

**Dharamsinh Desai University, Nadiad**

Faculty of Technology, Department of Computer Engineering

B.Tech. CE Semester – VI

Subject: (CE-621) System Design Practice

Project Title:

**BOOK RECOMMENDATION SYSTEM**

**By**

**BALAR RIDDHI (CE-010) (17CEUOG007)**

**TRUPTI LANGALIA (CE-065) (17CEUOS080)**

**DRASHTI MENAPARA (CE-068) (18CEUOD007)**

**Guided by:**

Prof. Hariom A. Pandya

Assistant Professor of

Dept. of Comp. Engg.

Dharamsinh Desai University, Nadiad.



Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

**CERTIFICATE**

This is to certify that System Design Practice project entitled “Book Recommendation System” is the bona fide report of work carried out by

**BALAR RIDDHI (CE-010) (17CEUOG007)**

**TRUPTI LANGALIA (CE-065) (17CEUOS080)**

**DRASHTI MENAPARA (CE-068) (18CEUOD007)**

Of Department of Computer Engineering, Semester VI, academic year 2019-20, under your supervision and guidance.

|  |  |
| --- | --- |
| Guide | HOD |
|  |  |
| **Prof. Hariom A. Pandya** | **Dr. C. K. Bhensdadia** |
| Assistant Professor of | Head of the Department of |
| Department of Computer | Department of Computer |
| Engineering, Dharmsinh Desai University, Nadiad. | Engineering, Dharmsinh Desai University, Nadiad. |

Table of Contents

1. [Abstract 4](#_Toc38613864)

2. [Introduction 5](#_Toc38613865)

2.1 [Brief Introduction 5](#_Toc38613866)

2.2 Tools/Technologies Used [5](#_Toc38613867)

3. Software Requirement Specification [6](#_Toc38613869)

3.1 [Product Scope 6](#_Toc38613865)

3.2 [Types of user 6](#_Toc38613866)

3.3 [System Functional Requirements 6](#_Toc38613865)

3.4 [Other Non-Functional Requirements 8](#_Toc38613866)

4. Design [9](#_Toc38613870)

4.1 [Use-Case Diagram 9](#_Toc38613865)

4.2 [Sequence Diagram 10](#_Toc38613866)

4.3 [Data Dictionary 11](#_Toc38613866)

5. Implementation Details [13](#_Toc38613871)

5.1. [Implementation of Collaborative Machine Learning Model 13](#_Toc38613872)

5.2. [Implementation of content based filtering 16](#_Toc38613877)

5.3. [Implementation of web app 17](#_Toc38613877)

6. Testing [18](#_Toc38613882)

7. Screenshots [20](#_Toc38613882)

8. [Conclusion 23](#_Toc38613883)

9. [Limitations and future extensions 24](#_Toc38613884)

10. [Bibliography 25](#_Toc38613885)

1. Abstract

Book Recommendation system is a recommender system built for book lovers. Using your profile, our system uses Machine Learning methods to provide you with highly personalized book recommendations.

Recommender systems are at the forefront of the ways in which content-serving websites like Facebook, Amazon, Spotify, etc. interact with their users. Given this climate, it is paramount that websites aim to serve the best personalized content possible. We are using a hybrid recommender system in order to provide recommendations from the dataset given by Goodreads users (ratings and item features). We also provide more 'traditional' recommendations that only use the book's features.

2. Introduction

**2.1 Brief Introduction**

We use a hybrid recommender system to power our recommendations. Hybrid systems are the combination of two other types of recommender systems: content-based filtering and collaborative filtering. Content-based filtering is a method of recommending items by the similarity of the said items. That is, if I like the first book of the Lord of the Rings, and if the second book is similar to the first, it can recommend me the second book. Collaborative filtering is a method by which user ratings are used in order to determine user or item similarities. If there is a high correlation of users rating the first Lord of the Rings book and the second Lord of the Rings book, then they are deemed to be similar. Our hybrid system uses both of these approaches. Our item similarities are a combination of user ratings and features derived from books themselves.

**2.2 Tools/Technologies Used**

**Technologies**

Django framework

**Tools**

Visual Studio Code

Jupyter Notebook

3. Software Requirement Specification

**3.1 Product Scope**

Our main goal is to provide best recommendation to user.

**3.2 Types of User**

1. End user

**3.3 System Functional Requirements**

**R1: End user**

R-1.1: Add area of interest

Description: Using this data machine can automatically suggest books accordingly user’s interest.

Input: Different Genre

Output: Genre added to area of interest.

R-1.2: Search book

Description: User can search book using different parameters.

R-1.2.1: Search book with book name.

Input: Book name

Output: Book details of the given book

R-1.2.2: Search book with author name.

Input: Author name

Output: Book details of every books written by that author

R-1.2.3: Search book with genre name.

Input: Genre

Output: Book details of every books fall in the given genre.

R-1.3: Get suggested books.

Description: Machine can automatic suggest book using different algorithms of machine learning.

Input: Area of interest

Output: All the suggested books

**R2: Registration**

Description: User can register

Input: Enter user details like email, password, name etc.

Output: Register successfully

**R3: Login**

Description: User can login

Input: Enter login details

Output: Login successfully and redirect to home page

**R4: Logout**

Description: User can log out after finishing work.

Input: User selection.

Output: Redirection to Login page.

**3.4 Other Non-functional Requirements**

**1. Performance**

The system must be interactive and the delays involved must be less. So in every action-response of the system, there are no immediate delays. In case of opening App components, of popping error messages and saving the settings or sessions there is delay much below 3 seconds.

**2. Safety**

User details should be securely stored to the server. The main security concern is for user account hence proper login mechanism should be used to avoid hacking.

**3. Reliability**

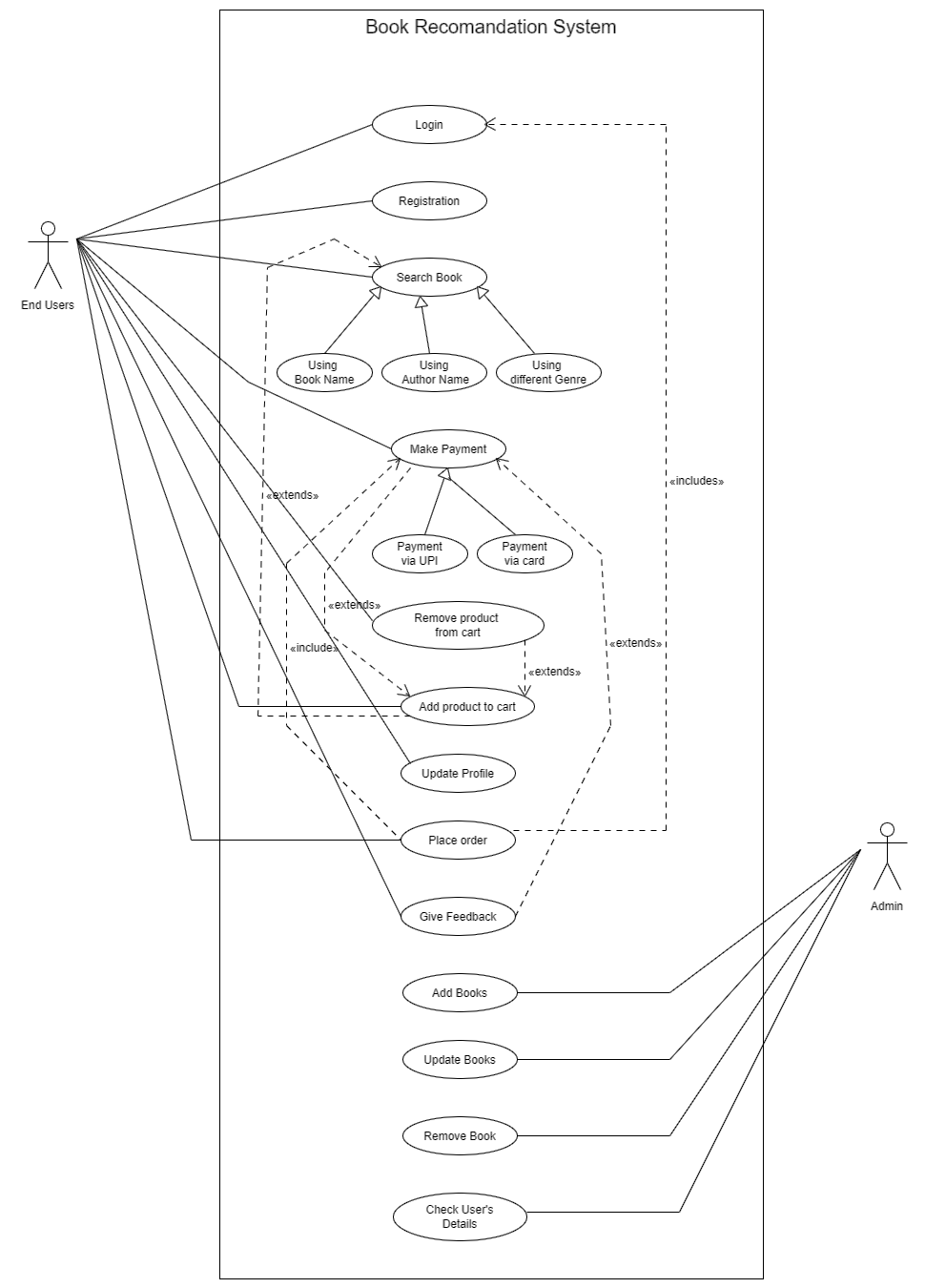
As the system provides the right tools for discussion, problem solving it must be made sure that the system is reliable in its operations and for securing the sensitive details.

**4. Database**

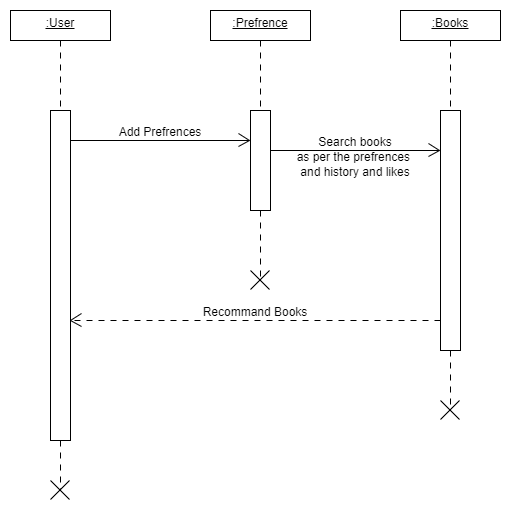
System requires to access users data fast to maintain the performance.

4. Design

**4.1: Use case diagram:**



**4.2: Sequence Diagram:**



**4.3: Data Dictionary:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| User | | | | | | | | |
| Sr. No | Field Name | Data Type | Width | Required | Unique | PK/FK | Reference  table | Description |
| 1. | User Id | Int | 20 | yes | yes | PK |  |  |
| 2. | User name | Varchar | 50 | yes | no |  |  |  |
| 3. | Password | Varchar | 50 | yes | Yes |  |  |  |
| 4. | Email Id | Varchar | 50 | no | no |  |  |  |
| 5. | Area of interest | Varchar | 30 | no | no |  |  | Multiple values allowed |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interest | | | | | | | | |
| Sr. No | Field Name | Data Type | Width | Required | Unique | PK/FK | Reference  table | Description |
| 1. | BookId | Int | 20 | yes | no |  |  |  |
| 2. | Booktitle | Varchar | 50 | yes | no |  |  |  |
| 3. | Genre | Varchar | 50 | yes | no |  |  |  |
| 4. | Author | Varchar | 50 | yes | no |  |  |  |
| 5. | Description | Varchar | 500 | no | no |  |  | Multiple values allowed |
| 6. | ImageURL | Varchar | 100 | yes | no |  |  |  |
| 7. | UserId | int | 20 | yes | No |  |  |  |

Dataset: <https://github.com/zygmuntz/goodbooks-10k>

**Dataset’s fields:**

* book\_authors

Description: This field include name of the author

Usage: We are using this field in searching functionality. In that user can search books using author name.

* book\_desc

Description: This field gives description of books

Usage: We are using this field to recommend people which kind of book they are interested in.

We are using Machine learning algorithm to find semantic similarity to find similar books based on description.

* book\_format

Description: This field tells book is in hard cover or in paper bag.

* book\_rating

Description: This field shows average rating of book.

Usage: Using book\_rating field we can recommend book to the people based on public review.

* book\_rating\_count

Description: This field counts total number of rating count i.e. how many people had gave rating to book.

* book\_title

Description: This field shows title of the book.

Usage: We are using this field in searching functionality. In that user can search books using author name.

* genres

Description: This field shows genres of the book. There can be many genres related to one book i.e. Catching fire, this book is contains Young adult and science fiction etc. genres.

Usage: We are using this field in searching functionality. In that user can search books using author name. Also using this field we can predict books which are having same genres.

* image\_url

Description: This field contains URL of book cover page which we are using while displaying the books in UI.

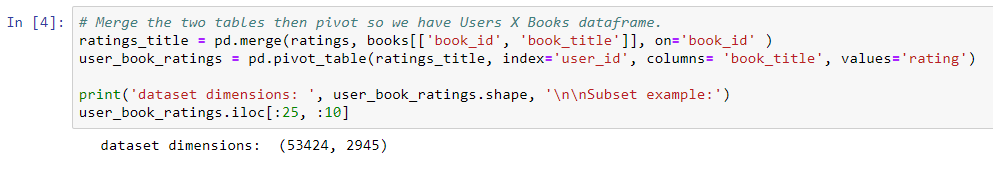
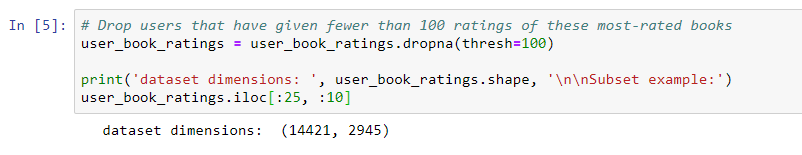
This is good way to display image using URL rather than storing all the photos into database. All the URLs are stored online so you must have internet

Connection to access it.

5. Implementation Details

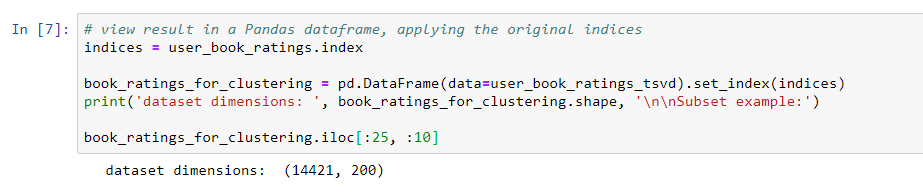
**5.1: Implementation of Collaborative Machine Learning Model:**

After loading and analyzing the data set we have performed Data preprocessing to reduce the data size, to merge the required data and to drop unnecessary columns and rows.

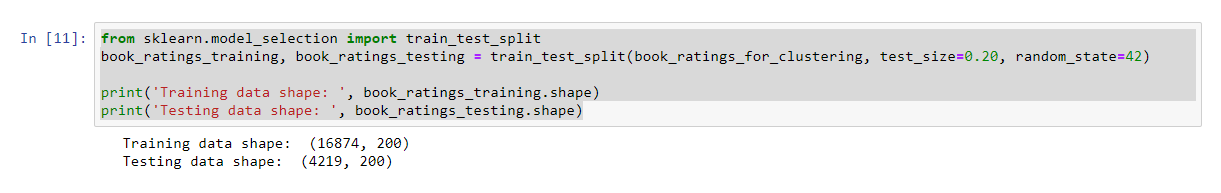
 

To further reduce dimensionality, we had applied truncated SVD, which is available for sparse matrices. We've determined that if we use 200 features, these features will explain about 45 percent of the variance in the data.

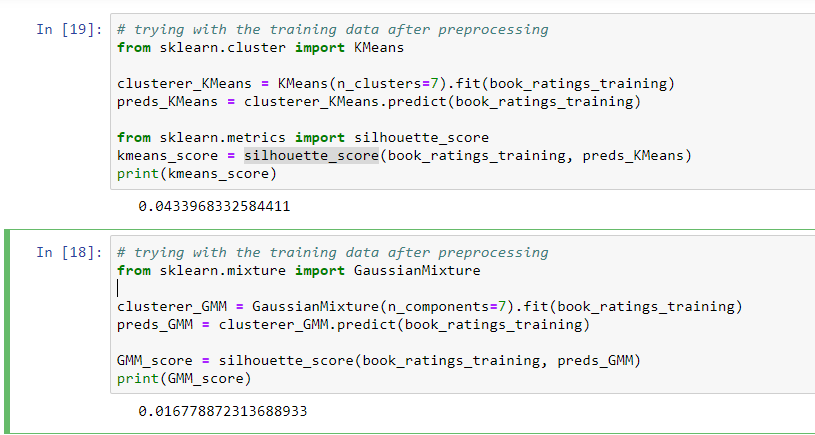
So we have scored the remaining 14,421 users on 200 features, each of which represents some combination of opinions about various books.



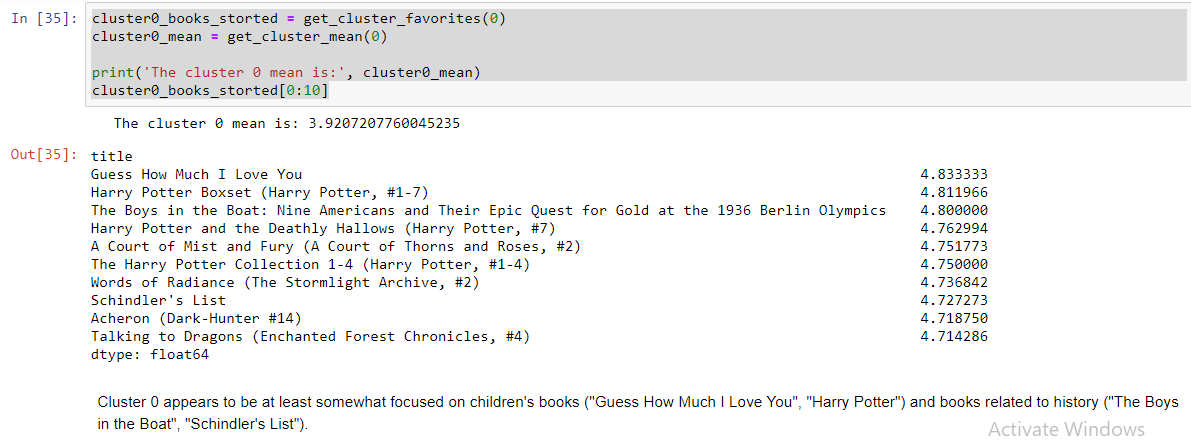
Then we separated out 20 percent of the users to create a test set.



Then we have performed clustering on training set. We have applied K-means clustering and Gaussian mixture modelling to cluster the users and reach the best silhouette score. After applying theses 2 methods with various cluster counts we have got the conclusion that K-means clustering with the cluster count of 7 gives the best silhouette score.



Then we have analyzed each cluster.

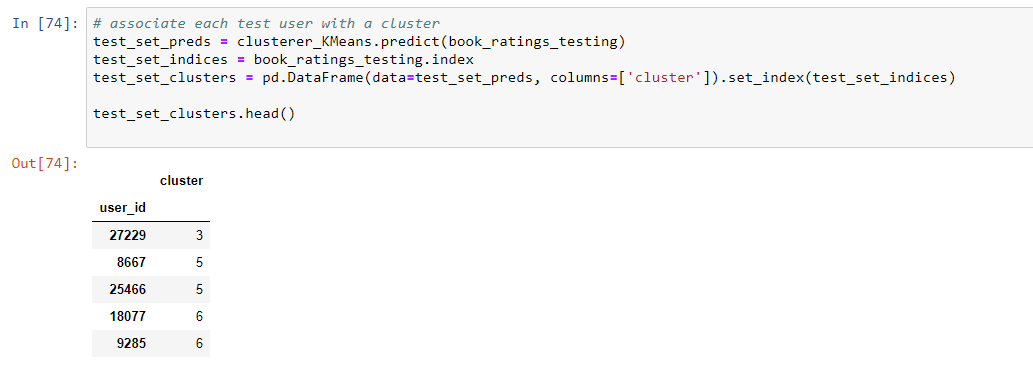


We have performed our clustering on a dataset that included 200 composite features. It is difficult to create a visualization that effectively illustrates all of these features. Therefore, we have selected the two top features, which played the most significant role in the clustering, and created a scatterplot that illustrates the clusters across those features.

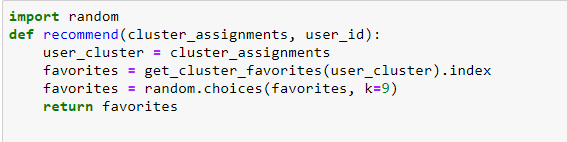


This indicates that the clustering model was effective in grouping users based on their values for composite features 0 and 1. The scatterplot also indicates that a few users had outlier values for composite feature 0.

Then we have tested our testing dataset with the model and found the the cluster associated with each user.

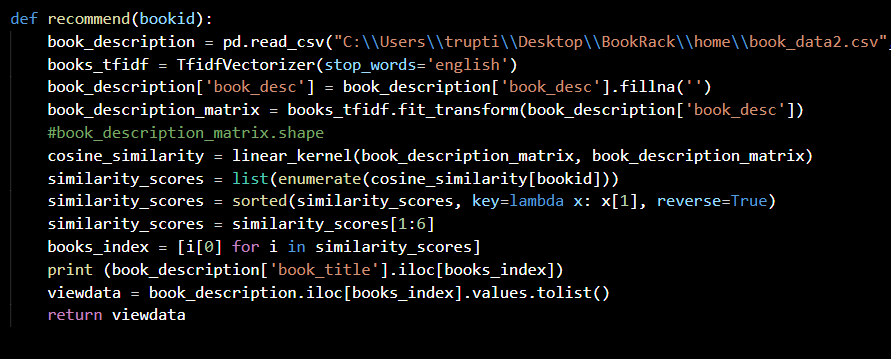


After associating each user with clusters, we have determined the cluster favorites to recommend the book to the user of particular cluster.



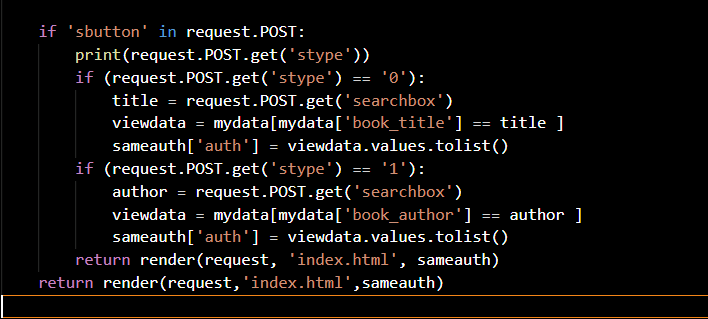
**5.2: Implementation of content based filtering:**

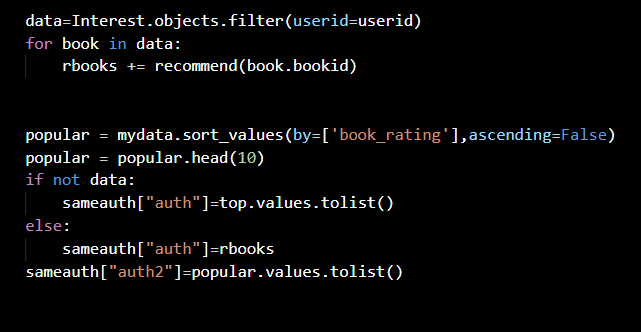
We have used cosine similarities on the book description to determine the similar book to a given book.



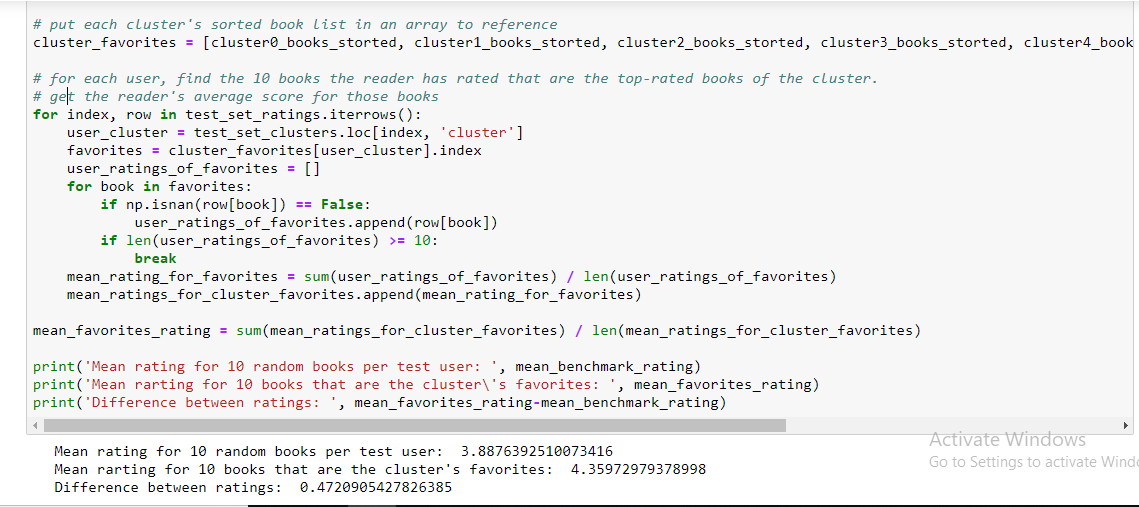
**5.3: Implementation of web app:**

With use of above 2 models we are taking user’s interest in book and showing them recommended books, popular books and searched books.





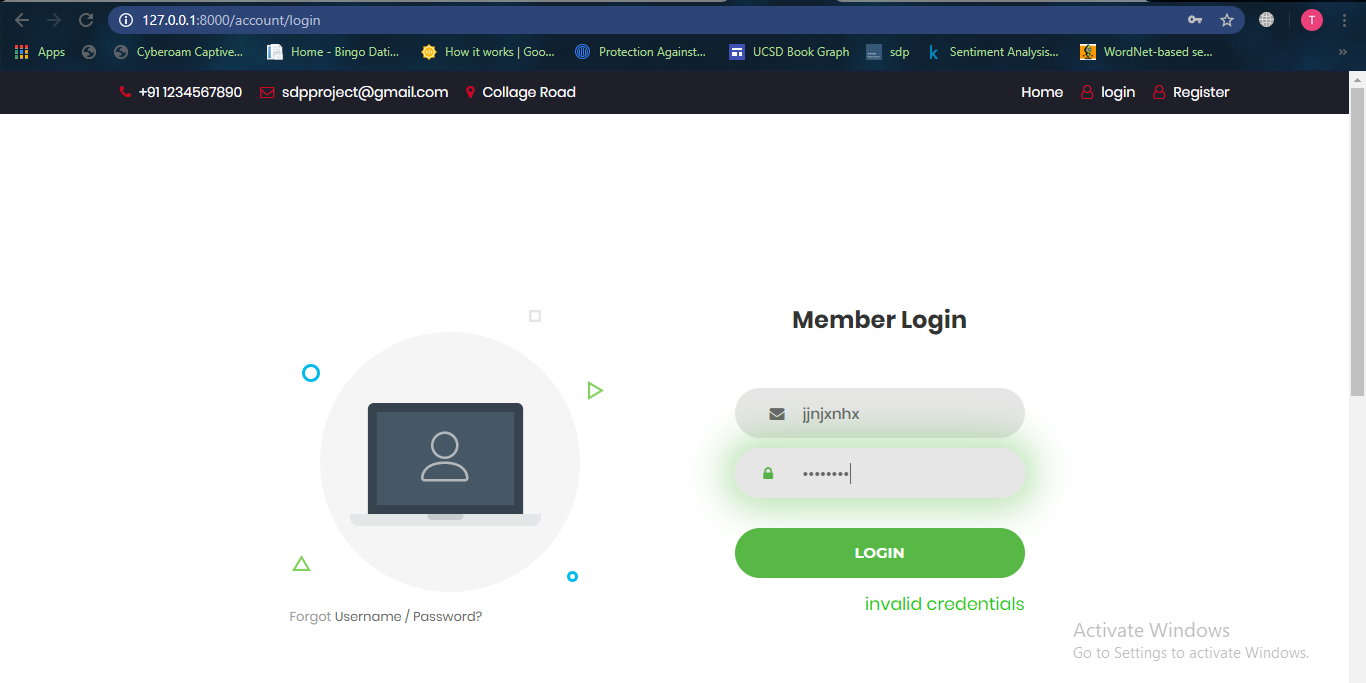
6. Testing



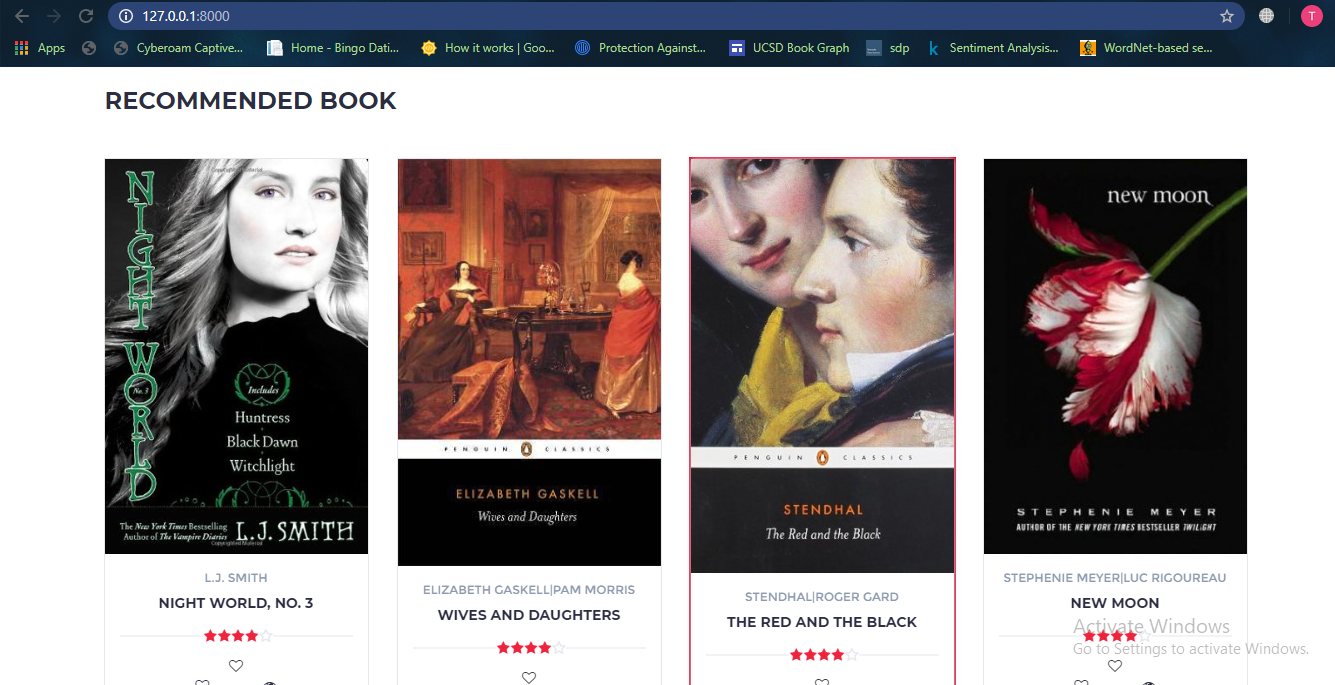
The users in the test set, on average, rated their clusters' favorite books higher than a random set of 10 books by 0.47 stars, or nearly half a star.

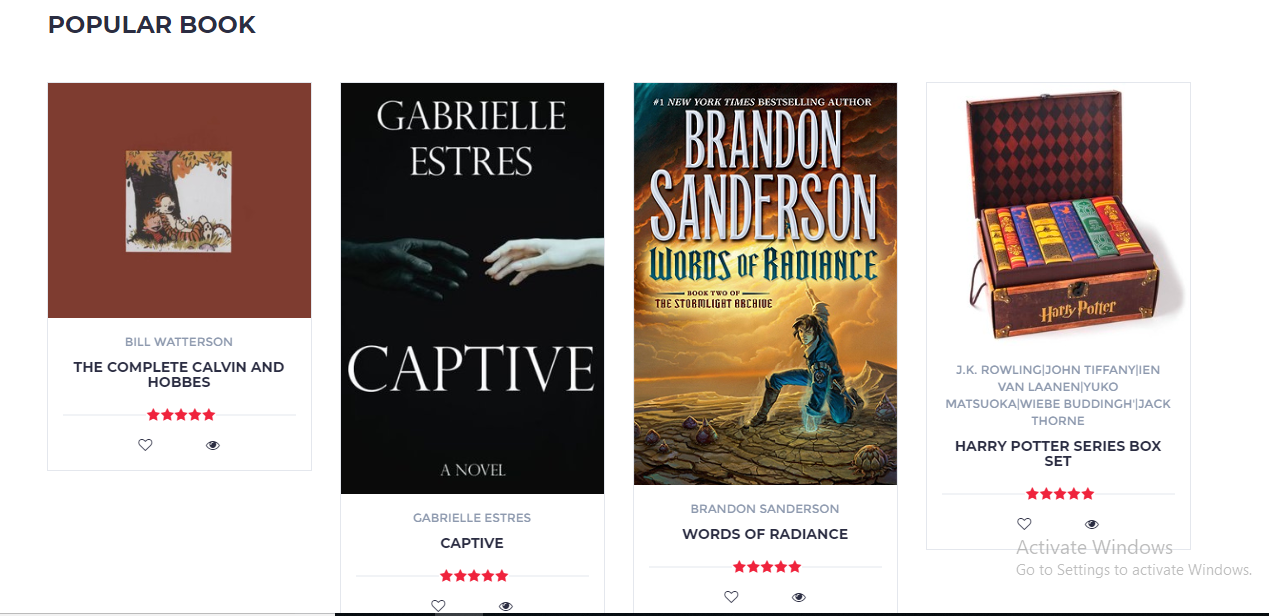
This provides some evidence that the model does a better job of recommending books than if it simply picked books at random. However, it is not clear that the clustering itself provided this benefit, or if the benefit came from simply recommending books that where highly rated overall across clusters.

We have also performed test cases to our web app.

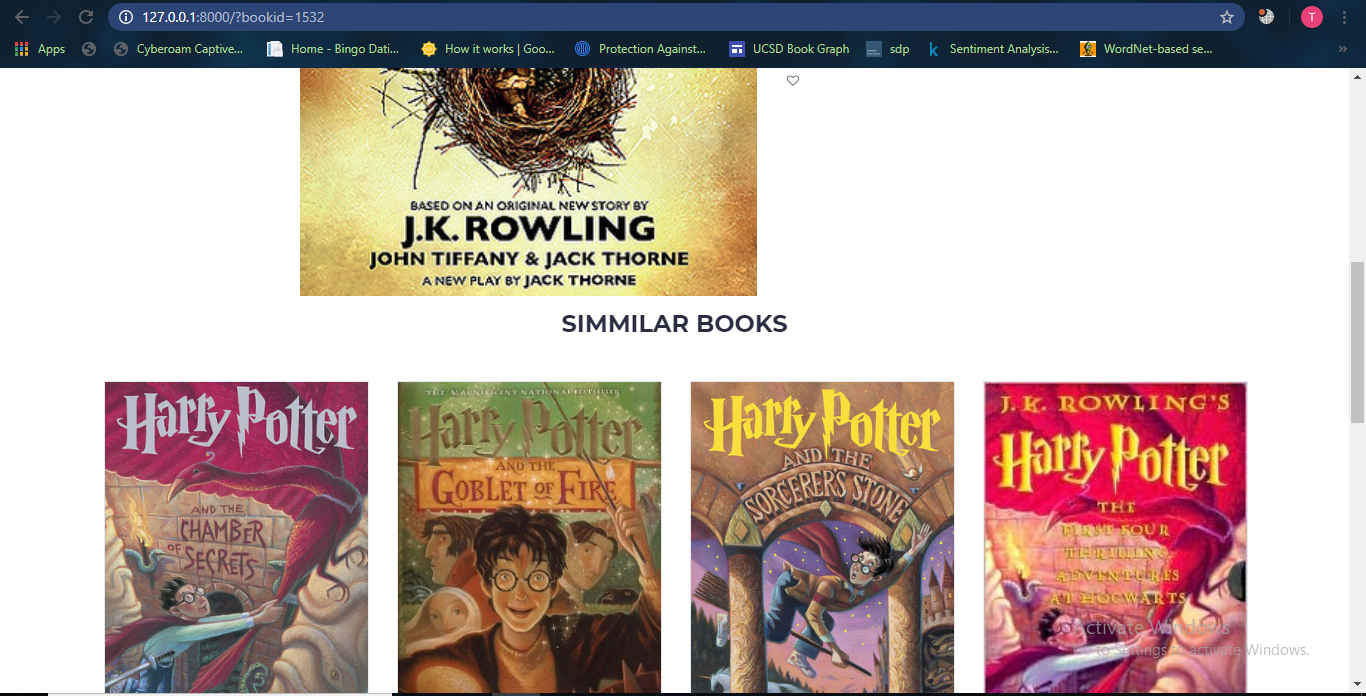


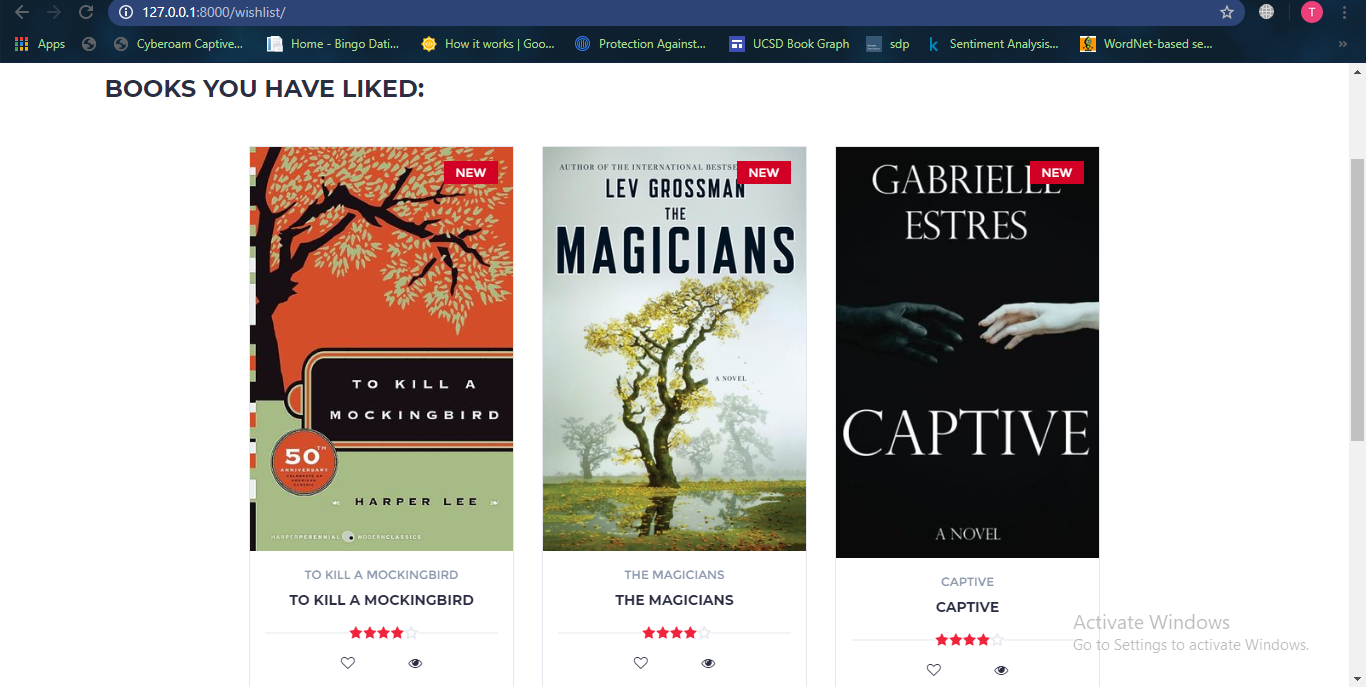
7. Screenshots











8. Conclusion

The functionalities implemented in system after understanding all the system modules according to the requirements. Functionalities that are successfully implemented in the system are:

* Book recommendation
* Search book
* Ratings

After the implementation and coding of system comprehensive testing was performed on the system to determine the errors and possible flaws in the system.

We were able to train efficient models that had high accuracy. The model performed well on random inputs outside the dataset as well. We tried pursuing different approaches to solve the same problem,

9. Limitations and Future Enhancements

The System has adequate scope for modification in future if it is necessary.

Development and launching of Mobile app and refining existing services and adding more service, System security, data security and reliability are the main feature.

The API for the shopping and payment gateway can be added.

In the existing system there are only some selected categories, so as an extension to the site we can add more categories as compared to existing site. Also we can add admin side with some functionalities like books management, User management etc.

10. Reference / Bibliography

Following links and websites were referred during the development of this project.

<https://github.com/zygmuntz/goodbooks-10k>

<https://towardsdatascience.com>

<https://docs.djangoproject.com/en/3.0/>

<http://stackoverflow.com/>

<https://scikit-learn.org/stable/>