background study of the field will be conducted.
- Research on the similar systems that intend to solve the same problem will beconducted.
- To research on how to make an effective recommendation.
- To gather the requirements and analyze them.

Objectives:

- Create a Book Recommendation System using collaborative filtering method.
- Get the required dataset for the system.
- Build a design/prototype.
- Evaluate and run tests on prototype to ensure the usability and functionality.
- To build an interactive interface allowing users to interact with the system.
- Documentation of the artefact.

2. Literature Review

Objective of the Chapter

This chapter gives summary on a detailed background of the field that is being covered on this project. It aims to provide a brief knowledge and understanding of the technical aspects of the project and the outcomes of the research that was conducted. Comparison,contrast and evaluation on this filed is conducted.

S. No.	Title of the Paper	Key Points	Conclusion	Reference
1.	A Case-Based Recommendation Approach for Market Basket Data.	CF; CB; AR (Association Rule); CBR (Case-Based Reasoning)	After compared the performance of developed RS conclude that CBR is the good method in case of transactions.	[1]
2.	Recommender Systems: An overviewof different approaches to recommendations	Recommendation System; Information Retrieval System; CF; CBF; Hybrid Filtering	The three approaches of recommendation system and their advantages and disadvantages.	[2]
3.	Recommendation analysis on Item-based and User-based Collaboration Filtering	IBCF; UBCF; Recommender System	IBCF and UBCF with implementation metrics, and conclude that IBCF provide better results than UBCF.	[3]
4.	Recommender Systems Handbook. Springer.	CF; CB; Multi-criteria recommender; Robust CF Neighborhood-based	In the unique approaches, hybrid robust filtering methods are better.	[4]
5.	Towards privacy in a context-aware social network based recommendation system	Content aware; social networking; privacy	Focus on protecting data and request for data, at the point of data collection.	[5]
6.	A study of hybrid recommendation algorithm based onuser.	Personalization; recommendation technology; collaborative filtering; hybrid algorithm	Hybrid algorithms are generates the results according to user’s rating and history record.	[6]
7.	Recommender systems in e-commerce	Electronic Commerce, cross-sell, up-sell, mass customization	The ideas of new applications in the field of recommendation systems in e-commerce sites.	[7]

3. Conventional Systems Used

There are several conventional systems used for book recommendation:

1. Content-based filtering: This system uses the attributes of the book to recommend similar books. Attributes such as author, genre, publisher, and publication date are considered.

2. Collaborative filtering: This system recommends books based on the user's past behavior and preferences. It uses the data of other users with similar preferences to recommend books.

3. Hybrid systems: This system combines both content-based and collaborative filtering to make more accurate recommendations. It uses a weighted approach to combine the results from both systems.

Motivation for Project

- A recommendation system helps an organization to create loyal customers. The recommendation system today are very powerful that they can handle the new customer too who has visited the site for the first time. They recommend the products which are currently trending or highly rated and they can also recommend the products which bring maximum profitto the company.

4. Devlopment

Objective of the chapter:

This chapter will give a full detail on the development of the artefact and describe the process of the development cycle along with proper justification of tools and techniques.The main objective of this chapter is to gather requirements, design and develop the system according to the gathered requirements.

Development Methodology

A software development methodology is a framework used to organize, plan and control the system development process. The development process is divided into various placesby a methodology. The use of a methodology ensures that tasks and time are properly managed and ensures product quality. Various methodologies have various processes and different phases, which can be advantageous according to the type of project. For developing this system, we will follow Rapid application development (RAD) methodology.

Rapid Application Development (RAD)

RAD is a sequence of software development techniques used to speed up the development of the software applications. To produce application RAD uses predefined prototyping techniques. A prototyping is a working model that functionally corresponds toa product component. RAD uses various tools and techniques to develop a predefined components, methods, structures and minimally coded applications models. In RAD functional modules are designed as prototypes in parallel and integrated so that the entire product can be delivered quickly. The development process can be easily customized as there is no detailed pre-planning in this methodology. These methodologies are specially used to demonstrate and collect requirements that help end users to envision the solution stacks.

RAD contains customizable processes, data and organizational integrated models (Tutorials point, 2019).

As, the waterfall software design approach often lead to the out of date efficient products by the time the software was released, leading to frustration RAD was born to tackle the problem (Rouse, 2019).

Reason for selecting Rapid Application Development (RAD)

- Any changes and progress can be measured.
- Development time is reduced.
- Increased reusability of the components of the system.
- Changing the requirements can be accommodated.
- Quick review on initial errors.
- With the use of powerful RAD tools iteration time can be short.

RAD model Vs Traditional SDLC

The traditional SDLC is based on a rigid model of processes that emphasizes needs analysis and collection before coding begins. While RAD support testing on each iteration, testing is carried out at the very end of the coding phase in traditional SDLC models. RAD model allows extensive user interfacing where developers can interact at every stage of the process with end users (Johnson, 2018). The traditional SDLC models, developers interact with the users during and after the requirements phase during the acceptance phase. After viewing the software, the user may need a few changes. In SDLC changingthe requirements in the product is quite rigid and may not be feasible to corporate. RAD model focuses on delivering the work models to the user in an iterative and progressive way. Therefore, this method reduces the risks of non-conformance as the requirements are satisfied by the user while delivering the products.

Stages involved in implementing RAD model

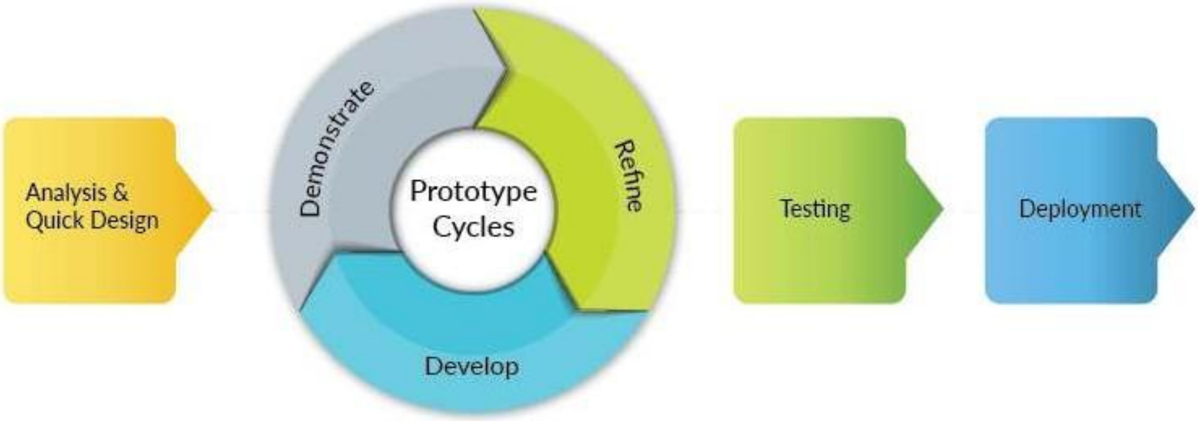


Figure4.1: Rapid Development Model (RAD)

(https://www.wavemaker.com/rapid-application-development-model/)

The following shows the basic stages involved in RAD model:

I. **Define the requirements:**

Here, the planning of the project is done, where developers gather requirement, limitation and specification. The basic structure of the project is determined, and the scope and solutions are identified in this phase (Kissflow, 2018).

II. **Prototype:**

In this phase, after gathering requirements of the system, initial prototyping and modeling is done. Prototype is created where user can view its features and function. Prototype are made without proper polish to, just show off specific features.

III. **Receive Feedback:**

In this stage, the feedbacks are received, what9s good and what9s not appropriate and what is not included. It isn9t only limited to functionality but also interface and visuals. After the receiving feedback, the process of prototyping continues. These two steps are repeated until a final product is declared. Testing is also carried along side.

IV. **Finalize Software:**

In this stage functions, models feature of the application are finalized with the user. Development of the application happens in this stage where stability, usability and maintainability are crucial before the deployment of the software (wave maker, 2013).

Development Tools and Techniques to be used

The following tools and techniques will be utilized for developing the system.

- **Python as the Core Programming language**

Python is a simple object-oriented and powerful language of programming allowing to focus on core functionality of the application. Due to availability of a wide variety of packages and libraries such as Pandas, NumPy, TensorFlow, etc. python is used extensively in the field of machine learning. It makes easy for developers to learn and implementing machine learning algorithms in their applications. It provides construct that enables clear programming on both small and large scales. Also, it is an open source programing language with an easy understandable syntax.

- **Flask as a Web Framework**

Flask is a lightweight web framework for Python. It is designed to be simple and easy to use, making it a popular choice for building web applications and APIs.

- **PyCharm**

PyCharm is a cross platform IDE (Integrated Development Environment) for python programmers by Jetbrains. It is used in Python and frameworks like Django for development. It includes all of the tools that a Python programmer needs to use, including an excellent text editor, syntax highlights, code finish, project browsing, database tools and web development projects.

- **HTML 5**

A standardized system for tagging text files to achieve content such as font, color, graphics and hyperlinks effects. HTML is a main mark-up language, which is used forstructure of the web pages in this system. HTML elements are represented by tags. HTML tags are used to render the content of the page, but browsers do not display the tags in this system.

- **CSS**

Cascading Style Sheets (CSS) is a style sheet language used to describe how HTMLelements are to be displayed on webpage on this system. CSS language is used for presentation of web pages, such as color, layout and fonts. It enables the display to be adapted to various devices types, such as large screens, small screens and printers.

- **Bootstrap**

Bootstrap is an easier and quicker web development, free front-end framework. Bootstrap contains HTML and CSS design templates and optional Java Script pluginsfor typography, shapes of form, buttons, navigation, tables, carousels or modals.

- **JAVA SCRIPT**

JavaScript (JS) is a dynamic computer programming language of HTML and the web. JavaScript enables interactive web pages, allow client to interact with the user, controlthe browser, communicate asynchronous and alter the content that is displayed. Thus,JS can be used on this system for interactive webpage.

System Requirements

A requirement is a feature that the system must have or a constraint that it must to be accepted by the client. Requirement Engineering aims at defining the requirements of the system under construction. Requirement Engineering include two main activities requirement elicitation which results in the specification of the system that the client understands and analysis which in analysis model that the developer canunambiguously interpret. A requirement is a statement about what the proposed system will do.

Requirements can be divided into two major categories:

- Functional Requirements.
- Non-Functional Requirements.

Functional Requirements:

A Functional Requirement is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputsto the software system, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional Requirements describe the interactions between the system and its environment independent of its application.

- Applying the algorithms on the train data
- Display the recommendations by the model.

Non-functional requirements:

Performance: The system should be able to handle large amounts of data and provide fast recommendations. The response time of the system should be minimal, even when dealing with high traffic volumes.

Reliability: The system should be highly reliable and available at all times. The system should be able to handle failures, recover quickly from errors, and provide redundancy to prevent data loss.

Scalability: The system should be able to scale up or down depending on the volume of data and the number of users. It should be able to handle an increasing number of users and books without compromising performance.

Security: The system should be secure, and user data should be protected from unauthorized access. The system should have mechanisms for user authentication, access control, and data encryption.

Usability: The system should be easy to use and provide a user-friendly interface. Users should be able to easily navigate through the system and make informed decisions based on the recommendations provided.

Maintainability: The system should be easy to maintain and update. The code should be well-structured and modular to allow for easy changes and updates. The system should also be documented to help with future maintenance.

Compatibility: The system should be compatible with a range of devices, operating systems, and web browsers to allow for broad access

5. Working of the System

A book recommendation system works by leveraging different types of data to generate personalized book suggestions for users. Here are the general steps involved in the working of a book recommendation system:

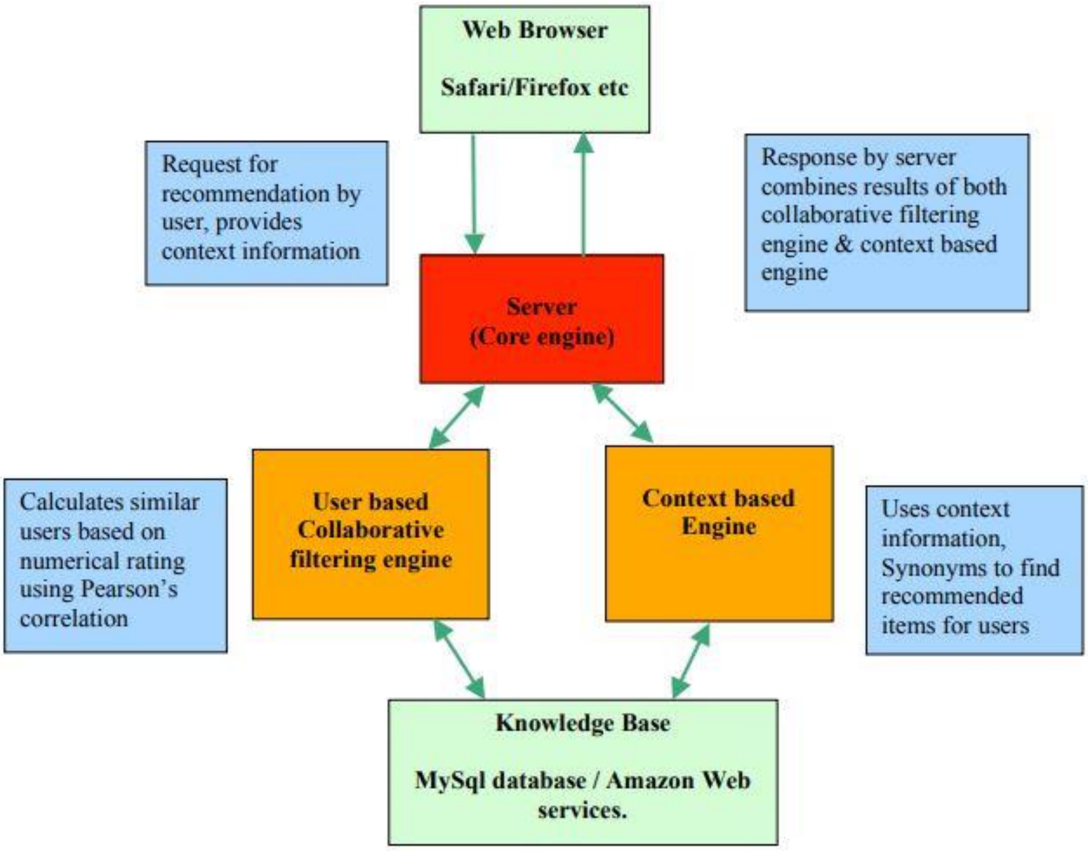

Web Browser
Safari/Firefox etc
Request for recommendation by user, provides context information
Response by server combines results of both collaborative filtering engine & context based engine
Server (Core engine)
Calculates similar users based on numerical rating using Pearson's correlation
User based Collaborative filtering engine
Context based Engine
Uses context information, Synonyms to find recommended items for users
Knowledge Base
MySql database / Amazon Web services.

Figure 5.1.1

Data Collection: The system collects data about books, authors, and users' reading history. The data can come from various sources, such as Goodreads, Amazon, and other book review websites.
Data Preprocessing: The collected data needs to be cleaned, transformed, and structured to ensure that it's accurate, consistent, and ready for analysis.

Feature Extraction: The system then extracts relevant features from the data, such as the book's genre, author, publication date, and user's reading history.

Data Analysis: The extracted features are analyzed to identify patterns, similarities, and correlations between books, authors, and users. This is typically done using machine learning algorithms.

User Profiling: The system creates a profile for each user based on their reading history, preferences, and behavior. This profile is used to generate personalized book recommendations for the user.

Recommendation Generation: The system generates book recommendations for the user based on their profile and the analysis of the collected data. The recommendations can be based on various factors, such as similarity to the user's reading history, popularity, ratings, and reviews.

Feedback Loop: The user's feedback on the recommendations is collected and used to improve the system's accuracy and relevance. The feedback can also be used to update the user's profile and generate new recommendations.

Overall, the book recommendation system uses advanced data analysis and machine learning techniques to provide users with personalized book recommendations based on their reading history and preferences.

Top 50 books are recommended to the users.

. The algorithm used for this method is very simple. It ranks the books based on their popularity score, which is calculated using the number of ratings and the average rating of each book. The formula used to calculate the popularity score is:

Popularity Score = (Rating * Votes) / (Rating + Votes)

Where,

Rating = the average rating of the book

Votes = the number of ratings the book has received

The books are then sorted in descending order of their popularity score, and the top 50 books are recommended to the users.

Collaborative filtering:

Collaborative filtering is a technique that uses the behavior and preferences of users to recommend items. In the context of a book recommender system, this means that the system would analyze the behavior of users, such as the books they have read, their ratings, and their reviews, to make recommendations to other users.

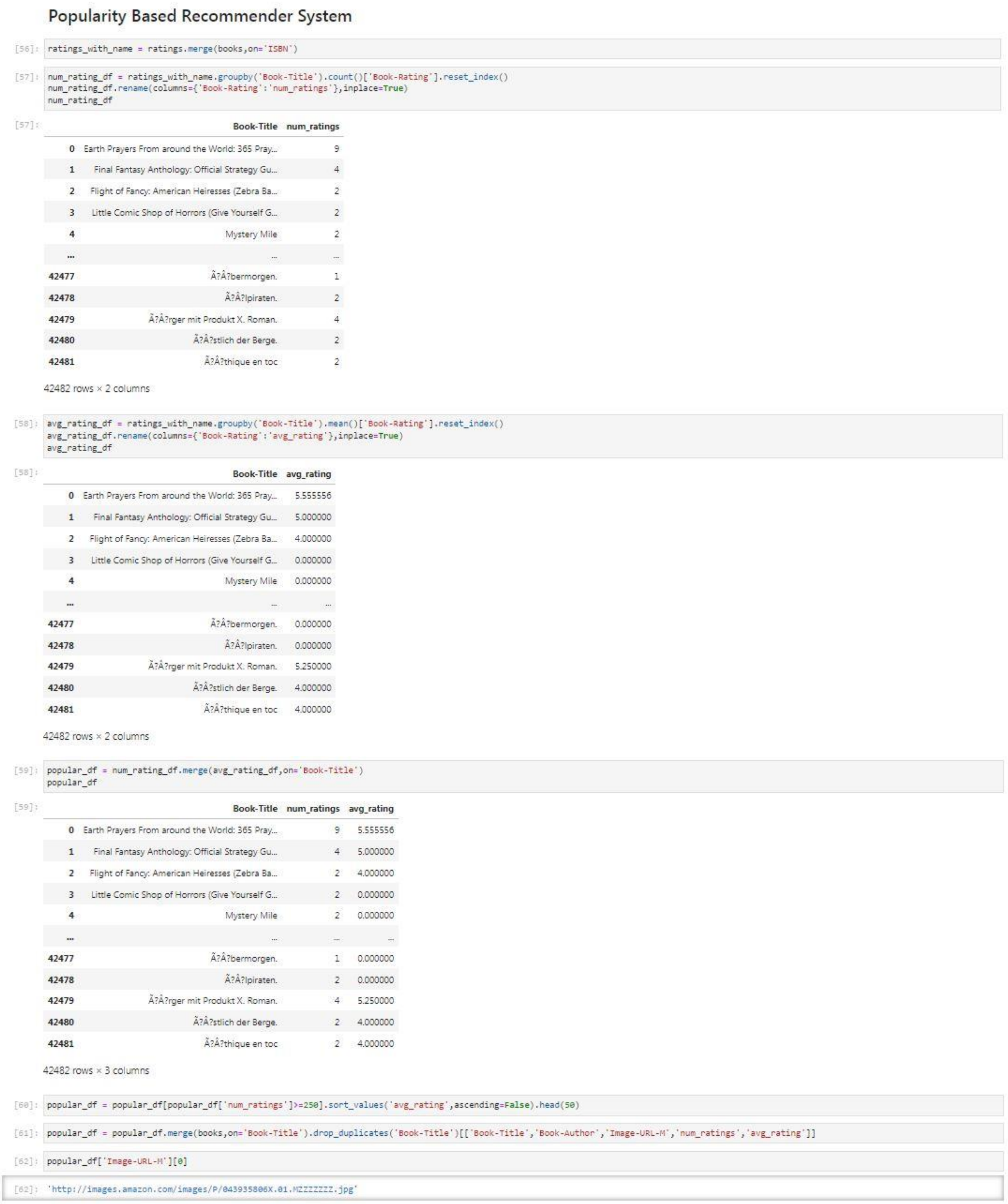
One approach to collaborative filtering is to use matrix factorization, which is a technique that decomposes the user-item interaction matrix into two lower-dimensional matrices. These matrices are then used to predict the missing ratings or recommendations. Another approach is to use user-based or item-based collaborative filtering, which uses similarities between users or items to make recommendations.

Overall, collaborative filtering can be a powerful technique for making book recommendations, especially in situations where there is a large dataset of user behavior and preferences. However, it also has some limitations, such as the cold-start problem, which occurs when there are not enough data points for new users or items. Therefore, it is important to carefully evaluate and choose the appropriate algorithm and approach for a book recommender system based on the specific needs and characteristics of the application.

6. Results and Conclusion

Result :

Popularity based :-



Popularity Based Recommender System

```
[56]: ratings_with_name = ratings.merge(books,on='ISBN')
```

```
[57]: num_rating_df = ratings_with_name.groupby('Book-Title').count()['Book-Rating'].reset_index()
      num_rating_df.rename(columns={'Book-Rating':'num_ratings'},inplace=True)
      num_rating_df
```

[57]:

	Book-Title	num_ratings
0	Earth Prayers From around the World: 365 Pray...	9
1	Final Fantasy Anthology: Official Strategy Gu...	4
2	Flight of Fancy: American Heiresses (Zebra Ba...	2
3	Little Comic Shop of Horrors (Give Yourself G...	2
4	Mystery Mile	2
...	...	...
42477	Ã?Â?bermorgen.	1
42478	Ã?Â?lpiraten.	2
42479	Ã?Â?rger mit Produkt X. Roman.	4
42480	Ã?Â?stlich der Berge.	2
42481	Ã?Â?thique en toc	2

42482 rows × 2 columns

```
[58]: avg_rating_df = ratings_with_name.groupby('Book-Title').mean()['Book-Rating'].reset_index()
      avg_rating_df.rename(columns={'Book-Rating':'avg_rating'},inplace=True)
      avg_rating_df
```

[58]:

	Book-Title	avg_rating
0	Earth Prayers From around the World: 365 Pray...	5.555556
1	Final Fantasy Anthology: Official Strategy Gu...	5.000000
2	Flight of Fancy: American Heiresses (Zebra Ba...	4.000000
3	Little Comic Shop of Horrors (Give Yourself G...	0.000000
4	Mystery Mile	0.000000
...	...	...
42477	Ã?Â?bermorgen.	0.000000
42478	Ã?Â?lpiraten.	0.000000
42479	Ã?Â?rger mit Produkt X. Roman.	5.250000
42480	Ã?Â?stlich der Berge.	4.000000
42481	Ã?Â?thique en toc	4.000000

42482 rows × 2 columns

```
[59]: popular_df = num_rating_df.merge(avg_rating_df,on='Book-Title')
      popular_df
```

[59]:

	Book-Title	num_ratings	avg_rating
0	Earth Prayers From around the World: 365 Pray...	9	5.555556
1	Final Fantasy Anthology: Official Strategy Gu...	4	5.000000
2	Flight of Fancy: American Heiresses (Zebra Ba...	2	4.000000
3	Little Comic Shop of Horrors (Give Yourself G...	2	0.000000
4	Mystery Mile	2	0.000000
...	...	...	...
42477	Ã?Â?bermorgen.	1	0.000000
42478	Ã?Â?lpiraten.	2	0.000000
42479	Ã?Â?rger mit Produkt X. Roman.	4	5.250000
42480	Ã?Â?stlich der Berge.	2	4.000000
42481	Ã?Â?thique en toc	2	4.000000

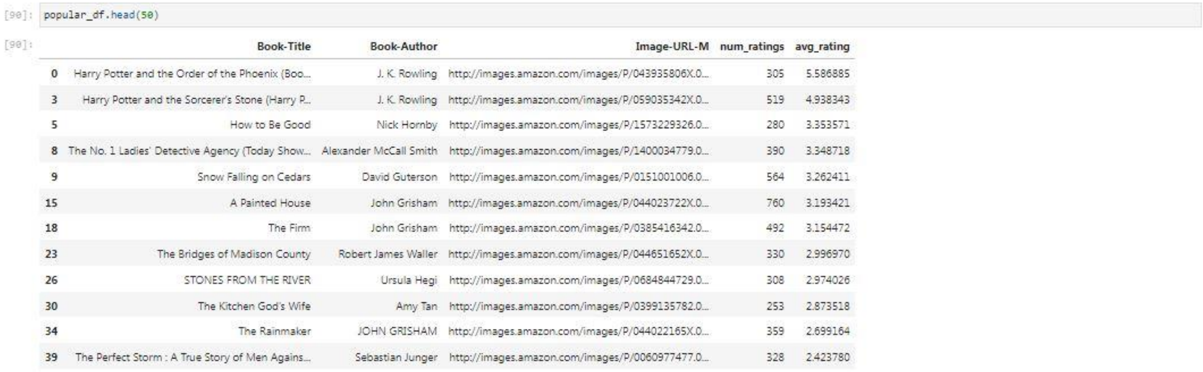
42482 rows × 3 columns

```
[60]: popular_df = popular_df[popular_df['num_ratings']>=250].sort_values('avg_rating',ascending=False).head(50)
```

```
[61]: popular_df = popular_df.merge(books,on='Book-Title').drop_duplicates('Book-Title')[['Book-Title','Book-Author','Image-URL-M','num_ratings','avg_rating']]
```

```
[62]: popular_df['Image-URL-M'][0]
```

```
[62]: 'http://images.amazon.com/images/P/043935806X.01.MZZZZZZZ.jpg'
```

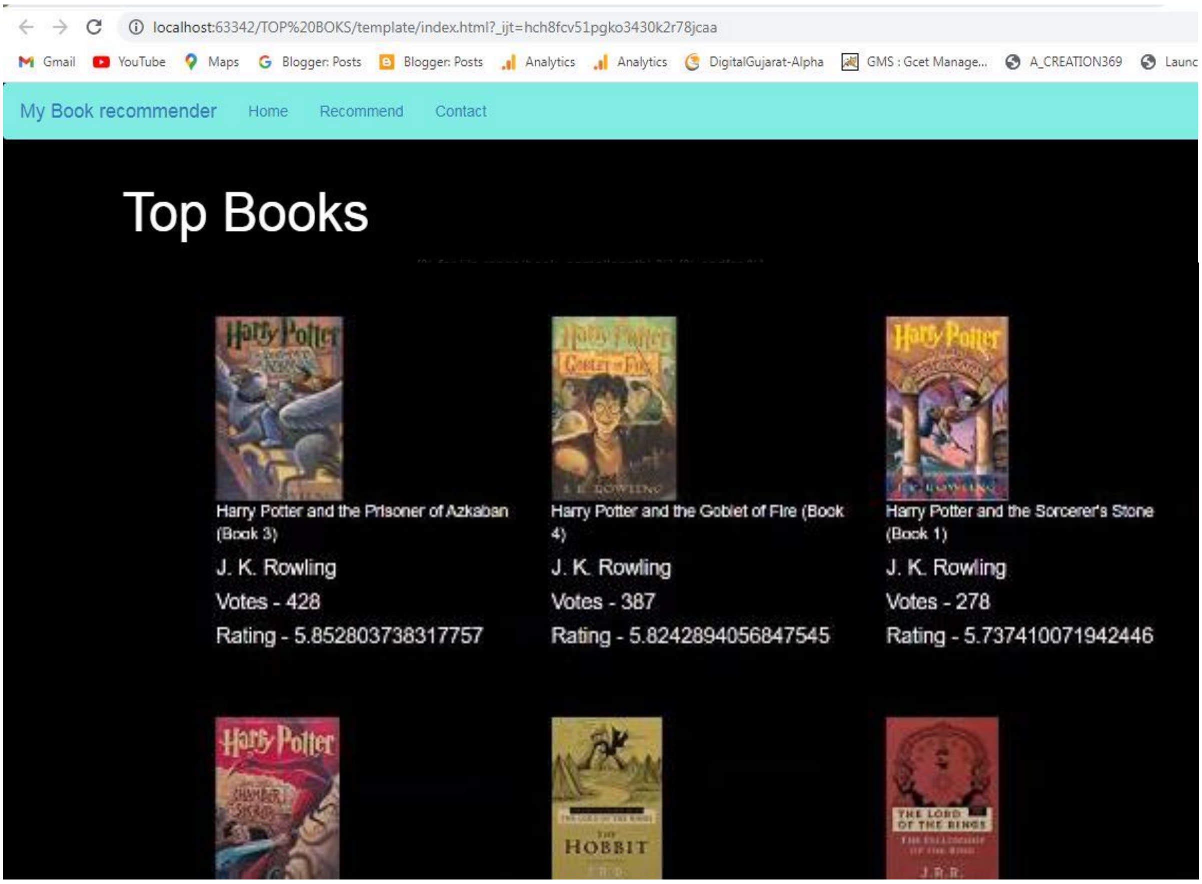


```
[90]: popular_df.head(50)
```

[90]:

	Book-Title	Book-Author	Image-URL-M	num_ratings	avg_rating
0	Harry Potter and the Order of the Phoenix (Boo...	J. K. Rowling	http://images.amazon.com/images/P/043935806X.0...	305	5.586885
3	Harry Potter and the Sorcerer's Stone (Harry P...	J. K. Rowling	http://images.amazon.com/images/P/059035342X.0...	519	4.938343
5	How to Be Good	Nick Hornby	http://images.amazon.com/images/P/1573229326.0...	280	3.353571
8	The No. 1 Ladies' Detective Agency (Today Show...	Alexander McCall Smith	http://images.amazon.com/images/P/1400034779.0...	390	3.348718
9	Snow Falling on Cedars	David Guterson	http://images.amazon.com/images/P/0151001006.0...	564	3.262411
15	A Painted House	John Grisham	http://images.amazon.com/images/P/044023722X.0...	760	3.193421
18	The Firm	John Grisham	http://images.amazon.com/images/P/0385416342.0...	492	3.154472
23	The Bridges of Madison County	Robert James Waller	http://images.amazon.com/images/P/044651652X.0...	330	2.996970
26	STONES FROM THE RIVER	Ursula Hegi	http://images.amazon.com/images/P/0684844729.0...	308	2.974026
30	The Kitchen God's Wife	Amy Tan	http://images.amazon.com/images/P/0399135782.0...	253	2.873518
34	The Rainmaker	JOHN GRISHAM	http://images.amazon.com/images/P/044022165X.0...	359	2.699164
39	The Perfect Storm : A True Story of Men Agains...	Sebastian Junger	http://images.amazon.com/images/P/0060977477.0...	328	2.423780

Module 1



Top Books

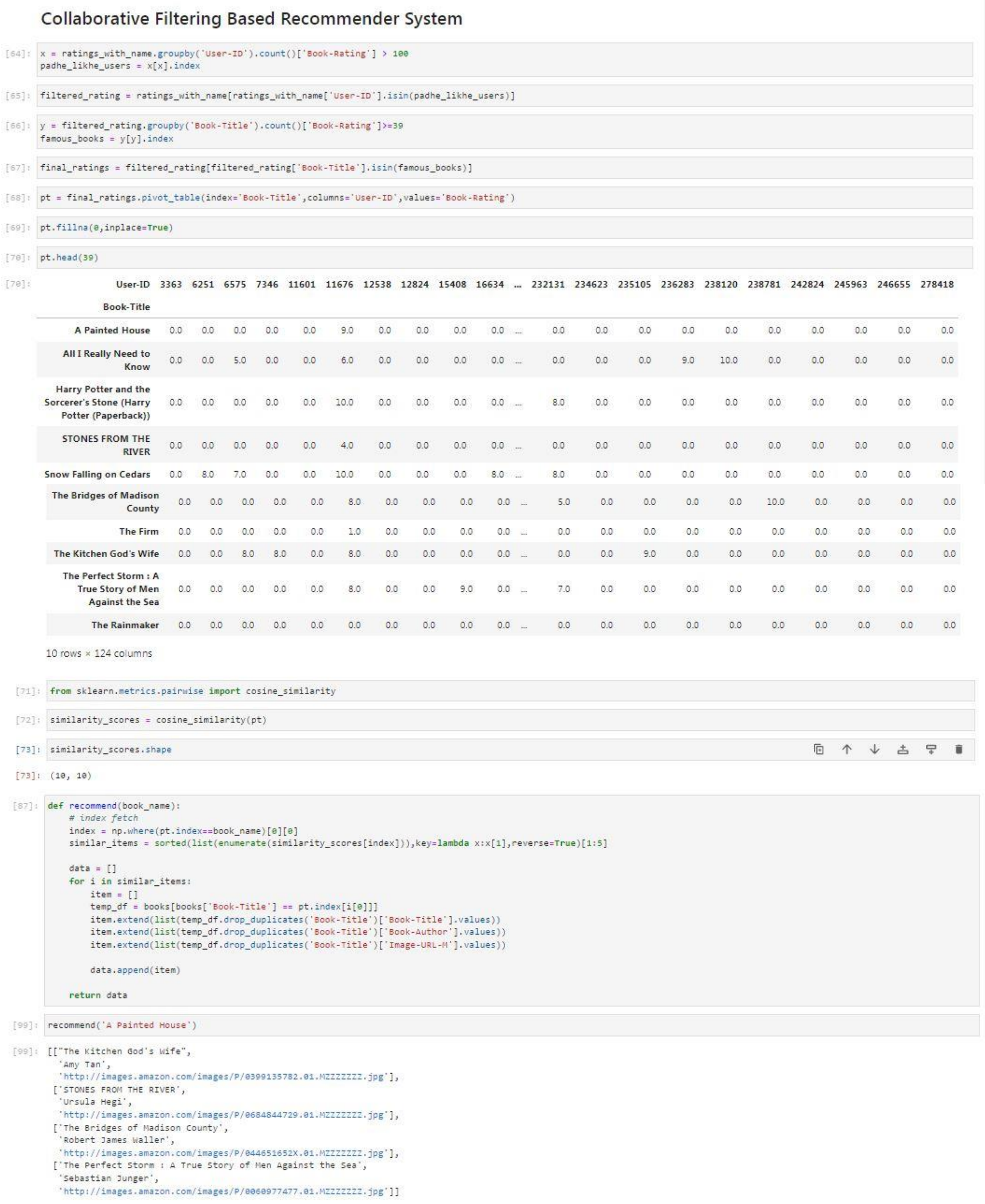
Harry Potter and the Prisoner of Azkaban (Book 3)
J. K. Rowling
Votes - 428
Rating - 5.852803738317757

Harry Potter and the Goblet of Fire (Book 4)
J. K. Rowling
Votes - 387
Rating - 5.8242894056847545

Harry Potter and the Sorcerer's Stone (Book 1)
J. K. Rowling
Votes - 278
Rating - 5.737410071942446

Figure 6.1.1

Collaborative Filtering-



Collaborative Filtering Based Recommender System

```
[64]: x = ratings_with_name.groupby('User-ID').count()['Book-Rating'] > 100
      padhe_likhe_users = x[x].index

[65]: filtered_rating = ratings_with_name[ratings_with_name['User-ID'].isin(padhe_likhe_users)]

[66]: y = filtered_rating.groupby('Book-Title').count()['Book-Rating']>=39
      famous_books = y[y].index

[67]: final_ratings = filtered_rating[filtered_rating['Book-Title'].isin(famous_books)]

[68]: pt = final_ratings.pivot_table(index='Book-Title',columns='User-ID',values='Book-Rating')

[69]: pt.fillna(0,inplace=True)

[70]: pt.head(39)
```

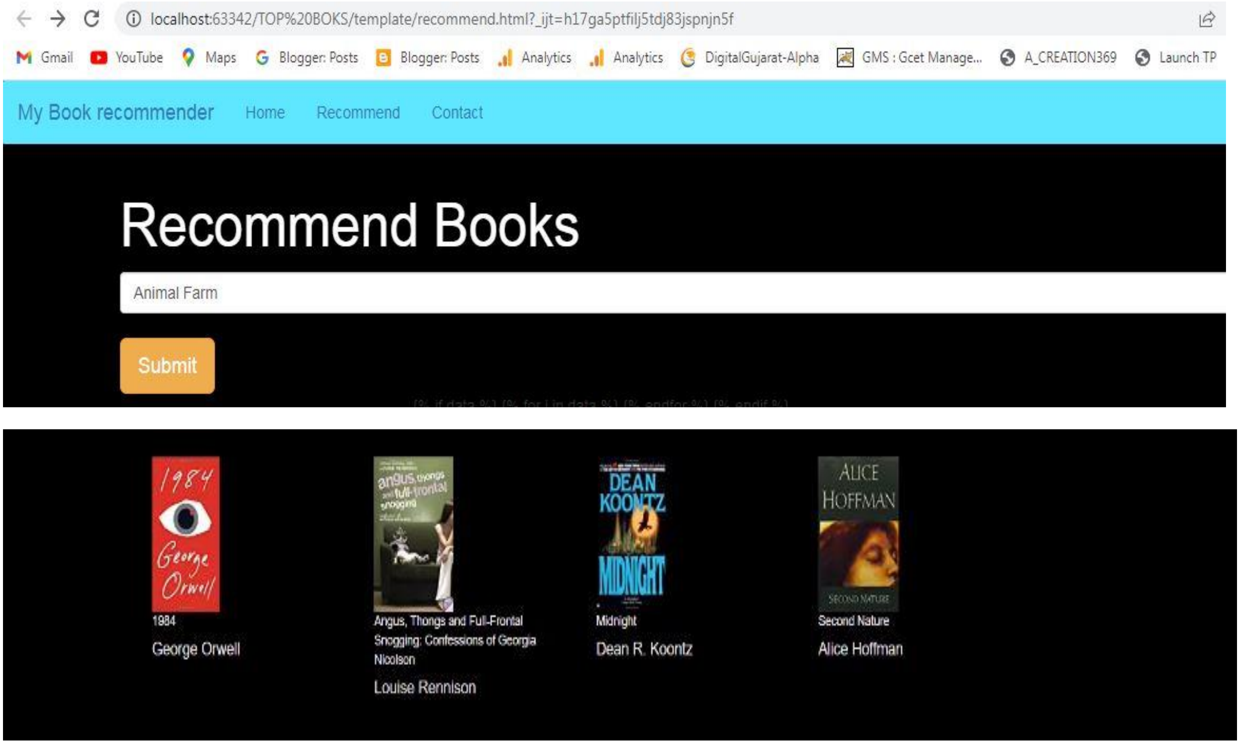
[70]:

User-ID	3363	6251	6575	7346	11601	11676	12538	12824	15408	16634	...	232131	234623	235105	236283	238120	238781	242824	245963	246655	278418
Book-Title																					
A Painted House	0.0	0.0	0.0	0.0	0.0	9.0	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
All I Really Need to Know	0.0	0.0	5.0	0.0	0.0	6.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	9.0	10.0	0.0	0.0	0.0	0.0	0.0
Harry Potter and the Sorcerer's Stone (Harry Potter (Paperback))	0.0	0.0	0.0	0.0	0.0	10.0	0.0	0.0	0.0	0.0	...	8.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
STONES FROM THE RIVER	0.0	0.0	0.0	0.0	0.0	4.0	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
Snow Falling on Cedars	0.0	8.0	7.0	0.0	0.0	10.0	0.0	0.0	0.0	8.0	...	8.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
The Bridges of Madison County	0.0	0.0	0.0	0.0	0.0	8.0	0.0	0.0	0.0	0.0	...	5.0	0.0	0.0	0.0	0.0	10.0	0.0	0.0	0.0	0.0
The Firm	0.0	0.0	0.0	0.0	0.0	1.0	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
The Kitchen God's Wife	0.0	0.0	8.0	8.0	0.0	8.0	0.0	0.0	0.0	0.0	...	0.0	0.0	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
The Perfect Storm : A True Story of Men Against the Sea	0.0	0.0	0.0	0.0	0.0	8.0	0.0	0.0	9.0	0.0	...	7.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
The Rainmaker	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	0.0	0.0

10 rows × 124 columns

```
[71]: from sklearn.metrics.pairwise import cosine_similarity

[72]: similarity_scores = cosine_similarity(pt)

[73]: similarity_scores.shape

[73]: (10, 10)

[87]: def recommend(book_name):
          # index fetch
          index = np.where(pt.index==book_name)[0][0]
          similar_items = sorted(list(enumerate(similarity_scores[index])),key=lambda x:x[1],reverse=True)[1:5]

          data = []
          for i in similar_items:
              item = []
              temp_df = books[books['Book-Title'] == pt.index[i[0]]]
              item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Title'].values))
              item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Author'].values))
              item.extend(list(temp_df.drop_duplicates('Book-Title')['Image-URL-M'].values))

              data.append(item)

          return data

[99]: recommend('A Painted House')

[99]: [["The Kitchen God's Wife",
        'Amy Tan',
        'http://images.amazon.com/images/P/0399135782.01.MZZZZZZZ.jpg'],
       ['STONES FROM THE RIVER',
        'Ursula Hegi',
        'http://images.amazon.com/images/P/0684844729.01.MZZZZZZZ.jpg'],
       ['The Bridges of Madison County',
        'Robert James Waller',
        'http://images.amazon.com/images/P/044651652X.01.MZZZZZZZ.jpg'],
       ['The Perfect Storm : A True Story of Men Against the Sea',
        'Sebastian Junger',
        'http://images.amazon.com/images/P/0060977477.01.MZZZZZZZ.jpg']]
```

Module 2



Recommend Books

Animal Farm

{% if data %} {% for i in data %} {% endfor %} {% endif %}

1984
George Orwell

Angus, Thongs and Full-Frontal Snogging: Confessions of Georgia Nicolson
Louise Rennison

Midnight
Dean R. Koontz

Second Nature
Alice Hoffman

Figure 6.1.2

Conclusion :

- A recommendation system helps an organization to create loyal customers. The recommendation system today are very powerful that they can handle the new customer too who has visited the site for the first time. They recommend the products which are currently trending or highly rated and they can also recommend the products which bring maximum profitto the company.

Future Work :

1. Given more information regarding the books dataset, namely features like Genre, Description etc, we could implement a content-filtering based recommendation system and compare the results with the existing collaborative-filtering based system.

2. We would like to explore various clustering approaches for clustering the users based on Age, Location etc., and then implement voting algorithms to recommend items to the user depending on the cluster into which it belongs.

7. References

[1] A Case-Based Recommendation Approach for Market Basket Data Anna Gatzioura and Miquel Snchez-Marr IEEE INTELLIGENT SYSTEMS 2015.

[2] Recommender Systems: An overview of different approaches to recommendations Kunal Shah, Akshaykumar Salunke, Saurabh Dongare, Kisandas Antala SIT, Lonavala India 2017

[3] Recommendation analysis on Item-based and User- based Collaboration Filtering Garima Gupta, Rahul Katarya, India

[4] Recommender systems, Handbook, Francesso Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor. Springer 2010.

[5] Zhao, Zhi-Dan, and Ming-Sheng Shang. "User-based collaborativefiltering recommendation algorithms on hadoop." In 2010 Third International Conference on Knowledge Discovery and Data Mining, pp. 478-481.
IEEE, 2010

[6] "A Study of Hybrid Recommendation Algorithm Based On User" Junrui Yang1, Cai Yang2, Xiaowei Hu3 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics

[7] Recommender Systems in E-Commerce J. Ben Schafer, Joseph Konstan, John Riedl GroupLens Research Project Department of Computer Science and Engineering University of Minnesota Minneapolis, MN 55455 1-612-625-4002

[8] https://towardsdatascience.com/brief-on-recommender-systems-b86a1068a4dd
[5.3]

[2.1] https://www.irjet.net/archives/V7/i9/IRJET-V7I9633.pdf