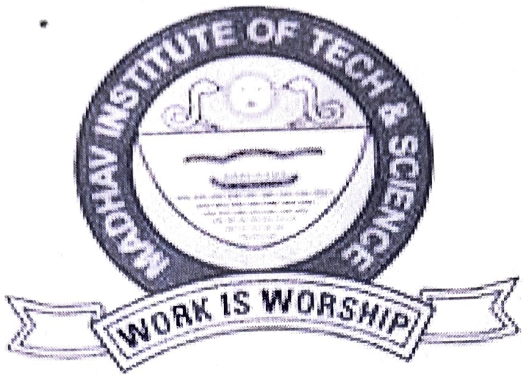
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR
(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade



Project Report

on

Collaborative Filtering (Book Recommendation System)

Submitted By:

Sneha Kaurav

0901AI211063

V Suraj Kumar

0901AI211067

Faculty Mentor:

Prof. Geetika Hazra Sharma

Assistant Professor

CENTRE FOR ARTIFICIAL INTELLIGENCE
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

JULY-DEC. 2023

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

CERTIFICATE

This is certified that **Sneha Kaurav(0901AI211063), V Suraj Kumar(0901AI211067)** has submitted the project report titled **Collaborative Filtering System** under the mentorship of **Prof. Geetika Hazra Sharma, Assistant Professor** in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **Artificial Intelligence and Robotics** from Madhav Institute of Technology and Science, Gwalior.

Geetika 23/11/2023

Prof. Geetika Hazra Sharma
Faculty Mentor
Assistant Professor
Centre for Artificial Intelligence

23/11/23

Dr. R. R. Singh
Coordinator
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in **Artificial Intelligence and Robotics** at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Prof. Geetika Hazra Sharma, Assistant Professor,** Center of Artificial Intelligence.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Sneha Kaurav
0901AI211063
3rd Year,
Centre for Artificial Intelligence

V Suraj Kumar
0901AI211067
3rd Year,
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Centre for Artificial Intelligence,** for allowing me to explore this project. I humbly thank **Dr. R. R. Singh**, Coordinator, Centre for Artificial Intelligence, **Prof. Geetika Hazra Sharma, Assistant Professor,** Center of Artificial Intelligence for her continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

V Suraj Kumar
0901AI211067
3rd Year,
Centre for Artificial Intelligence

Sneha Kaurav
0901AI211063
3rd Year,
Centre for Artificial Intelligence

ABSTRACT

In the era of pervasive Artificial Intelligence (AI), the integration of intelligent systems into various aspects of our lives has become indispensable. This recommendation system targets to recommend certain item or product to specific users on the basis of user's preference, interest, and rating. This project delves into the realm of AI-driven Recommendation Systems, exploring their significance and practical application in the context of book recommendations. With a focus on Collaborative Filtering techniques, the project aims to unravel the intricate algorithms behind these systems and elucidate their role in enhancing user experience. Through the lens of book recommendations, the project demonstrates the versatility and impact of Recommendation Systems across diverse domains, showcasing their potential to shape user behaviour and market trends. Collaborative filtering is a widely used technique in book recommendation systems. It involves analyzing the reading or purchasing history of other users to make recommendations for new books.

सार

आर्टिफिशियल इंटेलिजेंस (AI) के अग्रगामी युग में, बुद्धिमान प्रणालियों का जीवन के विभिन्न पहलुओं में से एक साथी बन गया है। यह परियोजना बुक सुझावों की संदर्भ में एक नए दृष्टिकोण में AI-संचालित सुझाव प्रणालियों की दुनिया में प्रवेश करती है, उनके महत्वपूर्णता को समझती है और उनके व्यावासायिक अनुप्रयोग को बढ़ावा देने की कोशिश करती है।

आर्टिफिशियल इंटेलिजेंस के इस युग में सूचना और प्रौद्योगिकी की संगम रचते हुए, सुझाव प्रणालियों का विशेष स्थान है। ये मशीन लर्निंग एल्गोरिदम्स का उपयोग करके उपयोगकर्ता पसंदों को विश्लेषण करने के लिए बनाई जाती हैं, उपयोगकर्ता अनुभव को बढ़ाने के माध्यम से। चाहे वह उत्पाद, सामग्री, या सेवाएं हों, इन सुझाव प्रणालियों जैसे सिस्टम्स ने अल्ग-अल्ग क्षेत्रों में अपने अंश को अद्वितीय बनाया है। इस परियोजना का उद्देश्य है बुक सुझावों के संदर्भ में, सहयोगात्मक फ़िल्टरिंग तकनीकों पर ध्यान केंद्रित करके इन प्रणालियों को समझाना है और इसके पीछे के जटिल एल्गोरिदम्स को स्पष्ट करना है और इनके योगदान को सुधारने में योगदान करना है।

इस परियोजना का उद्देश्य सहयोगात्मक फ़िल्टरिंग तकनीकों का उपयोग करके बुक सुझाव प्रणाली बनाना है और इस प्रणाली और इसमें उपयुक्त एल्गोरिदम्स के बारे में जानकारी प्रदान करना है। हम एक सहयोगात्मक फ़िल्टरिंग आधारित मॉडल बनाएंगे जो उपयोगकर्ताओं को उनकी पूर्व पसंदों और चयनों के आधार पर विभिन्न उत्पादों का सुझाव करेगा। सुझाव प्रणालियों ने हाल के वर्षों में बढ

1. Chapter 2

2. Chapter 3

3. Chapter 4

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Artificial Intelligence has become ubiquitous in our lives. Every field is trying to integrate AI tools and technologies to bring out further advancements and take things to next level. Artificial Intelligence is used in different field and has vast scope. Artificial Intelligence is used by different websites and browsers to recommend their users with products, content and services by integrating a Recommendation System. Recommendation systems are machine learning algorithm that uses various criteria and techniques to enhance user experience, thus increasing sales or customer satisfaction. Based on user's preference or liking, suggestions are made on different basis like what was previously viewed, bought, visited, searched etc. The engagement of user is studied and then used to employ such models. Suggestions on browsers like Google Chrome, Brave, MS Bing etc. for similar content that a user searches or; on websites like Youtube, Amazon, Netflix etc. for 'you may also like' are all examples of recommendation system.

Basically, Recommendation system uses Big Data to make the best possible suggestion. These can broadly be divided into 3 types based on the method used in the algorithm: Content-based Filtering, Collaborative Filtering and Hybrid Recommender system.

There are multiple benefits of incorporating it into any company's website, from providing customised user experience to improving engagement. According to a survey, product recommendations can account for up to 31% of e-commerce revenues. Amazon, the biggest online retailer in the world, drives 35% of its sales from its recommendation engine. Some of the benefits that companies enjoy on their online websites are:

- Increase in Sales: It may lead to impulsive shopping and presents a temptation to buy something which otherwise may not be on the list of items to be purchased. Around 45% of the users admit that they have bought products that they didn't intend to buy only because they were recommended to them.

- Customer Retention: If the suggestions made by any website is in accordance to the user's need and preference, then they are likely to visit again. This helps in retaining the users for that website and increasing churn rate.
- Influence Customer Habits and Market Trends: Due to increased convenience, customers will create a habit or pattern in their consumption of specific kind of products/services.

1.2 Objectives and Scope

The objective of the project is to create a recommendation system using collaborative filtering and to impart information about the recommendation system and various algorithms used to build them. We will create a Collaborative Filtering based model that will recommend similar products to different users based on their past preferences and choices.

Recommendation systems have become increasingly popular in recent years, and their scope is rapidly expanding across a variety of industries and applications, including e-commerce, media streaming, and social networks. Its scope is broad as it is used in almost every field like healthcare, agriculture, fashion and beauty, lifestyle, entertainment, retail etc.

1.3 Project Features

This project uses recommendation system for book recommendations. There are many books out in the market and one person can't read all. Some of us read books as a source of entertainment and some others for learning and bunch of us may read for self-growth and development. To make it easier to choose a book that one may like (based on their style and reading log). Here, we have used two methods to do the same. These are:

1.3.1 Content-based Filtering:

In this filtering, the recommendations are made keeping the products in mind. The product with highest rating may be suggested. Content-based filtering means to filter the products that are similar to each other. For example, in our project of book recommendation, we have books of different genres that the user may like.

1.3.2 Collaborative Filtering:

Collaborative filtering is a popular technique in machine learning for building recommendation systems. Collaborative filtering can be of following types: User-based collaborative filtering, Item-based collaborative filtering. We have implemented item-based filtering in our book recommendation engine. This approach focuses on the relationships between items rather than users. Measure the similarity between items based

Chapter 2: DATA PREPROCESSING

2.1 Importing Libraries

The first step to build and train any machine learning model is importing required libraries. The libraries used in this project are:

2.1.1 Pandas

Pandas is an open-source (software or other resources that can be used free-of-cost for manipulation and distribution) machine learning library which is built on top of python. It is used as an analysis tool for programming in python and helps in data-manipulation. It is designed to easily use high-level (easy to understand) data structures and functions that are needed to work with structured data. Generally abbreviated as pd.

2.1.2 Numpy

Numpy is imported as np to make large multi-dimensional arrays and matrices available for use. You can also perform mathematical functions- perform linear algebra-in python based projects. It supports parallel computing using multiprocessing and multithreading tools.

2.1.3 Matplotlib

A popular library used for data visualisation and performing exploratory data analysis and scientific researches on the provided datasets. Matplotlib is older than seaborn which is another library for data visualisation. It provides a collection of tools for creating visualisations in 2D and 3D. Generally abbreviated as plt.

2.1.4 Seaborn

Seaborn is another popular library that serves the purpose for data visualisation. It is built on Matplotlib. It uses the figure and axes of plt to plot its graph. Like Matplotlib, it also allows the programmer to plot various kinds of graphs, plots and charts but more.

2.1.5 Scikit Learn

Scikit learn, abbreviated as sklearn, is a powerful machine learning library. It provides various tools for claasification, regression, clustering, encoding, normalisation etc. It simplifies the development of machine learning model. In our project, we have imported cosine_similarity from metrics module of sklearn library.

2.2 Loading Datasets directly from Kaggle

The data used here is a structured(tabular) data. Datasets are imported in zipfile format directly from Kaggle.com. The snippet of code used to perform the task is given below. Functions of each line of code are mentioned as comments.

```
!ls -lha kaggle.json
!pip install -q kaggle # installing the kaggle package
!mkdir -p ~/.kaggle # creating .kaggle folder where the key should be placed
!cp kaggle.json ~/.kaggle/ # move the key to the folder
!pwd # checking the present working directory
!chmod 600 ~/.kaggle/kaggle.json
```

```
-rw-r--r-- 1 root root 67 Nov 23 02:27 kaggle.json
/content
```

fig 2.1

Unzipping the zipfile to extract all the datasets in it.

```
zip = zipfile.ZipFile("/content/book-recommendation-dataset.zip")
zip.extractall("/content")
zip.close()
```

fig 2.2

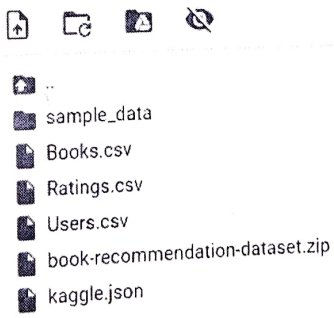


fig 2.3

2.3 Data Preprocessing

After the data is extracted, it is pre-processed. Preprocessing is cleaning, transforming raw data to a cleaned format. It is done to check the quality of data before any machine learning algorithm or analysis is applied to it. It is an important step of data mining. This includes handling missing values, dealing with outliers, normalisation and standardisation, dimensionality-reduction, encoding categorical data etc.

The data processing done on our project is very simple because the extracted data was already very clean.

- dataframe.info(): gives the information like range-index, data-columns count and name, total non-null values, dtype of a variable and memory usage in a dataframe.

```
books.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 271360 entries, 0 to 271359
Data columns (total 8 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   ISBN                 271360 non-null  object
 1   Book-Title           271360 non-null  object
 2   Book-Author          271359 non-null  object
 3   Year-Of-Publication  271360 non-null  object
 4   Publisher            271358 non-null  object
 5   Image-URL-S          271360 non-null  object
 6   Image-URL-M          271360 non-null  object
 7   Image-URL-L          271357 non-null  object
dtypes: object(8)
memory usage: 16.6+ MB
```

fig 2.4

- Dataframe.isnull().sum() : gives sum of the null values for each variable in a dataframe.

```
books.isnull().sum()

ISBN                 0
Book-Title           0
Book-Author          1
Year-Of-Publication  0
Publisher            2
Image-URL-S          0
Image-URL-M          0
Image-URL-L          3
dtype: int64
```

fig 2.5

- Dataframe.duplicated().sum : gives sum of the total duplicated values(rows) in the dataframe.

```
books.duplicated().sum()
```

```
0
```

fig 2.6

- Dataframe.dropna(): delete all the rows that have a null value for any variable.

```
rating_with_name = ratings_with_name.dropna()
rating_with_name.shape
```

```
(1031134, 8)
```

fig 2.7

- sns.heatmap(df.corr()): gives a correlation matrix between each variable in a dataframe.

```
sns.heatmap(ratings_with_name.corr(), annot=True, cmap='coolwarm', fmt=".2f", linewidt
```

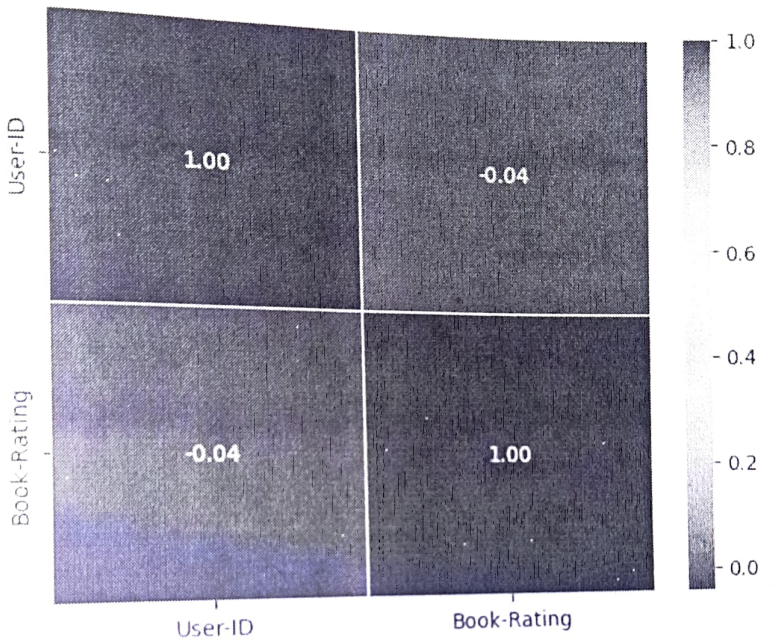


fig 2.8

Chapter 3

50 MOST POPULAR BOOKS

3.1 Introduction

There are variety of books in the market which caters to interest of varied public groups and individuals. Some books may have a niche of readers while other books may be popular among masses because of its inclusiveness or mass appeal. Such books that resonate with a huge amount of people becomes best-selling-books. Same book can be appreciated by one but not someone else. For a book to become popular, it should be worth the penny spend for most people. Popular books are those which are read as well as liked by the greatest number of people. Novels, finance books and motivational books leads the sales generally. Like the very loved fictional wizarding world of Harry Potter created by J.K. Rowling or well-appreciated advices from the Rich Dad, Poor Dad by Robert Kiyosaki. In our book recommendation system, popular books are sorted in descending order of their popularity ranking from 1 to 50.

3.2 Approach

The approach used in listing the 50 most popular books is mentioned in these steps.

1. Understanding the dataset: The dataframes (books.csv, reviews.csv and users.csv) are understood first during pre-processing.
2. Building required dataframe: From the set of given dataframes, merge different dataframes on common columns. Drop the inessential columns in the dataframe.
3. Applying conditions: Applying the condition for the books and users after they make a common matrix-like dataframe. The conditions applied are:
    - Calculate the total number of ratings for each book.
    - Calculate average rating of each book.
    - The book must have more than or equal to 250 ratings count.
4. Display the list: After dropping any existing duplicates in the dataframe, you have your list of most popular books. Select the top 50 books from the obtained dataframe using head(). This is the final list.

3.3 Code

Following is the code used for extracting the 50 most popular books.

```
ratings_with_name = books.merge(ratings, on='ISBN').drop(columns=['Image-URL-L','Image-URL-S'])
ratings_with_name.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1031136 entries, 0 to 1031135
Data columns (total 8 columns):
 #   Column               Non-Null Count    Dtype
---  ------               --------------    -----
 0   ISBN                 1031136 non-null  object
 1   Book-Title           1031136 non-null  object
 2   Book-Author          1031136 non-null  object
 3   Year-Of-Publication  1031136 non-null  object
 4   Publisher            1031134 non-null  object
 5   Image-URL-M          1031136 non-null  object
 6   User-ID              1031136 non-null  int64
 7   Book-Rating          1031136 non-null  int64
dtypes: int64(2), object(6)
memory usage: 70.8+ MB
```

fig 3.1

```
books_rated = ratings_with_name.groupby('Book-Title').count()['Book-Rating'].reset_index()
books_rated_avg = books_rated.groupby('Book-Title')['Book-Rating'].mean().reset_index()
books_rated_avg.rename(columns={'Book-Rating': 'avg-rating'},inplace=True)
books_rated_avg = ratings_with_name.groupby('Book-Title')['Book-Rating'].mean().reset_index()
books_rated_avg.rename(columns={'Book-Rating': 'avg-rating'},inplace=True)
popular_books = books_rated.merge(books_rated_avg, on='Book-Title')
popular_books.head()
```

	Book-Title	Book-Rating	avg-rating
0	A Light in the Storm: The Civil War Diary of ...	4	4.0
1	Always Have Popsicles	1	1.0
2	Apple Magic (The Collector's series)	1	1.0
3	Ask Lily (Young Women of Faith: Lily Series, ...	1	1.0
4	Beyond IBM: Leadership Marketing and Finance ...	1	1.0

fig 3.2

```
popular_books = popular_books[popular_books['Book-Rating']>250].sort_values('avg-rating',ascending=False)
popular_books = popular_books.merge(books,on='Book-Title').drop_duplicates('Book-Title')
[['Book-Title','Book-Author','Image-URL-M']]
```

3.4 Output

Here, sharing Top 10 most popular books as output instead of 50 by using popular_books.head(10).

```
popular_books.head(10)
```

	Book-Title	Book-Author	Image-URL-M
0	Wild Animus	richshapero	http://images.amazon.com/images/P/0971880107.0...
1	The Lovely Bones: A Novel	alicesebold	http://images.amazon.com/images/P/0316666343.0...
2	The Da Vinci Code	danbrown	http://images.amazon.com/images/P/0385504209.0...
8	A Painted House	johngrisham	http://images.amazon.com/images/P/044023722X.0...
13	The Nanny Diaries: A Novel	emmamclaughlin	http://images.amazon.com/images/P/0312278586.0...
15	Bridget Jones's Diary	helenfielding	http://images.amazon.com/images/P/0330332775.0...
21	The Secret Life of Bees	suemonkkidd	http://images.amazon.com/images/P/0142001740.0...
27	Divine Secrets of the Ya-Ya Sisterhood: A Novel	rebeccawells	http://images.amazon.com/images/P/0060928336.0...
30	The Red Tent (Bestselling Backlist)	anitadiamant	http://images.amazon.com/images/P/0312195516.0...
31	Angels & Demons	danbrown	http://images.amazon.com/images/P/0671027360.0...

fig 3.3

Chapter 4

COLLABORATIVE FILTERING RECOMMENDATION SYSTEM

4.1 Introduction

Recommendation System, as we have discussed earlier, are broadly of three types:

1. **Content-based Filtering Recommendation System**

   In content-based filtering, similar products are recommended based on the type or other similarity parameters. This approach will suggest the products, here books, similar to the books previously liked by that user. For. Example: If User A likes to read detective books and rates them highly, then she/he will be suggested book of similar genre, i.e., revolving around mystery and detective stories.

2. **Collaborative Filtering Recommendation System**

   In Collaborative Filtering, similarity between the preferences of various users is observed and calculated. If User A has similar taste as User B, then it is very likely that User B will be suggested a product rated highly by User A and is not yet chosen by User B, and vice versa.

3. **Hybrid Recommendation System (Content-based Filtering + Collaborative Filtering)**

   Hybrid Recommendation System is the combination of both Content-based Filtering and Collaborative Filtering. It means that hybrid recommendation algorithm considers users personal preferences as much as it regards the products that were shown fondness by similar users.

4.2 Approach

The approach used in listing the books liked by users with similar reading log is mentioned in thes steps. The steps are broadly similar to those mentioned above with difference in the conditio applied and dataframes created.

1. Understanding the dataset: The dataframes (books.csv, reviews.csv and users.csv) are understood first during pre-processing.
2. Building required dataframe: From the set of given dataframes, merge different dataframes on common columns. Drop the inessential columns in the dataframe. The user-item matrix can be formed to perform the task.
3. Applying conditions: Applying the condition for the books and users after they make a common matrix-like dataframe. The conditions applied are:
   - The book must have more than or equal to 250 ratings count.
   - The users/readers must have given more than or equal to 50 ratings.
4. Create User-Item Matrix: User-Item matrix is created by using pivot in python. Vectorisation can also be used. In our project, the corresponding matrix is Book-Title as rows and User-ID as column.
5. Calculate Similarity Scores: Calculate similarity scores using cosine_similarity() on the final dataframe(user-item matrix).
6. Recommend: Code a recommendation function with 'book_name' as functional argument. 'n' number of products can be recommended.

4.3 Code

The code for the required task is as follows:

```
x = ratings_with_name.groupby('User-ID').count()['Book-Rating'] > 200
avid_readers = x[x].index
filtered_rating = ratings_with_name[ratings_with_name['User-ID'].isin(avid_readers)]
y = filtered_rating.groupby('Book-Title').count()['Book-Rating']>=50
famous_books = y[y].index
final_ratings = filtered_rating[filtered_rating['Book-Title'].isin(famous_books)]
pt = final_ratings.pivot_table(index='Book-Title',columns='User-ID',values='Book-Rating')
pt.head(4)
```

User-ID	254	2276	2766	2977	3363	4017	4385	6251	6323	6543	...	271705	273979	274004	274061	274301	274308	275970
Book-Title																		
1984	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	10.0	0.0	0.0	0.0	0.0	0.0	0.0
1st to Die: A Novel	0.0	0.0	0.0	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
2nd Chance	0.0	10.0	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
4 Blondes	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

4 rows × 810 columns

fig 4.1

```
similarity_scores = cosine_similarity(pt)

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
```

fig 4.2

4.4 Output

```
recommend('1984')
```

```
[['Animal Farm',
  'georgeorwell',
  'http://images.amazon.com/images/P/0451526341.01.MZZZZZZZ.jpg'],
 ["The Handmaid's Tale",
  'margaretatwood',
  'http://images.amazon.com/images/P/0449212602.01.MZZZZZZZ.jpg'],
 ['Brave New World',
  'aldoushuxley',
  'http://images.amazon.com/images/P/0060809833.01.MZZZZZZZ.jpg'],
 ['The Vampire Lestat (Vampire Chronicles, Book II)',
  'annerice',
  'http://images.amazon.com/images/P/0345313860.01.MZZZZZZZ.jpg']]
```

fig 4.3