**INDEX**

**TITLE PAGENO**

**CHAPTER 1: INTRODUCTION 3-4**

1.1 OVERVIEW 4

1.2 PURPOSE 4

**CHAPTER 2: LITERATURE SURVEY 5-7**

2.1 EXSISTING PROBLEM 6

2.2 PROPOSED SOLUTION 7

**CHAPTER 3: THEORITICAL ANALYSIS 9-10**

3.1 BLOCK DIAGRAM 10

3.2 HARDWARE/SOFTWARE DESIGNING 10

**CHAPTER 4: EXPERIMENTAL INVESTIGATIONS 11-19**

**CHAPTER 5: FLOWCHART 20**

**CHAPTER 6: RESULT 21**

**CHAPTER 7: ADVANTAGES & DISADVANTAGES 22**

**CHAPTER 8: APPLICATIONS 23**

**CHAPTER 9: CONCLUSION 24**

**CHAPTER 10: FUTURE SCOPE 24**

**CHAPTER 11: BIBILOGRAPHY 25**

**CHAPTER 12: APPENDIX 26-35**

12.1 SOURCE CODE 26

12.2 UI OUTPUT SCREENSHOT 33

**CHAPTER 1**

**INTRODUCTION**

**1.1 OVERVIEW :**

In this age of information overload, people use a variety of strategies to make choices about what to buy, how to spend their leisure time, and even where to go. Recommendation systems automate some of these strategies with the goal of providing affordable, personal and high-quality recommendations. So, with the help of different techniques like collaborative and content-based filtering, we can build our own recommendation systems.

This Online book selling websites helps to buy the books online with Recommendation system which is one of the stronger tools to increase profit and retaining buyer. The book recommendation system must recommend books that are of buyer’s interest. Recommendation systems are widely used to recommend products to the end users that are most appropriate. This system uses features of collaborative filtering to produce efficient and effective recommendations. Collaborative recommendation is probably the most familiar, most widely implemented and most mature of the technologies. Collaborative recommender systems aggregate ratings of objects, recognize commonalities between users on the basis of their ratings, and generate new recommendations.

**1.1 PURPOSE :**

The main objective of the project is to build an application where the user is prompted with various types of book recommendations based on the input given by the user.

* To recommend books according to the input provided by the user.
* In the proposed book recommendation engine, books will be displayed according to the readers preferences in a hierarchical way to categorize readers interest in different genres, the users pattern of searching different books and to form an effective set of rules.
* Based on users‟ interest and books properties, a book recommendation system will be generating best and efficient book recommendations.

**CHAPTER 2**

**LITERATURE SURVEY**

# Recommender system has been so extensively used these days that it has become a preferable choice for researchers. First paper on recommender system was published in year 1998. Since then a significant number of papers had been published. Different factors have been explained to increase the reliability of recommender system. Mathwicketal., (2002) , “if online shopping meets this ideal by enabling the consumer to accomplish the shopping task he or she has set out to perform , then consumers will judge the Internet shopping performance positively. Childers et al., (2001) found “‘enjoyment’ to be a consistent and strong predictor of altitude towards online shopping experience, they have a more likely to adopt the Internet as a Shopping medium.” According to Hoge (1993), Electronic marketing (EM) is the transfer of goods or services from seller to buyer that involves one or more electronic methods or media. EMarketing began with the use of telegraphs in the nineteenth century. With the advent and mass acceptance of the telephone, radio, television, and then cable, electronic media have become the dominant marketing force. Hoge’s (1993) idea of E-marketing is simple but it does not touch the important aspect of customer relationship. Strauss and Ansary (2006) defined E-marketing in their latest book as the use of information technology in the process of creating, communicating, and delivering value to customers, and for managing customer relationships in ways that benefit the organization and its stake holders. Electronic Commerce (e-commerce) applications support the interaction between different parties participating in a commerce transaction via the network, as well as the management of the data involved in the process.The task of recommender algorithm concerns the prediction of the users rating for the target item that the user has rated, based on the user’s ratings on observed items. The increasing importance of e-commerce is apparent in the study conducted by researchers at the GVU (Graphics, Visualization, and Usability) Center at the Georgia Institute of Technology. In their summary of the findings from the eighth survey, the researchers report that "e-commerce is taking off both in terms of the number of users Shopping as well as the total amount people are spending via Internet based transactions".

**2.1 EXSISTING SYSTEM :**

The existing recommendation systems ones only consider one technique to recommend items to the users.

They do not recommend items using two or more techniques and are not Hybrid Recommendation System.

The disadvantages of existing system are as follows:

* required to understand item content
* item cold-start problem
* Do not capture the change in user interests over time
* Do not capture inherent subtle characteristics.

**2.2 PROPOSED SOLUTION:**

A recommendation system will help users who do not have enough individual knowledge through the different types of options offered by a website.it will provide the users with in them to make a decision as to which items to purchase. The proposed work alters from the existing recommendation systems since the existing ones only consider one technique to recommend items to the users.

They do not recommend items using two or more techniques and are not Hybrid Recommendation System.

The proposed system uses a combination of Collaborative Filtering and Association Rule Mining. Collaborative Filtering is used to predict the ratings of a particular item by calculating ratings given to similar items or ratings between different items. Thus the usage of these techniques can help to control the data sparsity problem and cols start problem in recommendation systems.

**CHAPTER 3**

**THEORITICAL ANALYSIS**

Classification techniques are an essential part of machine learning and data mining applications. Approximately 70% of problems in Data Science are classification problems. There are lots of classification problems that are available, but the a particular product or will they churn another competitor, whether the user will click on a given advertisement link or not, and many more examples are in the bucket.

K Nearest Neighbor algorithm falls under the Supervised Learning category and is used for classification (most commonly) and regression. It is a versatile algorithm also used for imputing missing values and resampling datasets. As the name (K Nearest Neighbor) suggests it considers K Nearest Neighbors (Data points) to predict the class or continuous value for the new Datapoint.

The algorithm’s learning is:

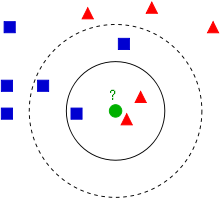
1. Instance-based learning: Here we do not learn weights from training data to predict output (as in model-based algorithms) but use entire training instances to predict output for unseen data.

2. Lazy Learning: Model is not learned using training data prior and the learning process is postponed to a time when prediction is requested on the new instance.

3. Non -Parametric: In KNN, there is no predefined form of the mapping function.

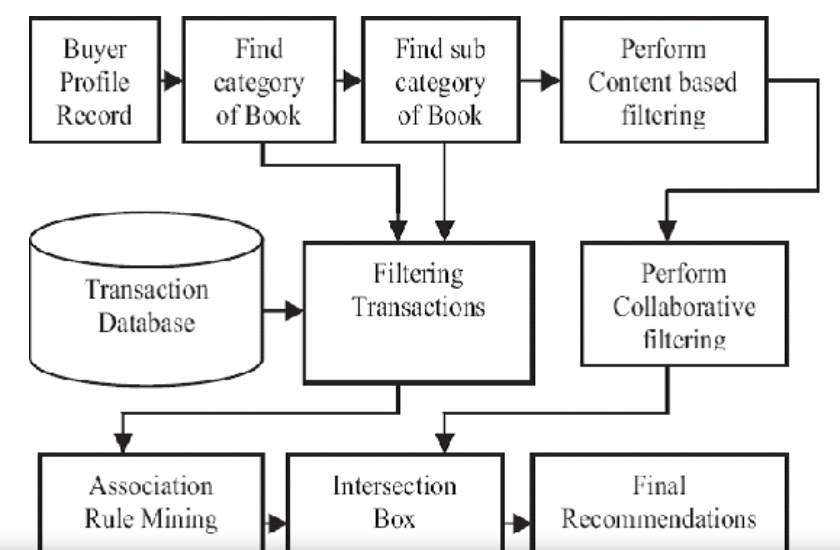
**Principle:**

The principle behind K Nearest Neighbors is here, nearest neighbors are those data points that have minimum distance in feature space from our new data point. And K is the number of such data points we consider in our implementation of the algorithm. Therefore, distance metric and K value are two important considerations while using the KNN algorithm. Euclidean distance is the most popular distance metric. We can also use Hamming distance, Manhattan distance, Minkowski distance as per our need. For predicting class/ continuous value for a new data point, it considers all the data points in the training dataset. Finds new data point’s ‘K’ Nearest Neighbors (Data points) from feature space and their class labels or continuous values.



Example of *k*-NN classification. The test sample (green dot) should be classified either to blue squares or to red triangles. If *k = 3* (solid line circle) it is assigned to the red triangles because there are 2 triangles and only 1 square inside the inner circle. If *k = 5* (dashed line circle) it is assigned to the blue squares (3 squares vs. 2 triangles insideter circle)

**3.1 BLOCK DIAGRAM**



# (a) Block diagram of a recommender system; (b) Block diagram of the proposed recommender system for cold users

# 3.2 HARDWARE AND SOFTWARE DESIGNING

**HARDWARE**

* Processor: Minimum 1 GHz; Recommended 2GHz or more.
* Hard Drive: Minimum 32 GB; Recommended 64 GB or more.
* Memory (RAM): Minimum 1 GB; Recommended 4 GB or above.

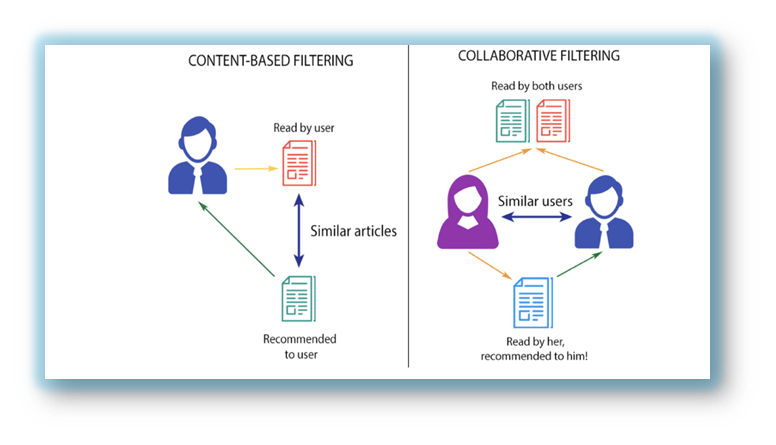
**SOFTWARE**

* Anaconda Individual Edition
* Any Browser

**CHAPTER 4**

**EXPERIMENTAL INVESTIGATIONS**

A user study is conducted by recruiting a set of test subject, and asking them to perform several tasks requiring an interaction with the recommendation system. While the subjects perform the tasks, we observe and record their behaviour, collecting any number of quantitative measurements, such as what portion of the task was completed, the accuracy of the task results, or the time taken to perform the task. In many cases we can ask qualitative questions, before, during, and after the task is completed. Such questions can collect data that is not directly observable, such as whether the subject enjoyed the user interface, or whether the user perceived the task as easy to complete. A typical example of such an experiment is to test the influence of a recommendation algorithm on the browsing behaviour of news stories. In this example, the subjects are asked to read a set of stories that are interesting to them, in some cases including related story recommendations and in some cases without recommendations. We can then check whether the recommendations are used, and whether people read different stories with and without recommendations. We can collect data such as how many times a recommendation was clicked, and even, in certain cases, track eye movement to see whether a subject looked at a recommendation. Finally, we can ask quantitative questions such as whether the subject thought the recommendations were relevant. Of course, in many other research areas user studies are a central tool, and thus there is much literature on the proper design of user studies. This section only overviews the basic considerations that should be taken when evaluating a recommender system through a user study, and the interested reader can find much deeper discussions elsewhere



**Implementation :**

To build a machine learning model, For that we have used the Jupyter notebook and Flask Application for training the model and deploying it. Building a machine learning includes the following steps.

1. Data Collection.
2. Data Pre- processing.
3. Collaborative Filterimg
4. Model Building
5. Flask Application Building

7. Deployment of the model

**Data Preprocessing:**

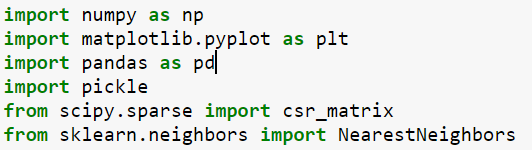
Data preprocessing includes of 5 steps. They are:

1. Importing Libraries and Reading the Dataset
2. Performing Data Analysis
3. Performing Data visualization

Firstly we need to import the data to the operating environment. For loading the data sets and to preprocess them we need to import the libraries such as pandas, numpy and matplotlib for visualization.

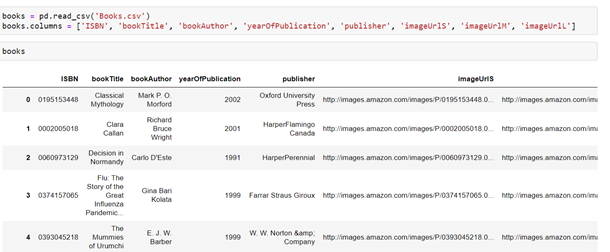
**Step 1:** Importing the Libraries

**Pandas:** It is a python library mainly used for data manipulation.  
**NumPy:** This python library is used for numerical analysis.  
**Matplotlib:** It is a data visualization library used for plotting graph which will help us for understanding the data.  
**csr\_matrix() :**A dense matrix stored in a NumPy array can be converted into a sparse matrix using the CSR representation by calling the csr\_matrix() function.  
**Pickle:** to serialize your machine learning algorithms and save the serialized format to a file.



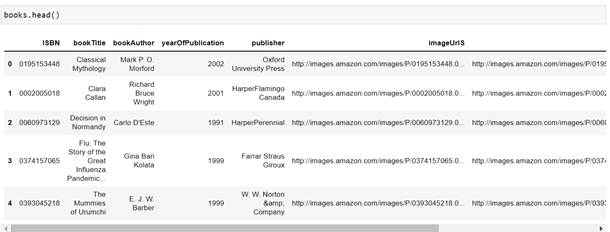
ii) Read the dataset and displaying the dataset

Reading the books datasets



**Step 2**: Performing the exploratory data analysis which helps to analyze the data sets to summarize their main characteristics and to discover patterns, spot anomalies ,test a hypothesis or check assumptions

* To check the first five rows of the dataset, we have a function call head( ).

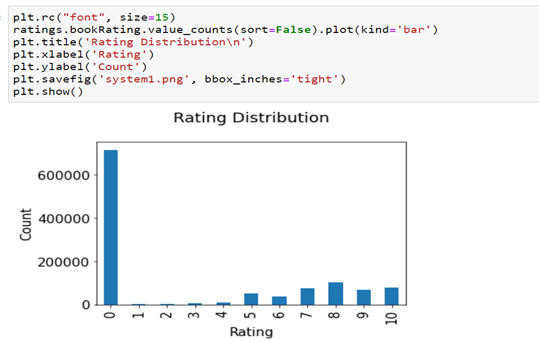


* books,info() gives the information of columns present dataset. (like datatype , number of values present etc)

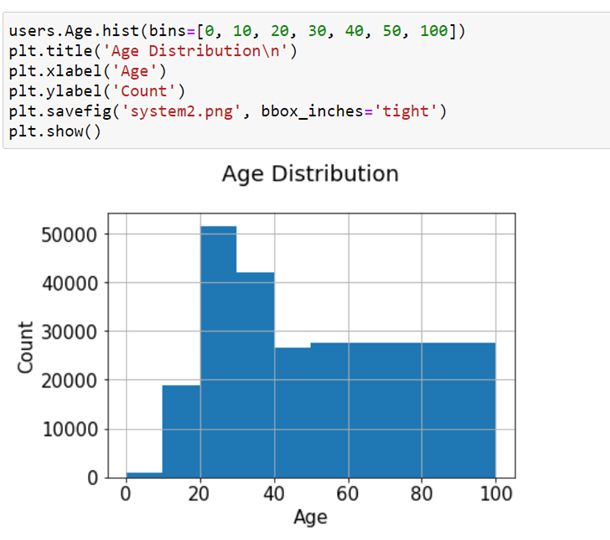
### 

**Step 3:** Performing the Data Visualization

* At first, we will be plotting a bar plot using matplotlib for showing the count of rating with respect to different rating which are present in our dataset.



* Here if we observe in the above plot we got to know that most of data contains 0 rating
* Now as in our dataset we have an Age column we will plot a histogram to showcase the age distribution among the different ages

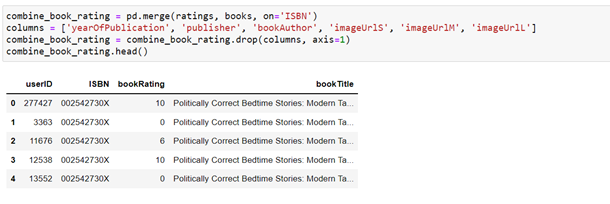


**Step 4: Building the model**

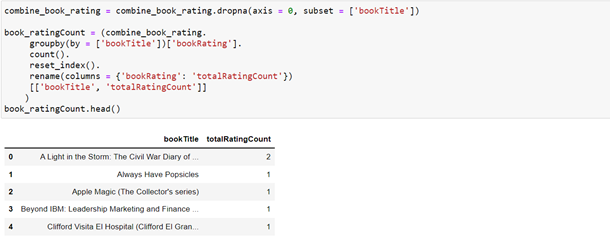
* As our dataset is analyzed now we use collaborative filtering to train our model.
* Collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating). In this particular milestone we will be performing Collaborating Filtering using K – nearest neighbors to build our recommendation system.



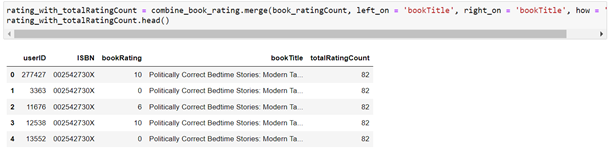
* Now with the help of the same columns (ISBN), we will be merging two datasets and remove some of the columns which are not necessary for our recommendation system.



* After merging the dataset we will be calculating the count of ratings of all the books and add that column to our new updated dataset.



* Now we will create a new variable inside which we will be merging the results which we got after adding the total rating count to the previous dataset.



* As our dataset is so big and training takes a lot of time so let's select the features for only two specific location for the recommendation

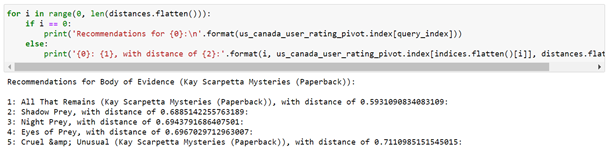


Finally, we will be creating our dataset into a 2D matrix which we will be providing to our model. With the use of scipy library.



**Step 5:** Testing the model

At last we will be predicting our result by giving some random index value to our model and with that index the book which is assigned our knn model will going to give the recommendations.



**Saving the Model:**

Pickle is used for serializing and de-serializing Python object structures, also called marshalling or flattening. Serialization refers to the process of converting an object in memory to a byte stream that can be stored on disk or sent over a network.



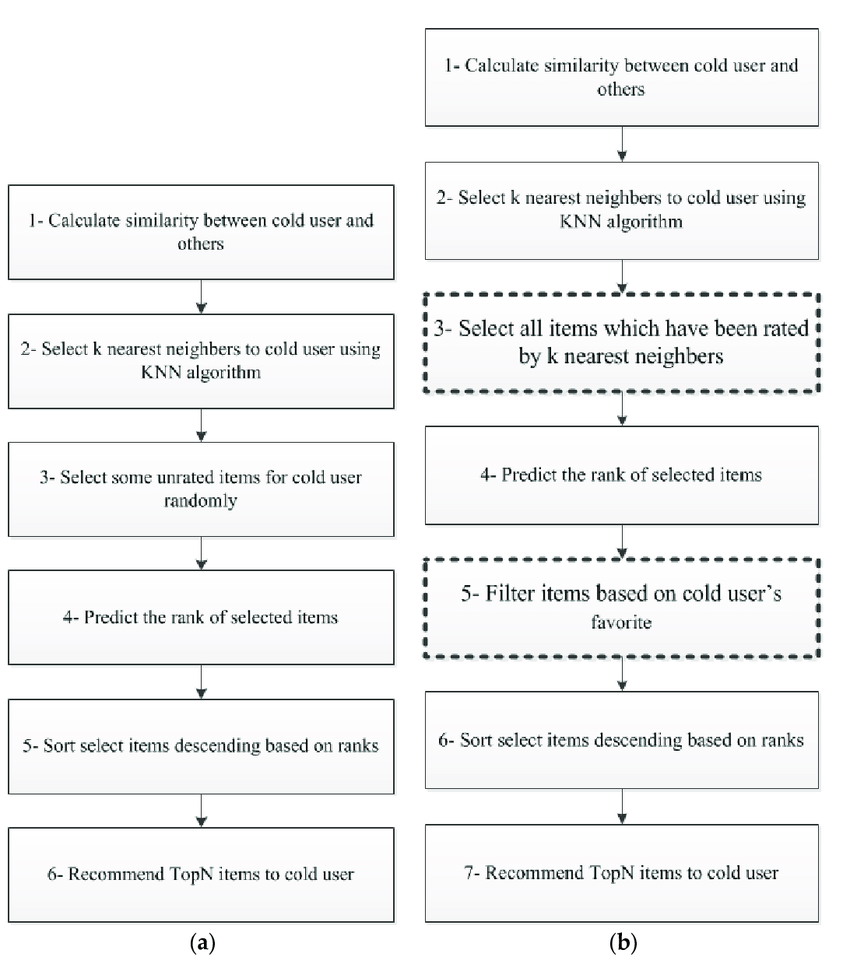
**Deploying the Prediction:**

The trained model is deployed in the Jupyter Notebook and UI is developed using the Flask Application using Spyder in Anaconda . It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine.Both are Pocco projects.

Flask is a web framework, it’s a Python module that lets you develop web applications easily. It’s has a small and easy-to-extend core: it’s a microframework that doesn’t include an ORM (Object Relational Manager) or such features.It does have many cool features like url routing, template engine. It is a WSGI web app framework.It is simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management.

**CHAPTER 5**

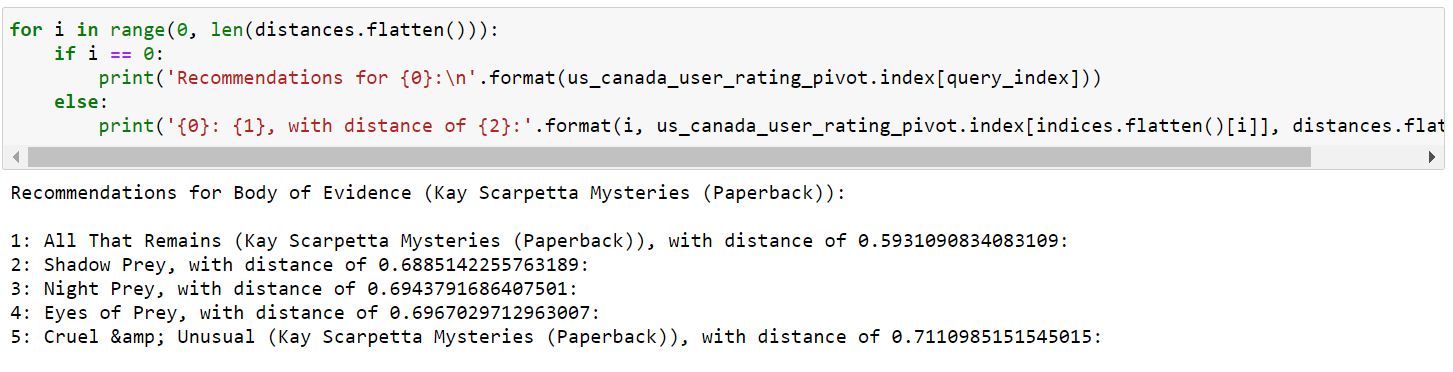
**FLOWCHART**

(a) Block diagram of a recommender system; (b) Block diagram of the proposed recommender system for cold users.

**CHAPTER 6**

**RESULT**

* At last model is predicting result by taking some random index value and with that index the book which is assigned our knn model will going to give the recommendations



* Here recommendations for the Book of Body of Evidence and recommeding the user with five similar genere books .
* After development of flask application we can see the result in our browser in a more efficient matter and the developed flask application output will be in further chapter

**CHAPTER 7**

**ADVANTAGES & DISADVANTAGES**

**Advantages****:**

* This system saves the precious time of customer and very efficient to use.
* Provides large number of choices for books & also recommend for books.
* User can buy book easily by making online payment.
* The system recommending algorithm scale well with co-rated items.
* The model can help users discover new interests. In isolation, the ML system may not know the user is interested in a given item, but the model might still recommend it because similar users are interested in that item.
* To some extent, the system needs only the feedback matrix to train a matrix factorization model. In particular, the system doesn't need contextual features. In practice, this can be used as one of multiple candidate generators.

##### **Disadvantages:**

* Dependent on human ratings for books.
* The prediction of the model for a given (user, item) pair is the dot product of the corresponding embeddings. So, if an item is not seen during training, the system can't create an embedding for it and can't query the model with this item. This issue is often called the cold-start problem. However, the following techniques can address the cold-start problem to some extent.
* Hard to include side features for query/item.

**CHAPTER 8**

**APPLICATIONS**

1. Book recommendation system a common scenario which we can see consists of set users and books. Books are rated by various who have read it previously and also these users give review about the book. These reviews can help to predict what genre people may like the book. The opinion whether the book is good or bad also matters a lot.
2. There are various sites like Amazon, Goodreads, Google books, Flipkart etc. provide ratings and reviews on various books. Comments given by various readers can also be used to depict the polarity of a book. In book recommendation system various opinion mining techniques are used. There are various approaches used in book recommendation system like collaborative filtering, content based filtering etc. The process consists of data collection in which information regarding book like abstract of the book, review and comment are noted, then based on this information complete analysis is done and then is evaluated. In book recommendation system the books are recommended based on their interest also, the books can be fiction, non-fiction etc.
3. After recommending books the book recommendation system also take feedback from users or readers to improvise the recommendation system. The well- known book recommendation system LIBRA (Learning Intelligent Book Recommending Agent) uses content based filtering with text categorization.

**CHAPTER 9**

**CONCLUSION**

* The project has been developed successfully and the performance of the system has been found good. The system provides platform to view the books similar genere recommends atleast five books similar to the users input . The books are recommended to the users using matrix factorization technique. The system is efficient in maintaining and can be incorpated in any e commerce sites related to books and reduce the workload and help the buisness of various organizations.

**CHAPTER 10**

**FUTURE SCOPE**

* New features could be added to this project for making this project more productive, reusable and flexible which include preview of the recommend books,images of the books and review .

**CHAPTER 11**

**BIBILOGRAPHY**

[M sharma,Smann, "A survey of Recommender systems :Approaches and Limitations," 2013. [Online]. Available:](https://www.semanticscholar.org/paper/A-Survey-of-Recommender-Systems%3A-Approaches-and-Sharma-Mann/fa41dc4b60eccedf1c41e2ae488044827dd79384)

[[[PDF] A Survey of Recommender Systems: Approaches and Limitations | Semantic Scholar](https://www.semanticscholar.org/paper/A-Survey-of-Recommender-Systems%3A-Approaches-and-Sharma-Mann/fa41dc4b60eccedf1c41e2ae488044827dd79384)](https://www.semanticscholar.org/paper/A-Survey-of-Recommender-Systems%3A-Approaches-and-Sharma-Mann/fa41dc4b60eccedf1c41e2ae488044827dd79384)

<https://github.com/muniah/Book-Recommendation-System>

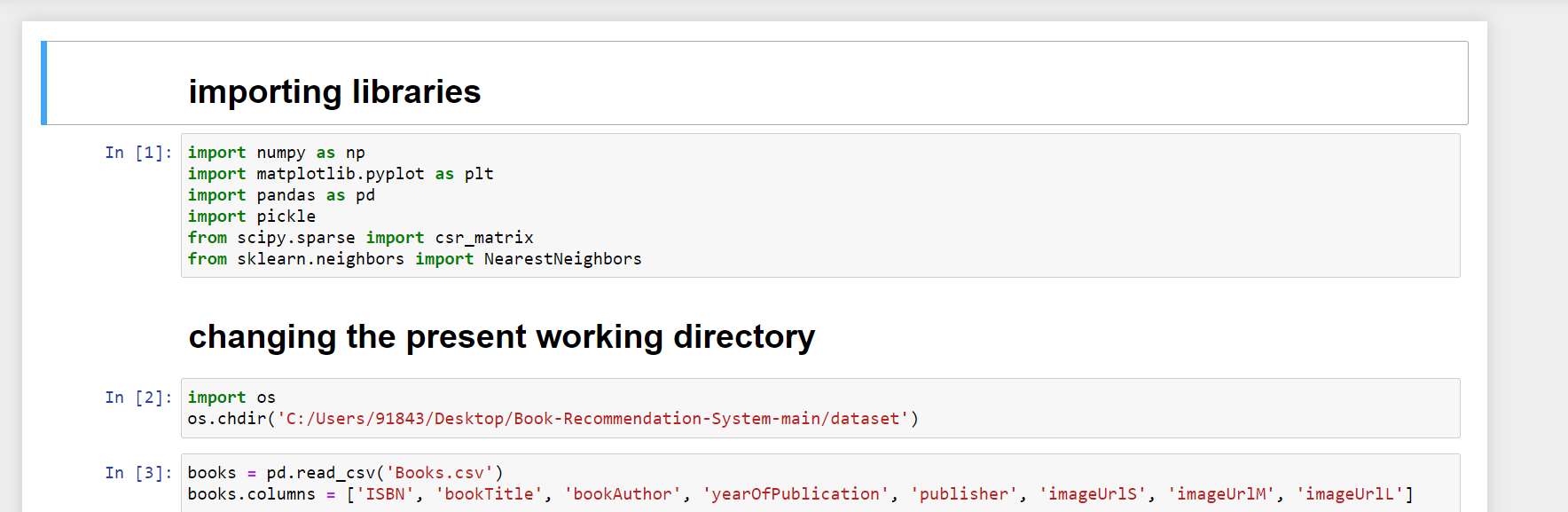
[[https://towardsdatascience.com/building-a-collaborative-filtering-recommender-system-with-tensorflow-82e63d27b420](https://github.com/muniah/Book-Recommendation-System)](https://towardsdatascience.com/building-a-collaborative-filtering-recommender-system-with-tensorflow-82e63d27b420)

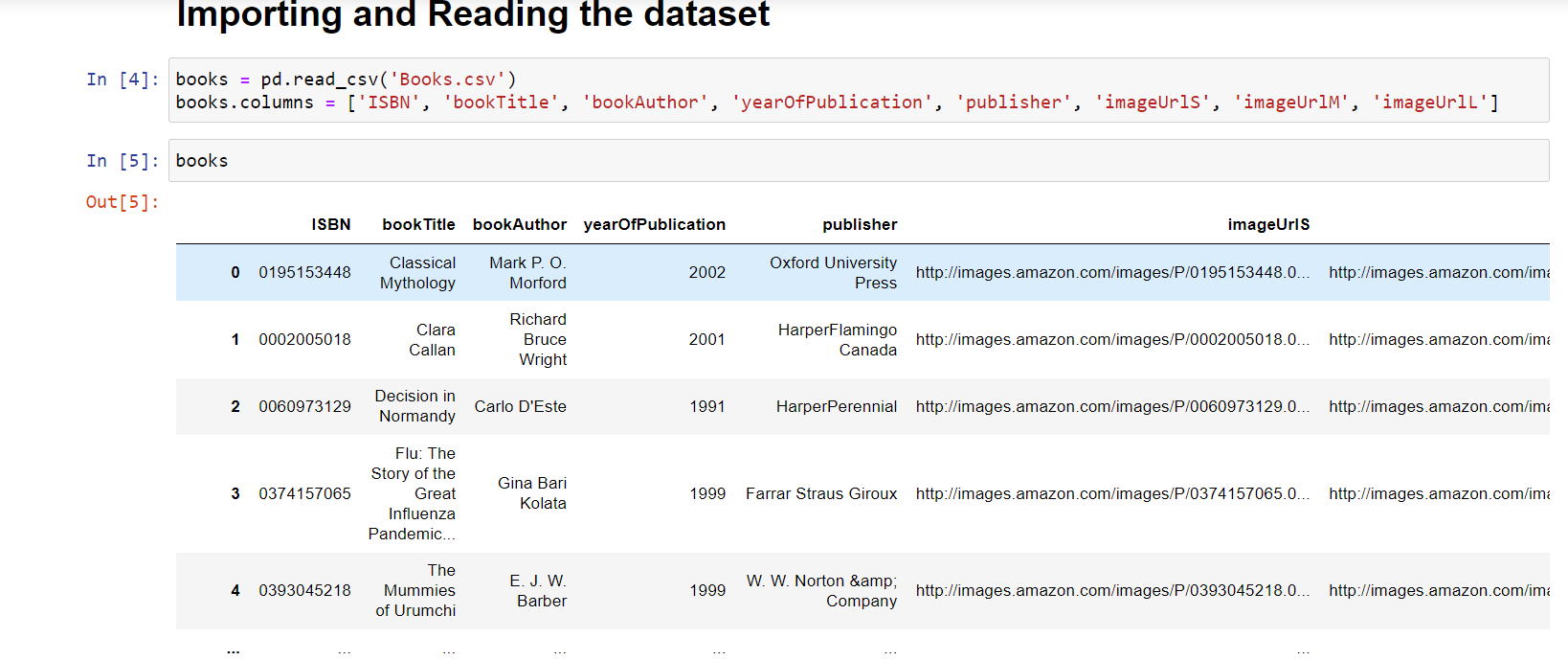
**CHAPTER 12**

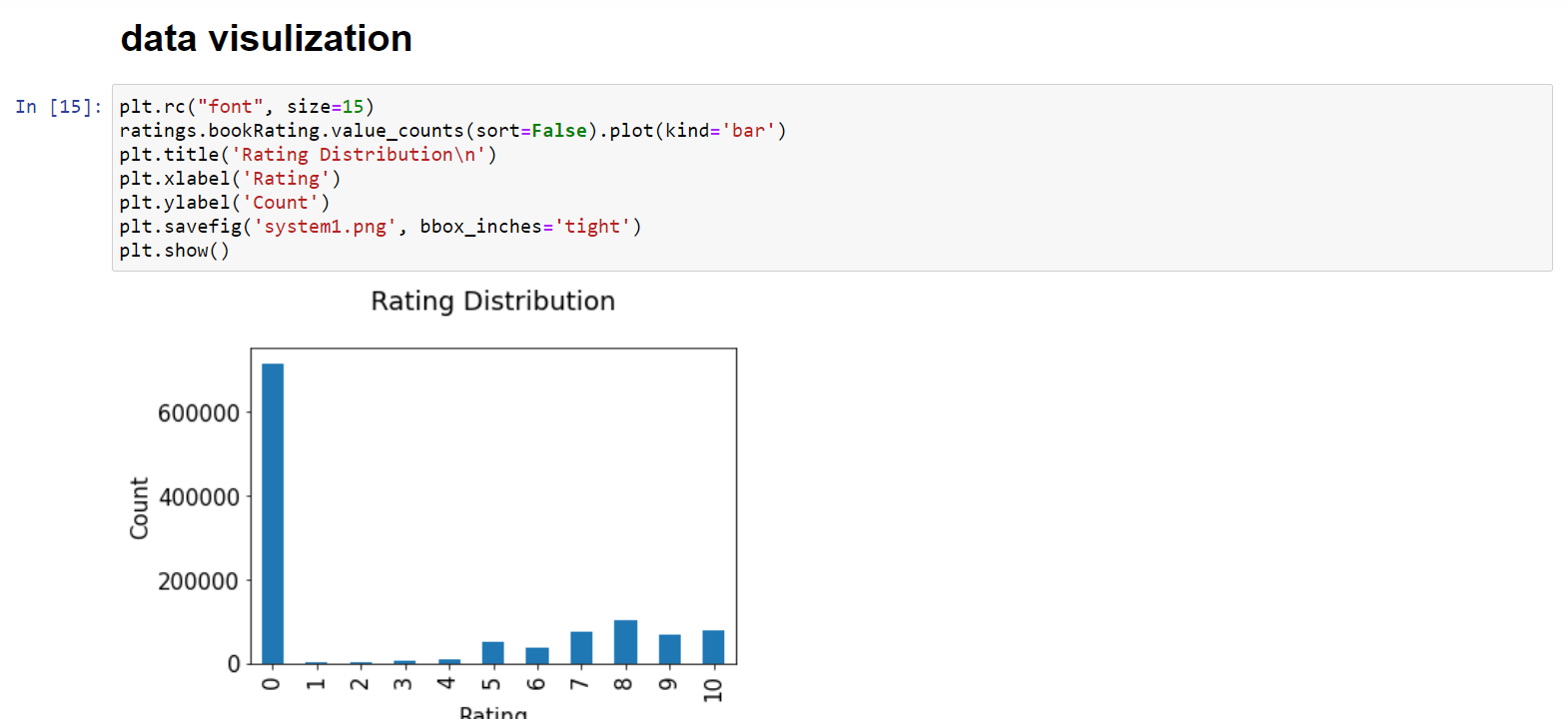
**APPENDIX**

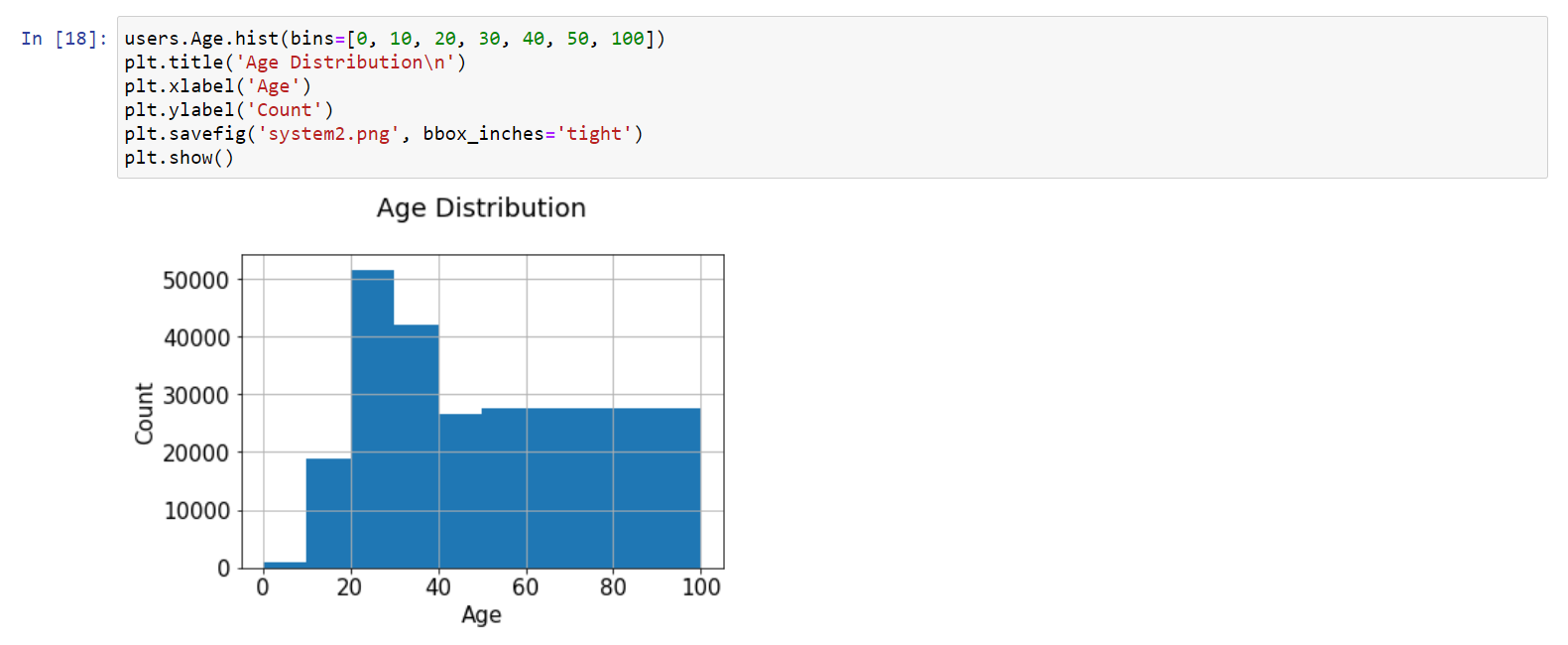
**12.1 SOURCE CODE**

**Book.ipynb model**

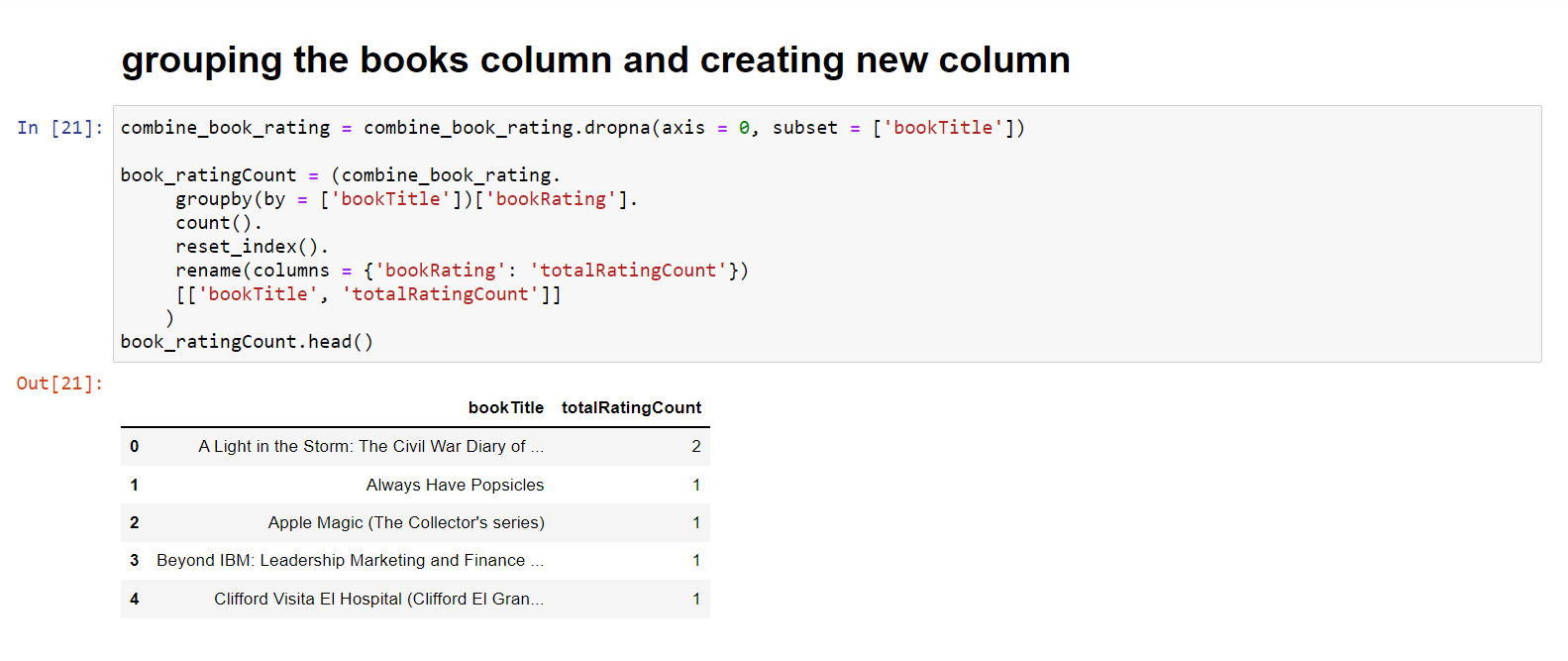


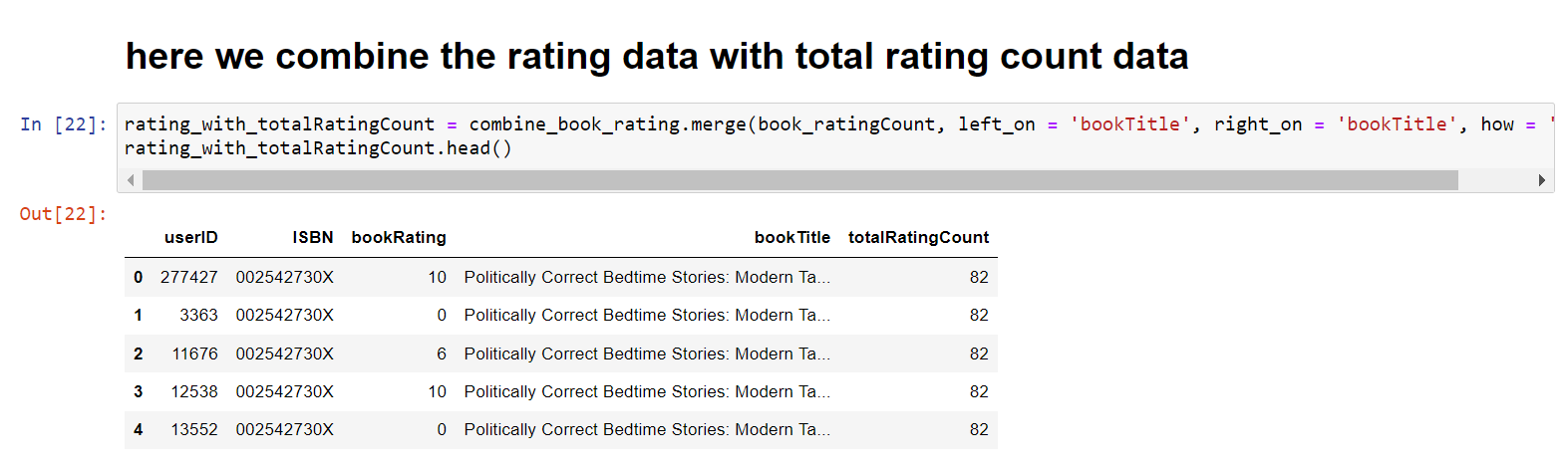


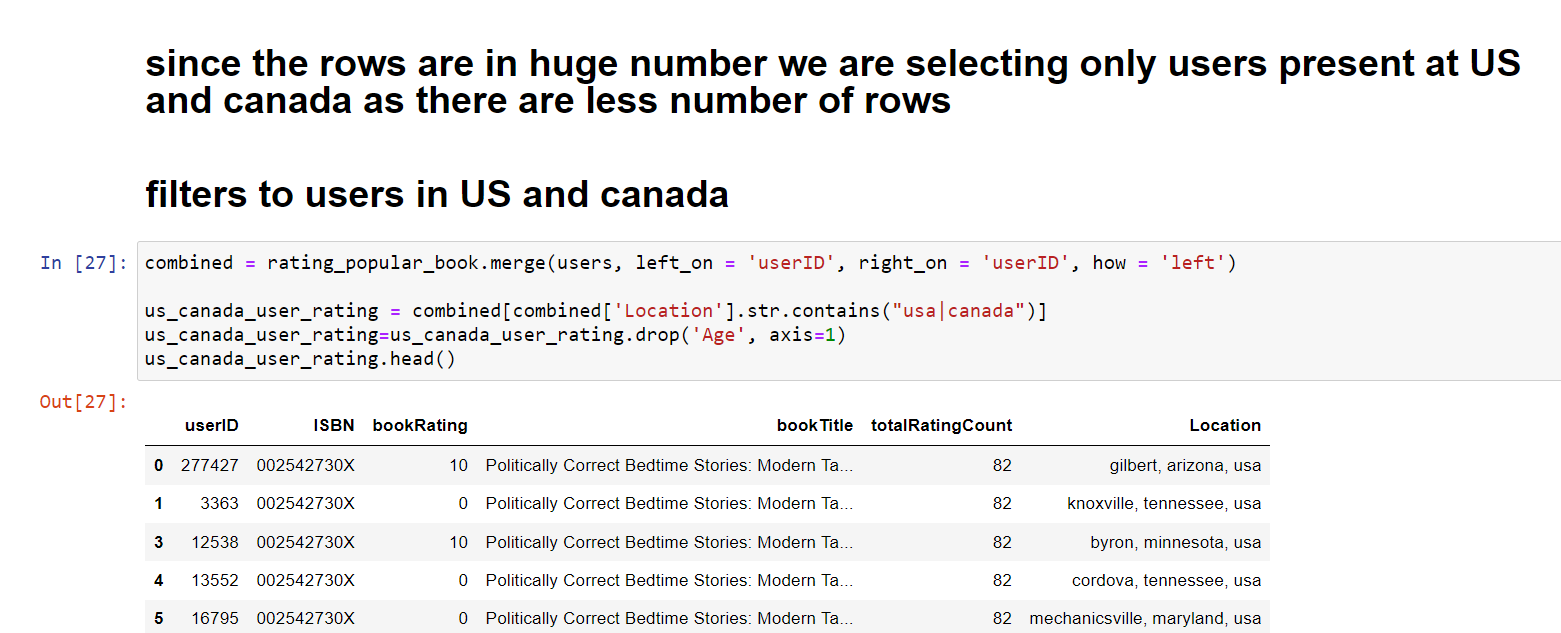




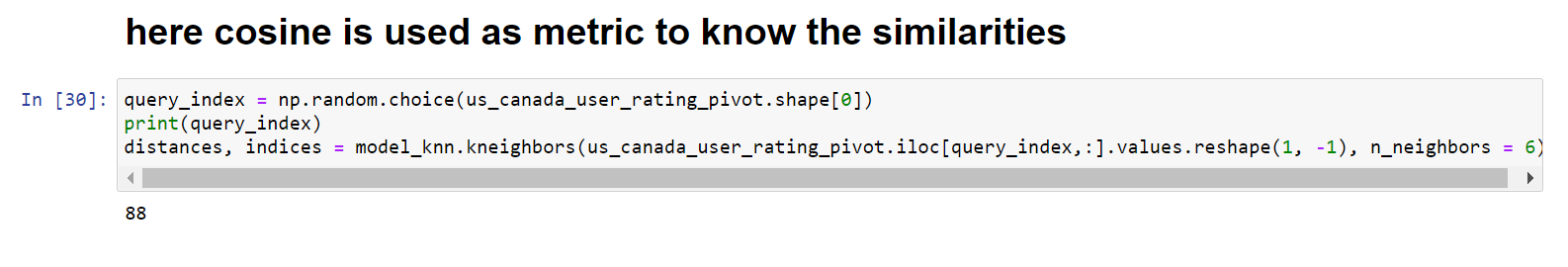


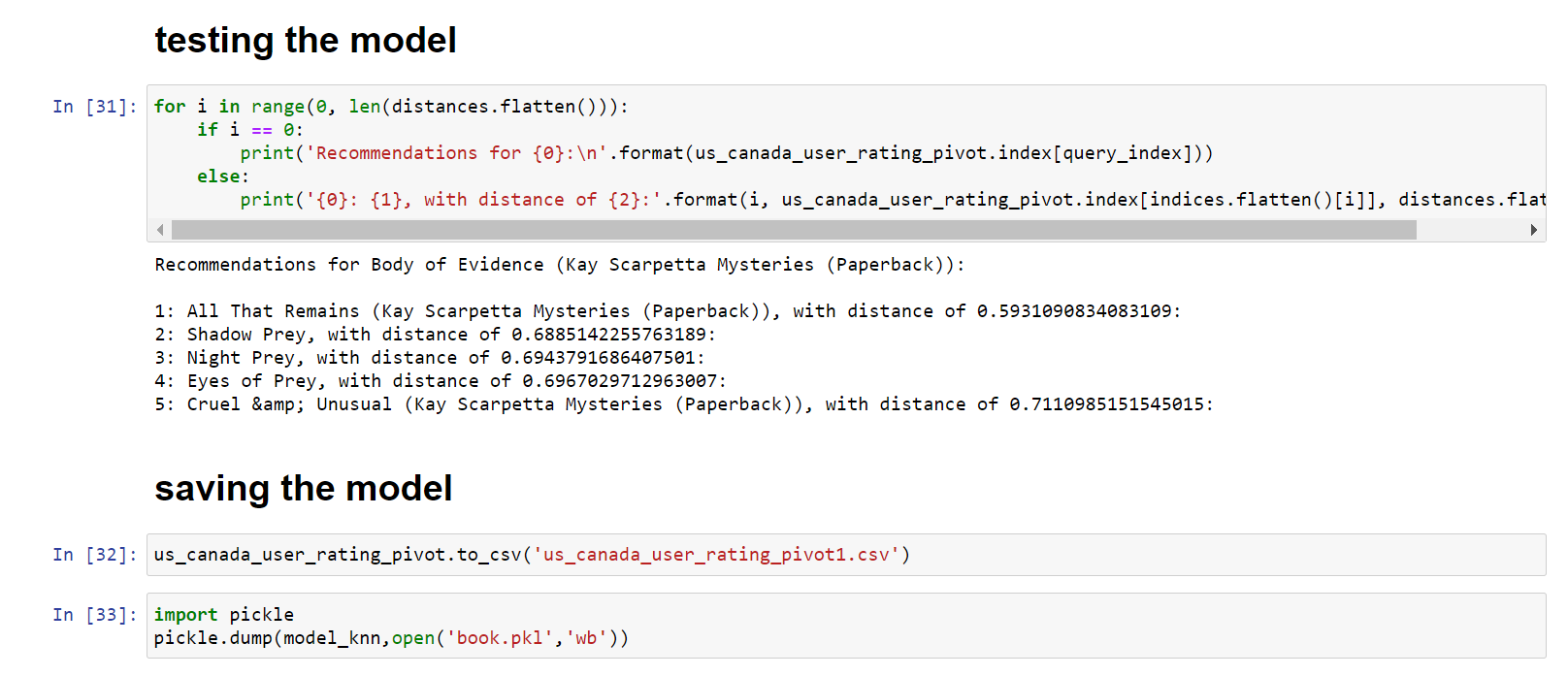




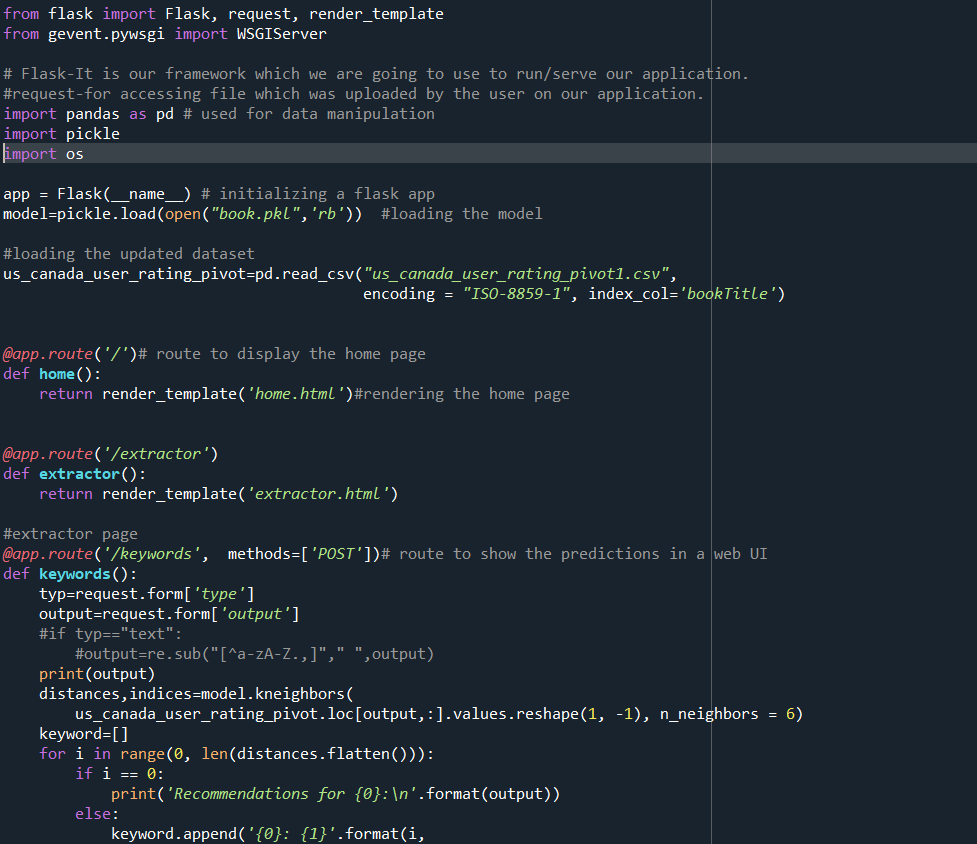


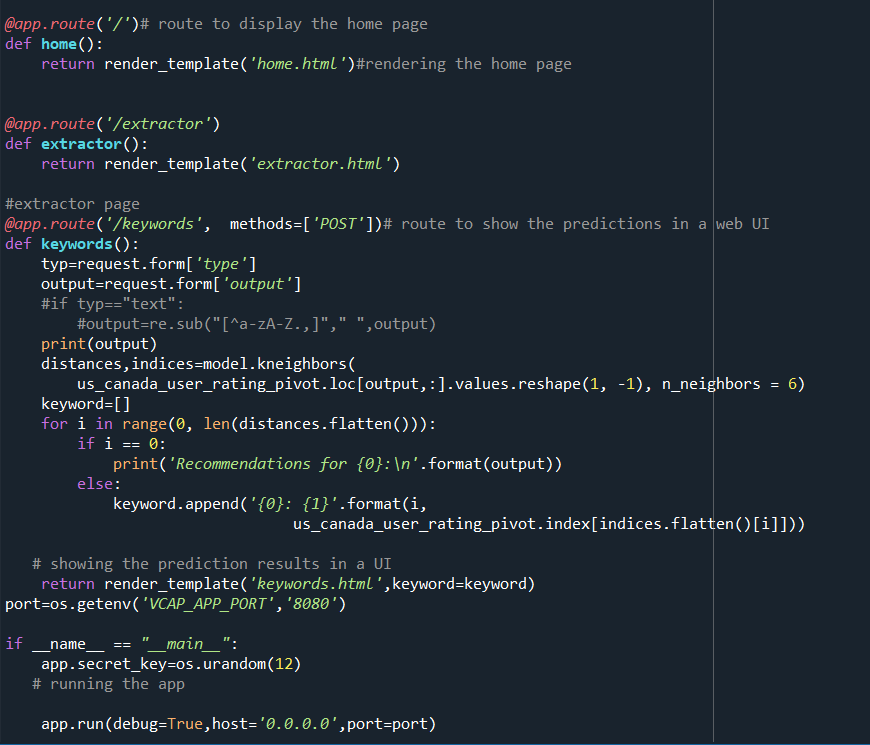






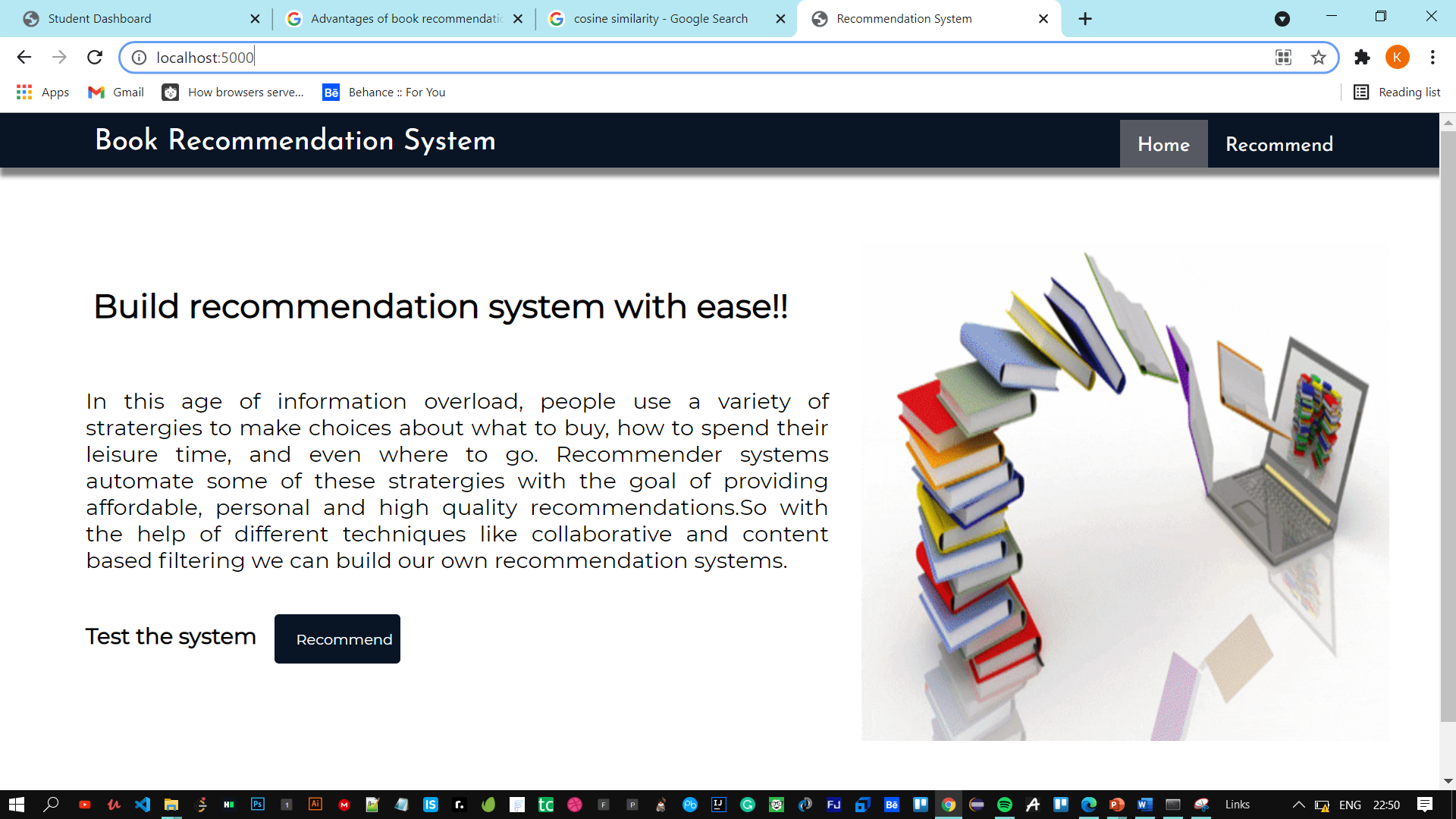
**FLASK CODE:**



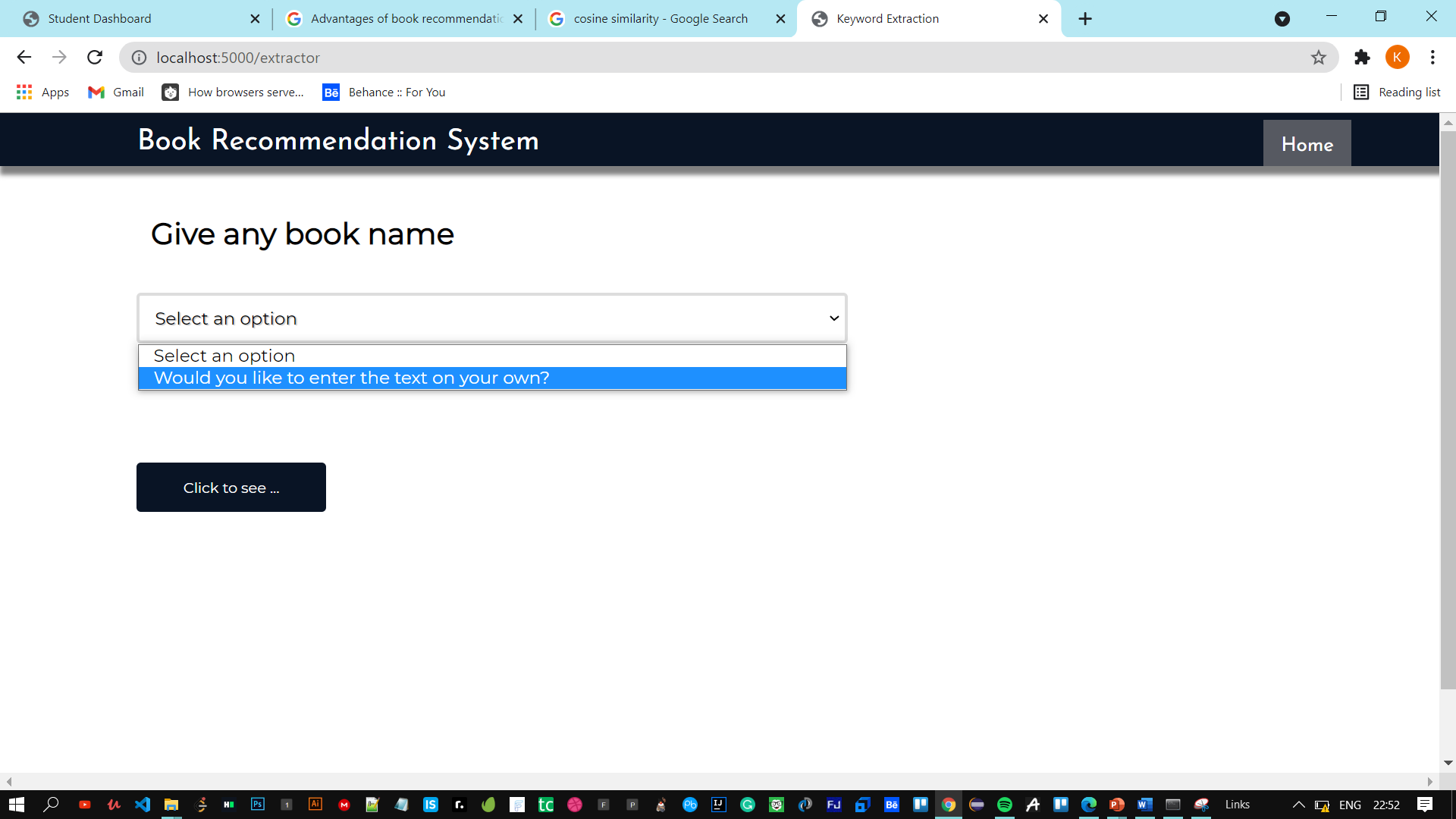


**12.2 UI OUTPUT SCREENSHOT :**

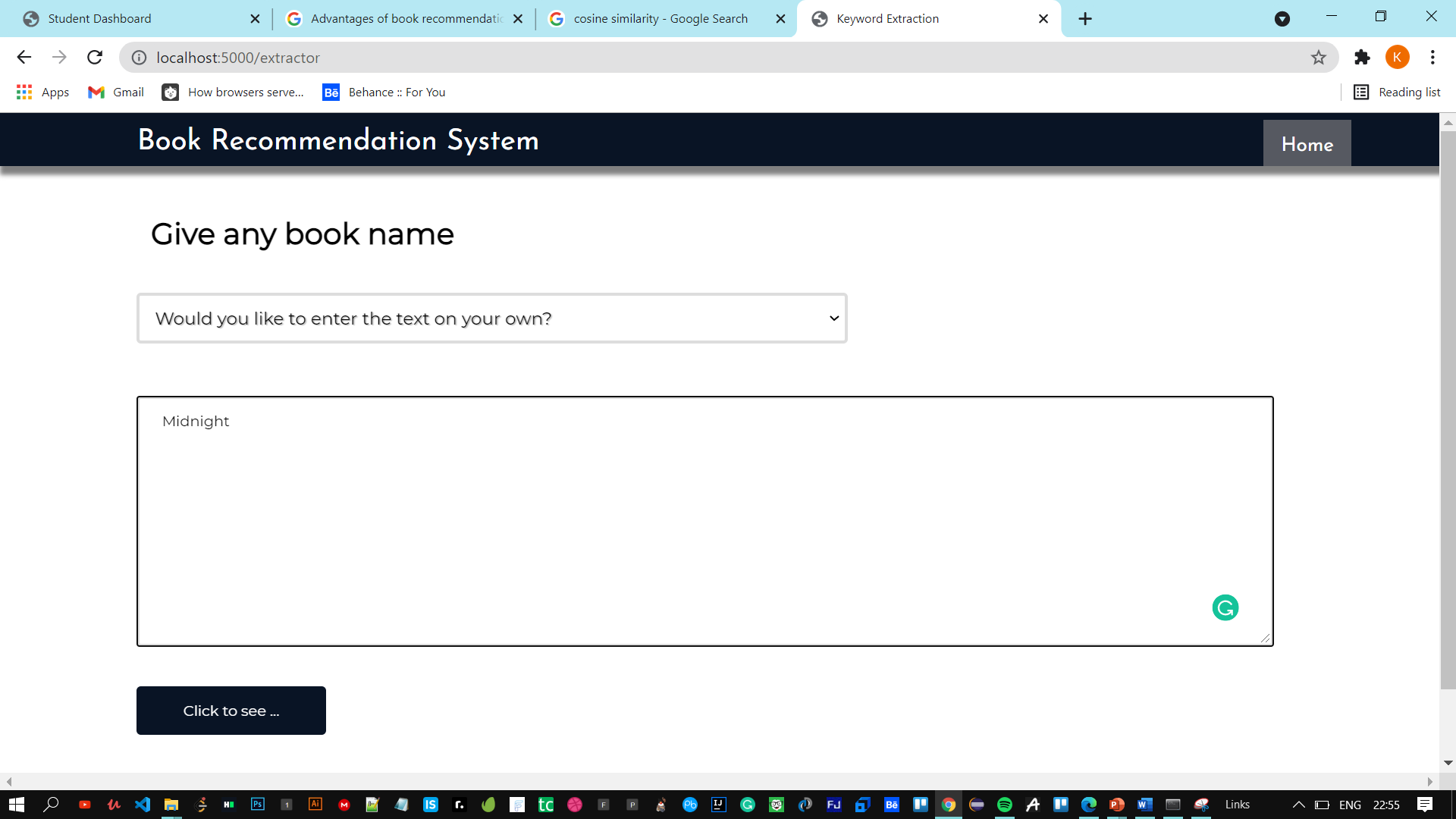
**HOME PAGE**



**USER INPUT PAGE :**



**USER INPUT IS BEING RECIEVED**



**OUTPUT: BASED ON USER INPUT BOOKS ARE RECOMMENDED**

