

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION TECHNOLOGY AND COMMUNICATION



Introduction to Data Science
Report Project

Project name: Book Recommendation

Lecturer: Tran Viet Trung

Group: 08

Student names: Đào Quang Dương- 20194747

Nguyễn Thu Hương - 20194775

Phạm Thị Nhung - 20194814

Vũ Thị Phụng - 20194820

Thần Mạnh Thắng - 20194841

Hà Nội, 12/2023

Book recommendation

Abstract. In recent years, in online bookstores with extensive collections, choosing books can be challenging. So we propose a book recommendation system based on their preferences, reading history, and other relevant factors. Readers don't have to sift through thousands of books to find suitable ones. The recommendation system helps them save time and effort.

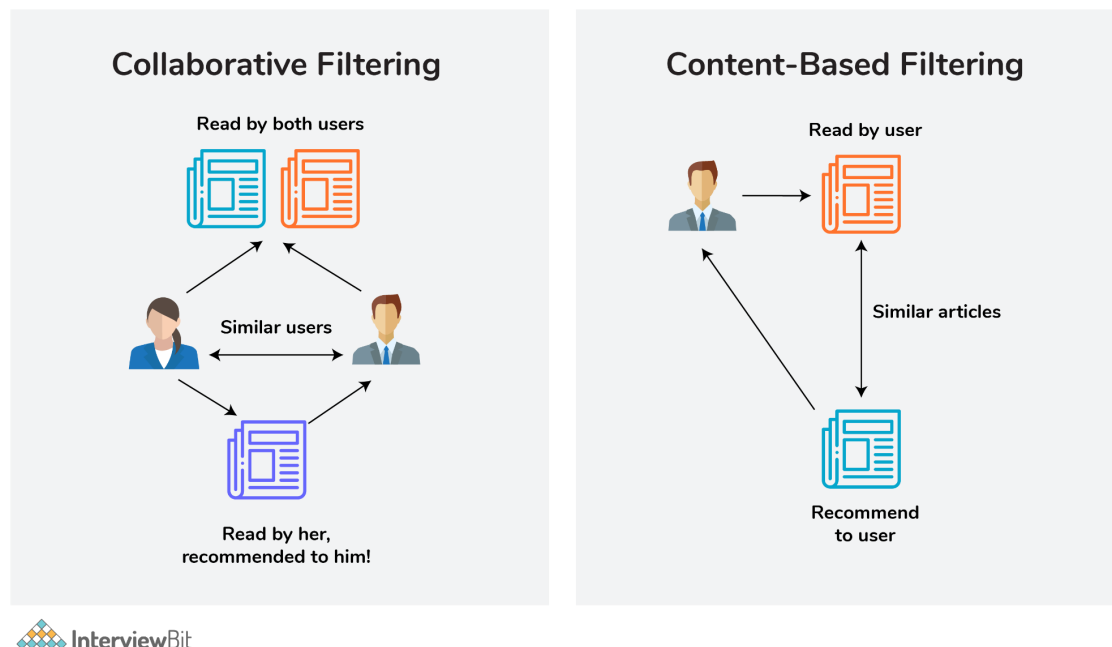
The book recommendation system is an intelligent application designed to provide personalized reading suggestions based on the user's interests and preferences. Built on artificial intelligence technology, this system utilizes data analysis algorithms to understand readers and generate precise recommendations.

With its self-learning capabilities and continuous updates, the book recommendation system not only tracks current reading preferences but also anticipates potential new interests. By combining information about genres, authors, and previous reader reviews, it creates an interactive and engaging reading experience.

Featuring a user-friendly interface, the system not only helps readers discover new works but also enhances interaction and the sharing of reading experiences through social media. Simultaneously, the system's ability to interact with users through feedback contributes to its ongoing improvement and better responsiveness to the diverse needs of the reader community. The book recommendation system is a reliable partner, offering a unique and enjoyable reading experience for all book enthusiasts.

1. Introduction

It is becoming a very difficult task for the users to select the appropriate books for a specific topic as there are a lot of choices available. There is a need for a system which takes user preferences into consideration while searching and recommending online books to the user. So the objective of this research work is to design an application that recommends books based on users ratings. The system being proposed uses machine learning algorithm like Content-based recommendation, collaborative filtering [CF] that first construct the user-item interaction matrix, then construct vector matrix using cosine similarity measure from user-item interaction matrix and then find the similarity between the books using vector matrix and recommend the top n books similar to the book given by the user as input to the algorithm. The results indicate that recommendation performance is better when both average ratings and cosine vector similarity measure is used as compared to the existing systems.



2. Algorithm

2.1. Content-based recommendation:

Content-based recommendation is an important approach in recommender systems. A content-based recommendation system recommends items to a user by taking the similarity of items. This recommender system recommends products or items based on the information of items. It identifies the similarity between the products based on their description.

The first task is converting each book's information into vectors using TF-IDF

Tf-term frequency: used to estimate the frequency of a word's appearance in the text.

$TF(t, d) = (\text{number of times word } t \text{ appears in text } d) / (\text{total number of words in text } d)$

IDF- Inverse Document Frequency: used to estimate the importance of that word.

$IDF(t, D) = \log_e(\text{Total number of documents in sample set } D / \text{Number of documents containing word } t)$

$TF - IDF(t, d, D) = TF(t, d) * IDF(t, D)$

The second task is to calculate the similarity between all the books using the cosine similarity. Cosine similarity is a measure of similarity between two non-zero vectors defined in an inner product space. Cosine similarity is the cosine of the angle between the vectors; that is, it is the dot product of the vectors divided by the product of their lengths

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}},$$

The resulting similarity ranges from -1 meaning exactly opposite, to 1 meaning exactly the same, with 0 indicating orthogonality or decorrelation, while in-between values indicate intermediate similarity or dissimilarity.

2.2 Collaborative-filtering Recommendation

Collaborative-filtering recommendation is the most famous algorithm in recommender systems. This algorithm models the user's taste according to the history of user behavior. GroupLens published the first paper about collaborative filtering and the paper raised user-based collaborative filtering. In 2000, Amazon came up with

item-based collaborative filtering in their paper. These two algorithms are very famous in business recommender systems.

2.2.1 User-based collaborative-filtering

In user-based collaborative filtering, it is considered that a user will like the items that are liked by users with whom they have similar taste. So the first step of user-based collaborative filtering is to find users with similar tastes. In collaborative filtering, the users are considered similar when they like similar items. Simply speaking, given user u and v , $N(u)$ and $N(v)$ are items set liked by u and v respectively. So the similarity of u and v can be simply defined as:

$$s_{uv} = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

There are a lot of similarity algorithm, Equation above is one of them. User u 's likeability for item i can be calculated by:

$$p_{ui} = \sum_{v \in S(u,k) \cap N(i)} s_{uv} p_{vi}$$

2.2.2 Item-based collaborative-filtering

Item-based collaborative-filtering is different, it assumes users will like items that are similar with items that the user liked before. So the first step of item-based collaborative-filtering is to find out items that are similar with what the user liked before. The core point of item-based collaborative-filtering is to calculate the similarity of two items. Item CF considers that items that are liked by more same users, the more similar they are. Assume $N(i)$ and $N(j)$ are user sets who like i and j respectively. So the similarity of i and j can be defined as:

$$s_{ij} = \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|}$$

User u 's likeability of item i can be calculated by:

$$p_{ui} = \sum_{j \in S(i,k) \cap N(u)} s_{ij} p_{uj}$$

Table is an example of Item-based CF recommendation. According to the interest history of all the users for Item A, people who like Item A like Item C as well, so we can conclude that Item A is similar with Item C. While User C likes Item A, so we can deduce that perhaps User C likes Item C as well.

User/Item	Item A	Item B	Item C
User A	X		X
User B	X	X	X

User C	X		recommended
--------	---	--	-------------

User-based and Item-based collaborative-filtering algorithms are all neighborhood-based algorithm, there are also a lot of other collaborative-filtering algorithms. Hoffman raised the Latent Class Model in this paper, the model connects user and item by latent class, which considers that a user will not become interested in items directly. Instead, a user is interested in several categories that contain items, so the model will learn to create the categories according to the user's behavior. On top of the Latent Class Model, researchers came up with the Matrix Decomposition Model, which is called Latent Factor Model as well. There are a lot of models based on matrix decomposition and they mostly came from the Netflix Prize Competition, such as RSVD[28], SVD++[18] and so on.

Besides the Matrix Decomposition Model, the Graph Model is widely applied in collaborative filtering. Baluja introduced a graph model of co-view behind the recommender algorithm of YouTube and also raised a broadcast algorithm on graphs to measure how much a user likes an item. This literature research how to increase serendipity of recommendation result by means of the analysis of the path between nodes.

3. Overview of Dataset:

3.1. Importing Dataset:

Datasets are critical component of AI development because they provide the training data that is used to train and test machine learning models. In this project Datasets such as `final_data.json`, `interactive_test.json` were crawled from a URL (`book_urls.json`) on the internet for implementation purposes.

- Information of Dataset:

```
books = pd.read_json('./data/book_best_001_050.json', lines=True)
books.head(3)
books.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11403 entries, 0 to 11402
Data columns (total 22 columns):
#   Column                Non-Null Count  Dtype
---  -
0   url                   11403 non-null  object
1   title                 11384 non-null  object
2   titleComplete        11384 non-null  object
3   description           11369 non-null  object
4   imageUrl             11368 non-null  object
5   genres               11185 non-null  object
6   asin                 9838 non-null   object
7   isbn                 9073 non-null   object
8   isbn13               9243 non-null   float64
9   publisher            10972 non-null  object
10  author               11384 non-null  object
11  publishDate          11332 non-null  float64
12  characters           6411 non-null   object
13  places               5362 non-null   object
14  ratingHistogram      11384 non-null  object
15  ratingsCount         11368 non-null  float64
16  reviewsCount        11347 non-null  float64
17  numPages             11253 non-null  float64
18  language             11271 non-null  object
19  awards              5003 non-null   object
20  series              5915 non-null   object
21  avgRating            11309 non-null  float64
```

```
interactives.head(3)
```

Python

	book_url	title	user	user_rate	user_review
0	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	992038-tara	Rating 1 out of 5	Ok, before I start a few warnings. This will c...
1	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	167451-suzanne	Rating 5 out of 5	"I'm going to keep going until I succeed — or ...
2	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	8937622-kassidy	Rating 5 out of 5	It's hard for me to believe that I finished th...

3.3. Data Cleaning:

In the data cleaning step the dataset is analyzed and checked for null values, missing values, and duplicate values. The goal of data cleaning is to ensure that the data is accurate, consistent, and free of errors. It is important to clean data as leaving them can negatively impact ML models. This can be done using the pandas.

Drop some columns

```
In [23]: books.drop(['titleComplete', 'imageUrl', 'asin', 'isbn', 'isbn13', 'publishen', 'series', 'characters', 'places', 'awards'],
# books.info()
books.head(3)
```

	url	title	description	genres	author	publishDate	ratingHistogram	ratingsCount	reviewsCount	numPages	language	avgRating
ids.com/book/show/2165.The_O...		The Old Man and the Sea	Librarian's note: An alternate cover edition c...	[Adventure, Fiction, Literary Fiction, America...	[Ernest Hemingway]	8.204832e+11	[47620, 91381, 253168, 362501, 355777]	1110447.0	37865.0	96.0	English	NaN
ads.com/book/show/295.Treasu...		Treasure Island	"For sheer storytelling delight and pure adven...	[Pirates, Adventure, Fiction, Classics, Fantas...	[N.C. Wyeth, Robert Louis Stevenson]	1.000537e+12	[8528, 28655, 128405, 176772, 136475]	478835.0	15923.0	352.0	English	NaN
ads.com/book/show/7244.The_P...		The Poisonwood Bible	The Poisonwood Bible is a story told by the wi...	[Historical, Fiction, Literary Fiction, Classi...	[Barbara Kingsolver]	1.117523e+12	[23947, 39130, 112637, 227128, 329801]	732643.0	26807.0	546.0	English	NaN

Drop books with the same url or uncrawable books

```
books.drop_duplicates(subset='url', keep='first', inplace=True)
books.dropna(subset=['title'], inplace=True)
# title duplicates because of reprints, versions, publishing companies
books.drop_duplicates(subset='title', keep='first', inplace=True)
```

- Book: title, author, description
 - convert title, author to str type
 - fill description missing values with default value(=title)
 - translate description to English, remove meaningless words, convert to lower case, remove punctuations, special characters, emojis and multiple spaces
- Book: genres, language
 - convert to categorical variables
 - fill genres, language missing values with default value

```
books['genres'].fillna('', inplace=True)
# check and handle non-list-like data
books['genres'] = books['genres'].apply(lambda x: x if isinstance(x, list) else [])
books['genres'] = books['genres'].apply(lambda x: pd.Categorical(x))
# default language should be the most popular language
books['language'].fillna('English', inplace=True)
books['language'] = books['language'].astype('category')
```

- Interactives: user reviews and user ratings for the above books

	book_url	title	user	user_rate	user_review
0	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	992038-tara	Rating 1 out of 5	Ok, before I start a few warnings. This will c...
1	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	167451-suzanne	Rating 5 out of 5	"I'm going to keep going until I succeed — or ...
2	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	8937622-kassidy	Rating 5 out of 5	It's hard for me to believe that I finished th...

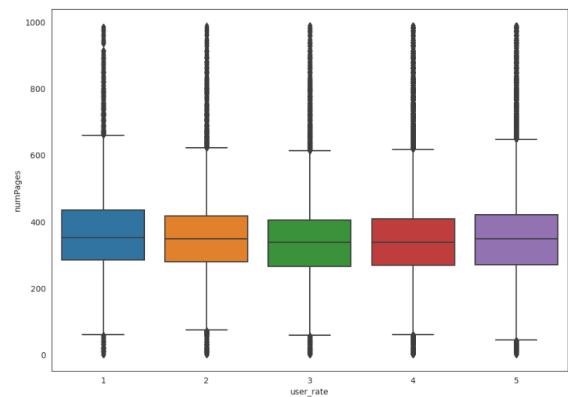
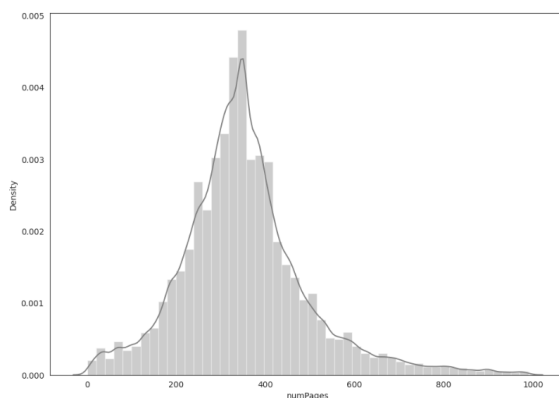
- user-rate: convert to numeric
- user-review: Due to limitations of the web version of google translate, googletrans API does not guarantee that the library would work properly at all

times `text_cleaning_lighter` has removed the translation function and the meaningless words removing function. It's probably, Google has banned your client IP address TODO: use [Google's official translate API] (<https://cloud.google.com/translate/docs>)

	book_url	title	user	user_rate	user_review
0	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	992038-tara	1	ok before i start a few warnings this will con...
1	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	167451-suzanne	5	i m going to keep going until i succeed or die...
2	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	8937622-kassidy	5	it s hard for me to believe that i finished th...
3	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	22106879-jayson	5	a 86 extraordinarynotes it ends too expository...
4	https://www.goodreads.com/book/show/136251.Har...	Harry Potter and the Deathly Hallows	30728719-lily	5	i can t believe its over i ve finally read the...

3.4. Data visualization:

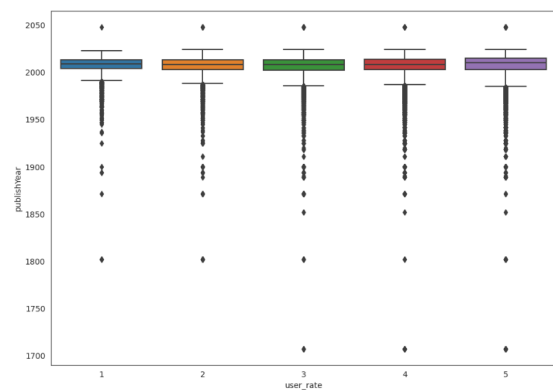
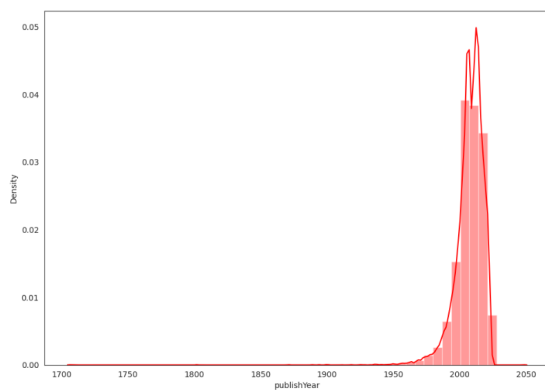
- Visualization is shown through distplot and boxplot for each numeric feature:
- + `num_pages`:



99% of the books have less than 990 pages.

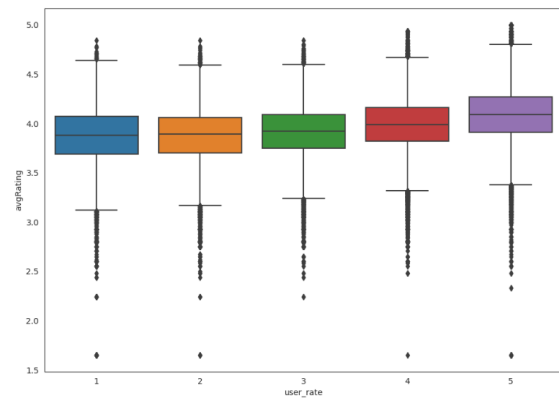
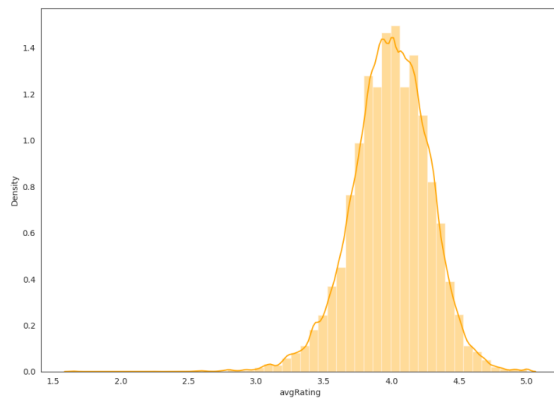
There are nearly 155 books which have ≤ 4 pages, this is absurd. These are outliers we will drop them.

- + `publication_year`:



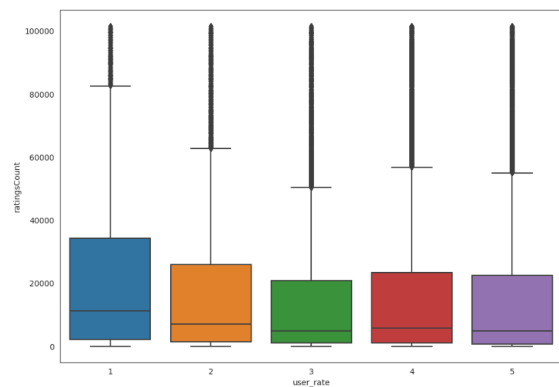
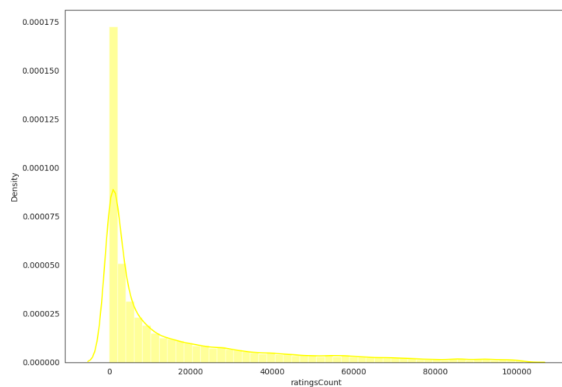
There were some outliers, `publishYear` for some books was >2023 and there were some books where `publication year` <1900 . We removed the outlier data points from our dataset.

- + average rating:



We can see that the majority of books have got average rating of 3.8 and there are very few books which have a very high and low average-rating, which is expected. We cannot drop the books in any range on this feature.

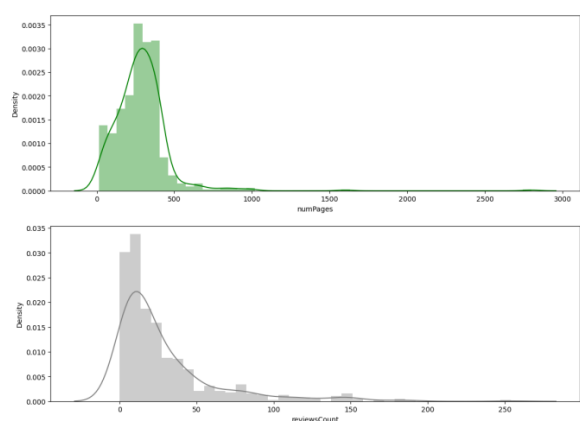
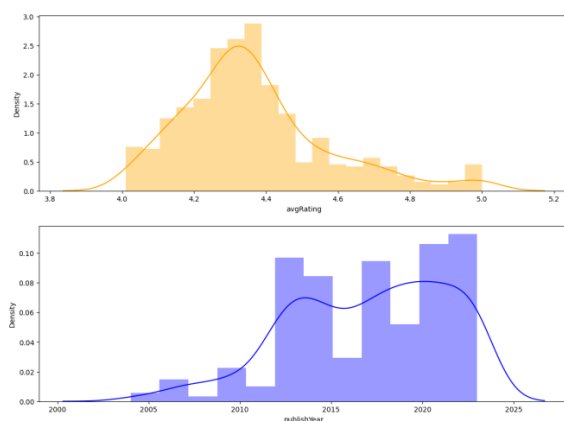
+ ratings_count:



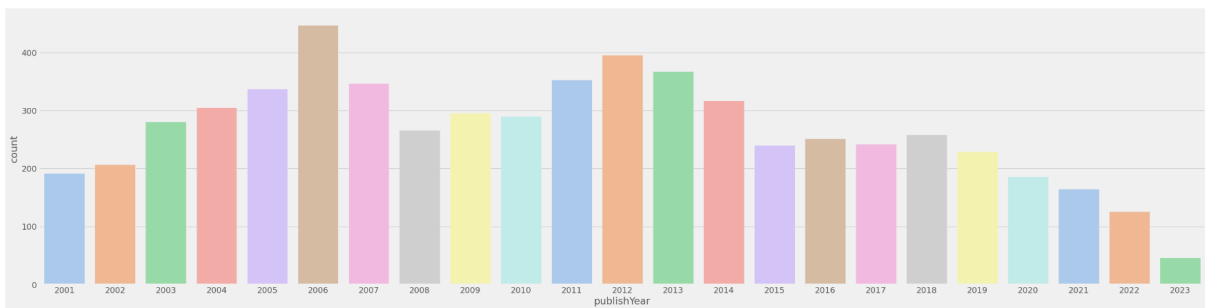
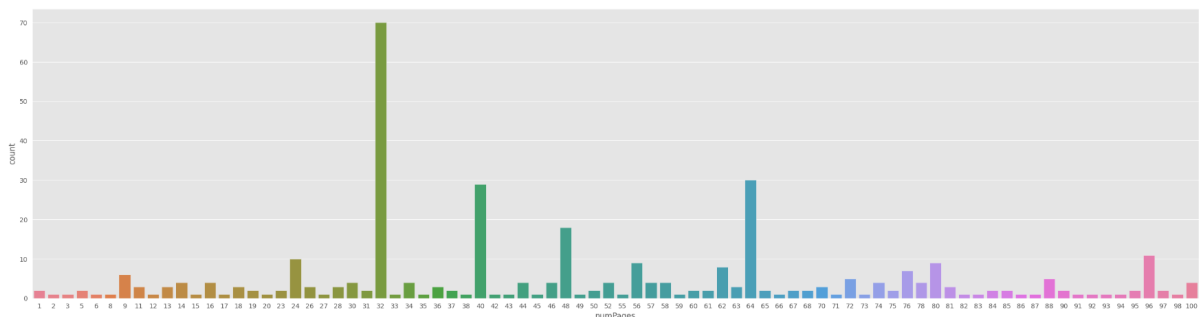
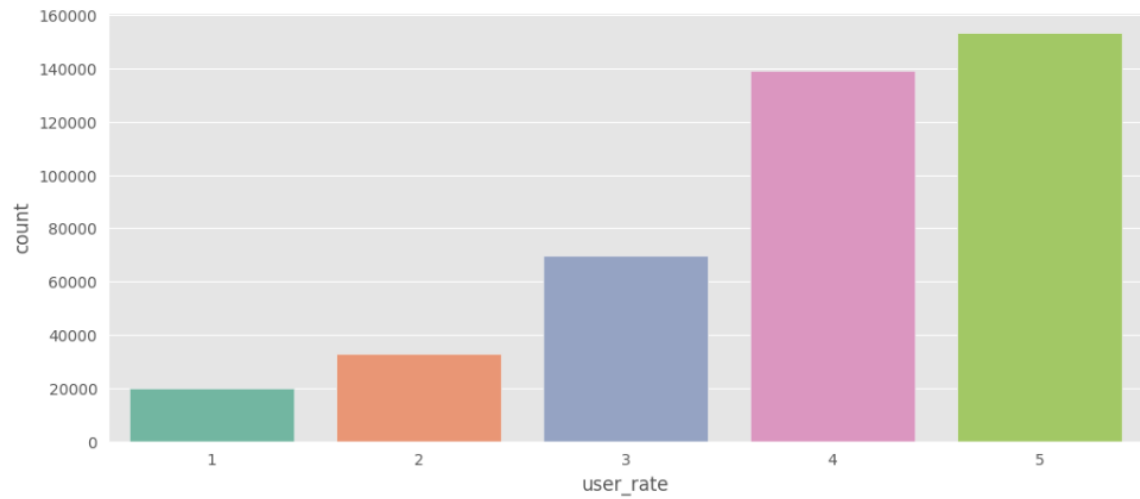
90% of books have been rated by less than 101601 users. Data is distributed across all userRating, we will keep it.

- Visualization through count plot and distplot:

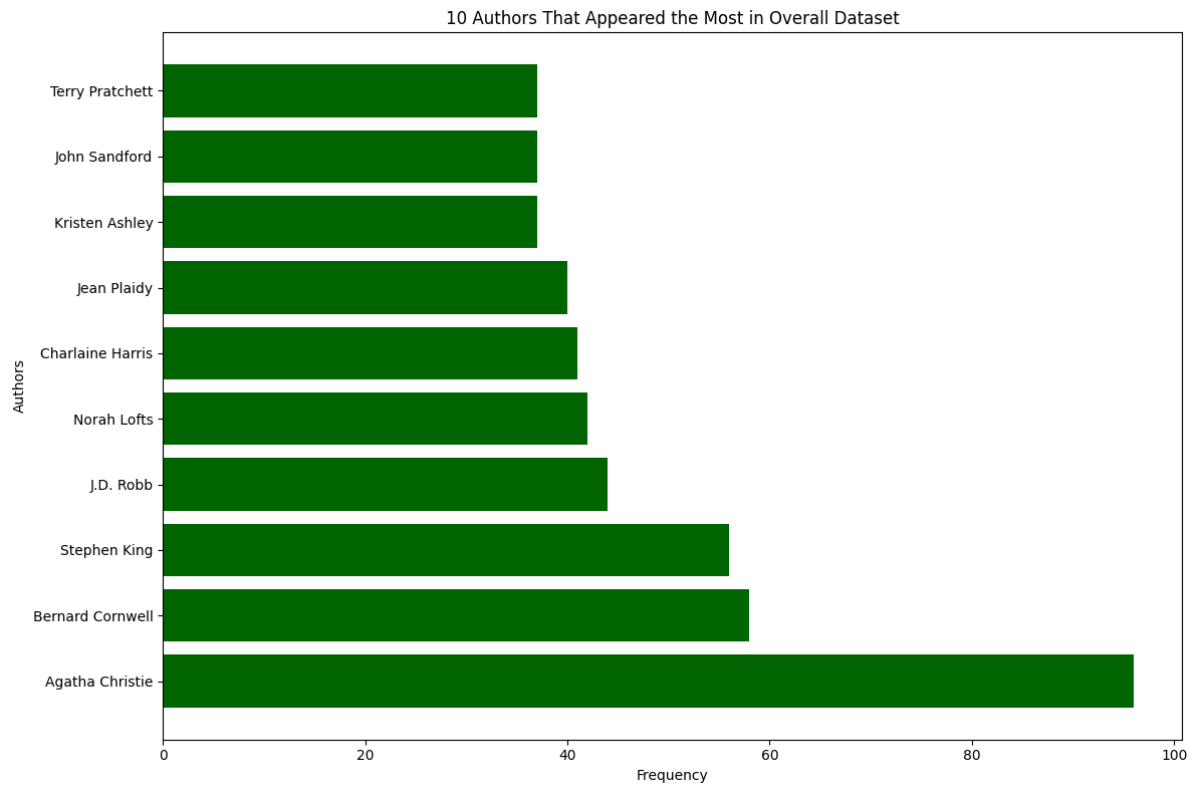
+ It shows the variation in the data distribution corresponding to different features:



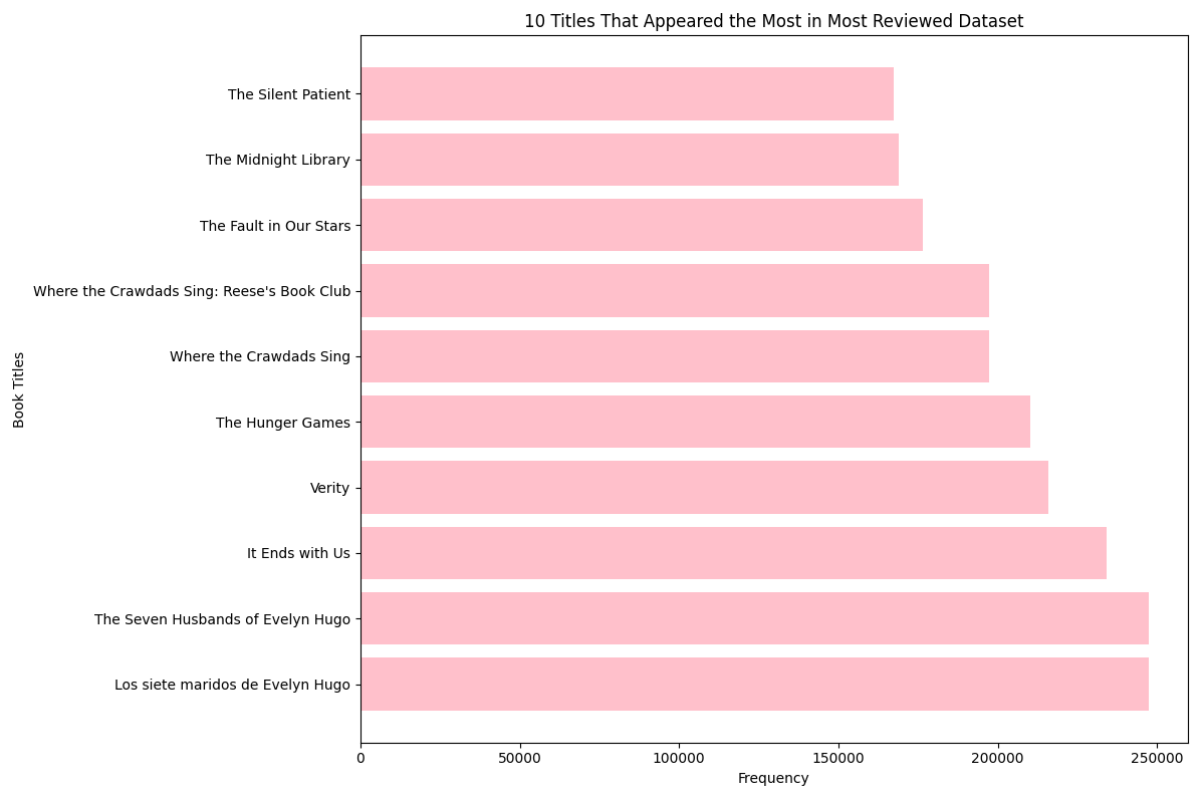
+ It shows the count of different categories present in data.



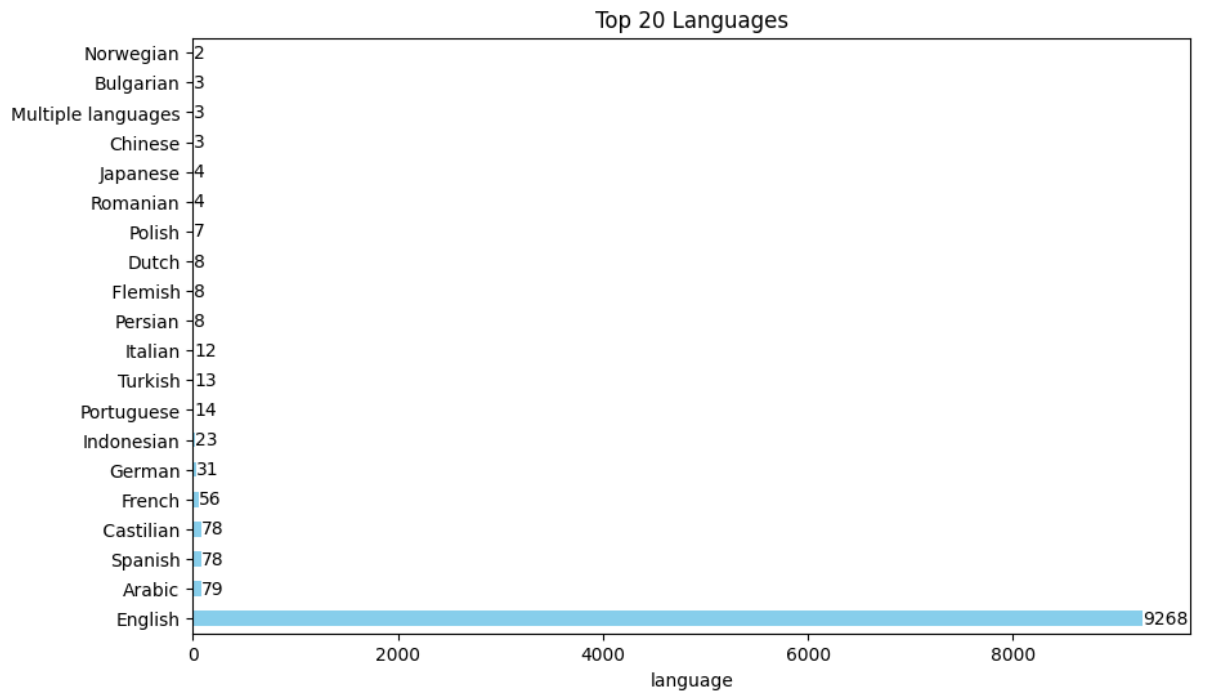
- Top 10 Authors that appeared the Most in Overall Dataset:



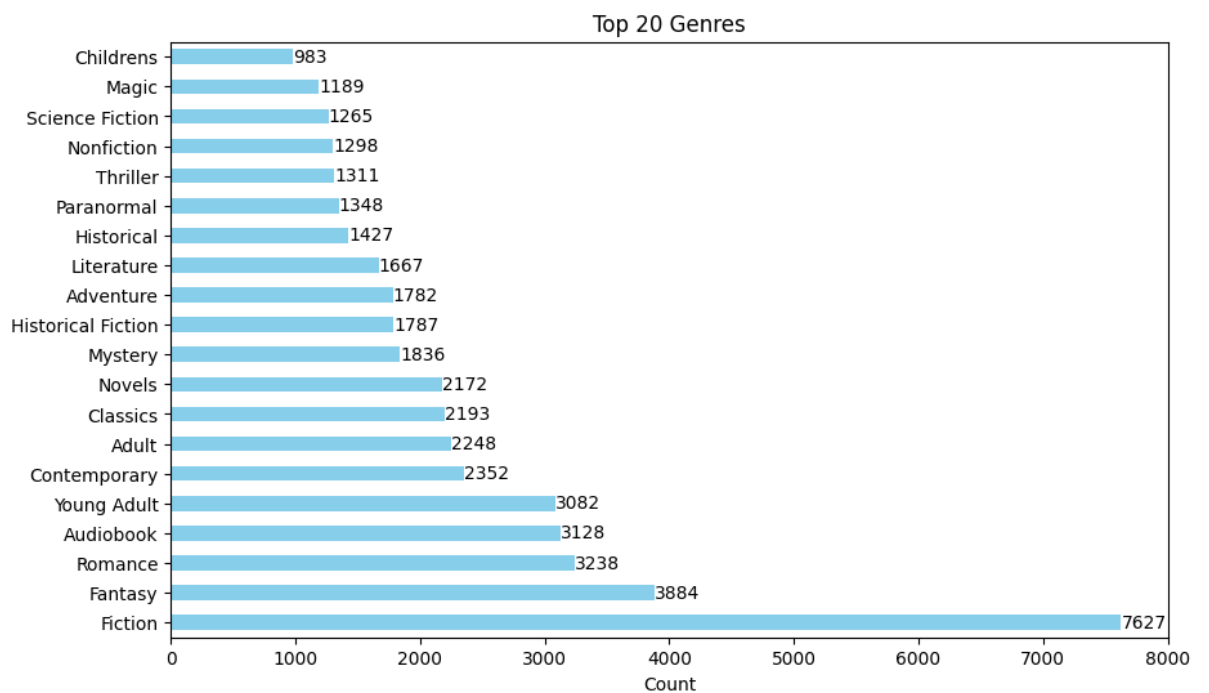
- Top 10 Books have the highest review:



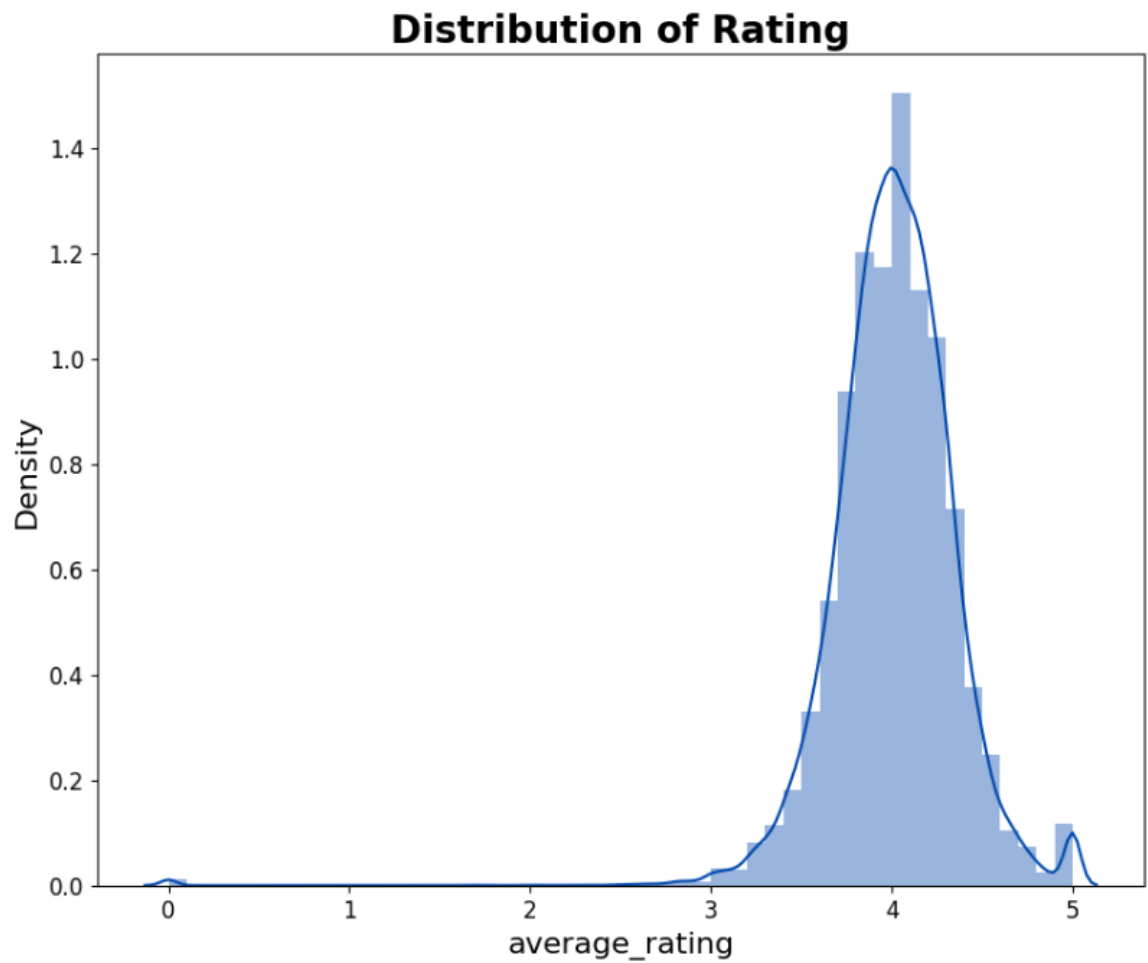
- 20 languages that appeared the most in overall dataset:



- Top 20 most popular book genres:

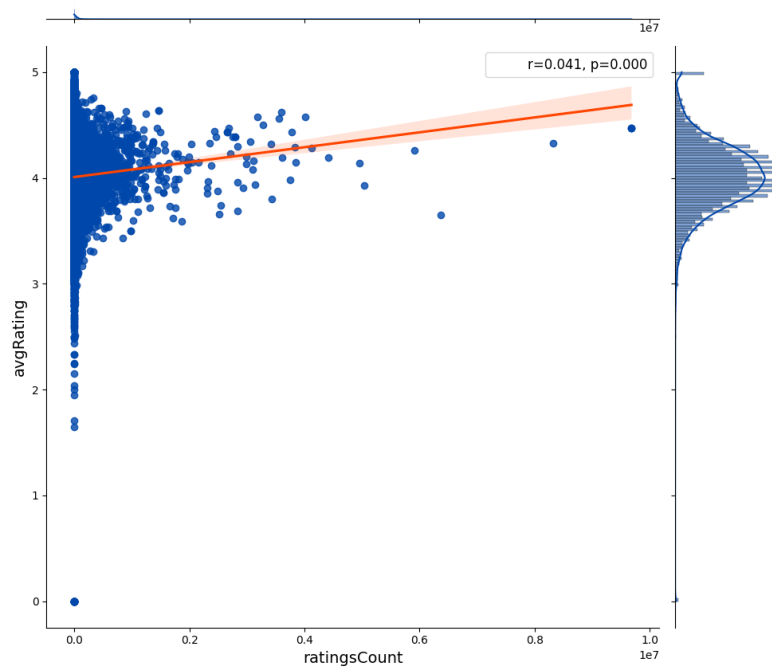


- Distribution of ratings:



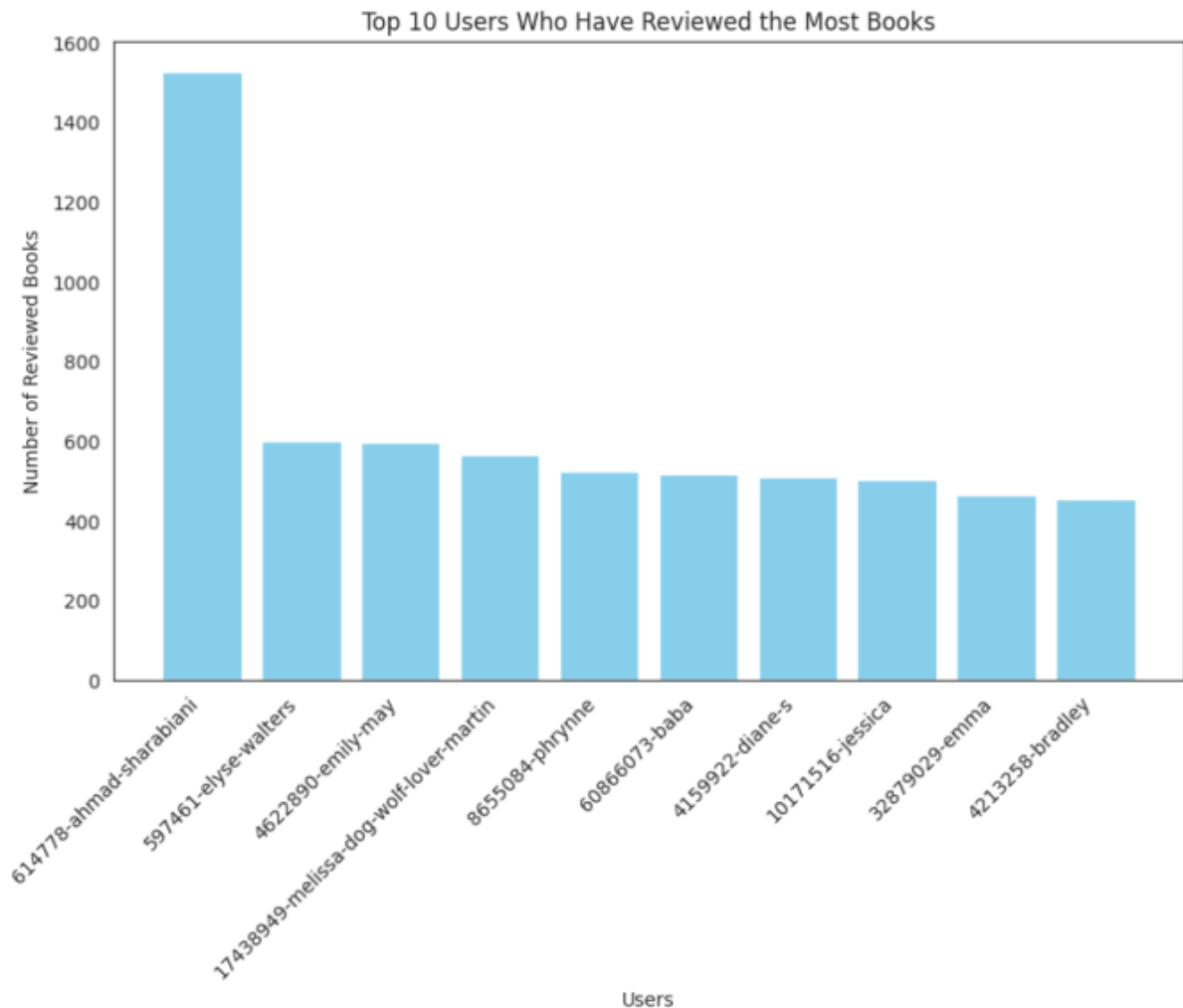
- Comparison of ratings count and avg rating per book:

Comparison of ratings count and average rating per book



A book that is popular (has lots of ratings) is more likely to get a good rating. However, if we look at our data, the correlation between avgRating and ratingsCount is not too big, which means that many popular books have low ratings.






- Top 10 users who have reviewed the most books:



4. Experiment Results

```
content_based_recommender(["the selection","the ghost writer"])
```

Recommendations books:

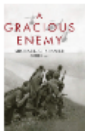




	url	title	imageUrl
12106	Goodreads	story of the ghost	
5232	Goodreads	collected ghost stories	
7868	Goodreads	the selection series 0.5, 1-2 box set	
30527	Goodreads	love you to death, darling	
10504	Goodreads	the ghost writer	



```
book_recommend_with_user('13056902')
```

```
Index([1825, 28234, 3771, 30788, 1692, 9724, 19646, 10894, 2789, 17313], dtype='int64')
```

Recommendations books:

	url	title	imageUrl
1825	Goodreads	a gracious enemy	
28234	Goodreads	the crooked hinge	
3771	Goodreads	a gracious enemy & after the war volume one	
30788	Goodreads	hag's nook	
1692	Goodreads	princess: a true story of life behind the veil in saudi arabia	

5. Conclusion

Recommendation systems are very popular in e-commerce applications such as online book stores and can significantly rise the company's revenue generation. Recommendation system helps us to lighten the information overloading problem where the user will have a lot of choices and not be able to understand perfectly what to buy or what to see. Recommendation system solves this problem and provides users with personalized content after searching a large volume of information. The proposed work to design a recommendation system based on content-based method and collaborative filtering is much more accurate than the existing systems.