ASSIGNMENT SUBMISSION FORM

This will be the first page of your assignment

Course Name	: **Machine Learning – Unsupervised Learning 2**
Assignment Title	: **Group Project**
Submitted by	: **Group 6**

Student Name	**PG ID**
Unnati Khinvasara	12120097
Harsimar Singh Arora	12120011
Rohini Purnima	12120027
Rohit Thakur	12120040
Mohit Kothari	12120035

ISB Honour Code

- I will represent myself in a truthful manner.
- I will not fabricate or plagiarise any information with regard to the curriculum.
- I will not seek, receive or obtain an unfair advantage over other students.
- I will not be a party to any violation of the ISB Honour Code.
- I will personally uphold and abide, in theory and practice, the values, purpose and rules of the ISB Honour Code.
- I will report all violations of the ISB Honour Code by members of the ISB community.
- I will respect the rights and property of all in the ISB community.
- I will abide by all the rules and regulations that are prescribed by ISB.

Note: Lack of awareness of the ISB Honour Code is never an excuse for a violation. Please go through the Honour Code in the student handbook, understand it completely. Please also pay attention to the following points:

- Please do not share your assignment with your fellow students under any circumstances if the Honour Code scheme prohibits it. The HCC considers both parties to be guilty of an Honour Code violation in such circumstances.
- If the assignment allows you to refer to external sources, please make sure that you cite all your sources. Any material that is taken verbatim from an external source (website, news article etc.) must be in quotations. A much better practice is to paraphrase the source material (it still must be cited).

(Please start writing your assignment below)

Book Recommendation System using Collaborative Filtering

Submitted by –

Student Name	PG ID
Unnati Khinvasara	12120097
Harsimar Singh Arora	12120011
Rohini Purnima	12120027
Rohit Thakur	12120040
Mohit Kothari	12120035

EXECUTIVE SUMMARY

- **Stakeholder & Business Problem** – Implementing a Book Recommendation System for Crossword Bookstore's recently launched website to drive revenue & satisfy customer experience
- **Business Solution -** Using Crosswords' User Rating History to build a recommendation engine based on collaborative filtering
- **Success Criterias** – Metrics such as incremental revenue, clicks on recommendation pane, decrease in page bounces, change in books sold per user, etc. in an AB Testing environment
- **Data Description** – 2 related Datasets taken from stakeholder's historical data –
  - Books Dataset – 271k rows in total with features like Title, Author, ISBN Code & Publication Year
  - Ratings Dataset – 1.14m rows from 105k users with features UserID, ISBN Code, Book Ratings
- **Data Preparations** – Both Datasets have been merged based on common key ISBN Code and pivot is created on the basis of Top Users & Top Books criteria for quality recommendation dataset
- **Methodology –** Usage of item-to-item collaborative filtering using cosine similarity as a metric
- **Interpretation –** Highest cosine metric for the input book-name would be sorted and filtered in descending order, out of which Top-5 Books (with highest cosine metric) would be recommended
- **Recommendations for improvements**
  - Adding more dimensions to the data to increase accuracy of recommendation (like genre, age, etc) (Secondary Model taking Publication Year as additional input has been made as sample)
  - Adopt Hybrid Filtering approach using collaborative as well as content-based filtering
  - Match Recommendations from Model to available inventory & stock in hand
  - Enhance business offerings by providing options for bundles or "Book of the Month" category

"People don't know what they want until you show it to them."

~ Steve Jobs

TITLE - Book Recommendation System using Collaborative Filtering

SCOPE OF THE PROJECT

Background:

Crossword India is a huge book chain in India present in over 25+ states and has recently launched their bookstore’s website for keeping up with the e-commerce trend. They have a huge catalogue of books available in the offline stores which they have taken this online store. They have developed capability to deliver books across all pin codes in India and are getting orders from all new as well as existing customers.

Thus, Crosswords sees a huge potential in this e-commerce vertical of the book sales and wishes to explore ways of enhancing and driving business through the same.

Business Objective:

The high-level business objective for Crosswords is to drive up the business revenue by increasing sales through enhancing user engagement / interactions (clicks & book views) / visits on their newly-launched website.

Problem Statement:

Crossword wishes to improve their e-commerce sales through enhancing customer experience by adding book recommendations on the website that provides additional reading options that their customer can choose from.

Possible Solution:

We propose on implementing a recommendation system which focuses on improving customer experience and engagement by recommending relevant books through collaborative filtering. This keeps the customer engaged within the website, drives additional sales, and reduces page bounces.

The benefit of this solution would be that with the implementation of recommender system, online sale and users’ engagement on website should improve. Further, we get the opportunity of improving the visibility of crossword's catalogue and providing curated deals and offers to the customers.

Data-driven Approach:

Crosswords has provided us with their user history and book catalogue whereby assuring authenticity of the data being used for the recommendation engine. Using the same, we have proposed to adopt collaborative filtering approach to create the recommendation system.

This book recommendation engine will take into consideration similar books based on their ratings. For this purpose, Crosswords Management has given instructions that only those books and user ratings should be recommended which are above a certain threshold as given below -

- Top-rated Books Criteria – Books which have been rated more than 35 times
- Top User Criteria – Users who have purchased & rated more than 100 books

Thus, based on instructions by Crosswords, the recommendation system would disregard books which have fewer ratings than 35 in its suggestions.

Success Criteria:

AB Test will be done to quantify the impact of recommender system. Test group users will be exposed to the Recommendation Engine whereas the control group user will be status quo.

We can quantify the success of the book recommendation engine using the following KPIs from the Crossword website –

(i) Increase in Revenue / Sale Metric
(ii) Change in Books sold per user
(iii) Increase in User Engagement via click per recommendation page
(iv) Decrease in page bounces
(v) Increase in Books viewed per user

DATA DESCRIPTION AND PREPARATION

Data Description

- Two Datasets (books & users) were given by Crossword. Here are the details of each dataset:
  - Books: Dataset contains the following attributes of books -

    a. Title: Title of book
    b. Author: Book author
    c. Year-of-publication
    d. Publisher
    e. ISBN code: 271k Unique codes
    f. Cover images.

    Dataset has 242k no. of unique books from 102k authors.
    Refer Appendix for Sample Books Database.

  - Ratings: Dataset has ratings given by users to each book. Attributes in this dataset are:

    a. Userid: Unique identifier for users
    b. ISBN code
    c. Ratings: Rating by user for each ISBN code

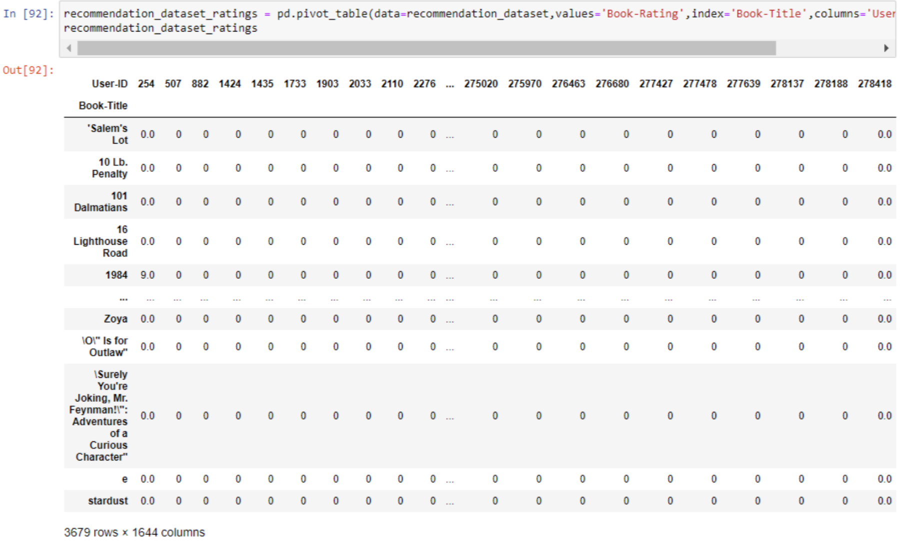
    Total Ratings available are 1.14 million wherein 105k no. of users have rated 340k no. of books. Refer Appendix for sample Ratings dataset.

Data Preparation

- Both datasets were merged on primary key of ISBN to create a common dataframe having users, book titles, author ratings, etc. as attributes. The sample merged dataset is available in the appendix.

- The newly merged dataframe, was then filtered on the conditions given by Crossword (for Top Books & Top Users) to create optimal input for recommendation engine.
  - Top Rated Books: Created a list of 3679 books having >35 ratings.
  - Top Users: Created a list of 1648 users rated who have rated > 100 books

- Post this step, the list of books & users were filtered from the merged dataset. After which, we created a pivot table of book title x user ratings which can be used to identify similar books using cosine similarity score. The pivot table dimensions are 3679x1644 (sample below)



```
In [92]: recommendation_dataset_ratings = pd.pivot_table(data=recommendation_dataset,values='Book-Rating',index='Book-Title',columns='User
         recommendation_dataset_ratings
```

Out[92]:

User-ID	254	507	882	1424	1435	1733	1903	2033	2110	2276	...	275020	275970	276463	276680	277427	277478	277639	278137	278188	278418
Book-Title																					
'Salem's Lot	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
10 Lb. Penalty	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
101 Dalmatians	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
16 Lighthouse Road	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
1984	9.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Zoya	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
\O\" Is for Outlaw"	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
\Surely You're Joking, Mr. Feynman!\": Adventures of a Curious Character"	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
e	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0
stardust	0.0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0.0

3679 rows × 1644 columns

DATA MINING APROACHES

The approach that we followed for recommendation of books based on ratings is *"item to item collaborative filtering"*. Our main goal in our approach is to recommend the book most suited according to the rating given by users, we have discussed in data preparation part about selecting the popular books based on ratings and the users who have frequently rated the books.

Item-to-item Collaborative Filtering

The approach here is to match the book ratings to similar rated books rather than matching the user ratings of one user to similar users. To determine the most-similar match for a given item, the algorithm builds a similar-items table by finding items that users tend to rate together. We were able to build a book-to-book matrix by iterating through all pairs and then computing the *cosine-similarity matrix*. The key to Item-to-item Collaborative Filtering's performance is that it creates similar items table. The important component it looks is similar items from users' ratings and it scales independently to the catalogue size (total number of users). One of the reasons that this algorithm is best suited because it recommends highly correlated similar items.

Cosine-Similarity

The similarities between the pair of items are computed using cosine similarity metric. The reason of following this approach is because our data is sparse which means many ratings are undefined. Cosine similarity simply measures the similarity between two Vectors, lesser the angle between the vectors more similar they are to each other. Our idea of using Cosine similarity is because even if the two similar documents are far apart by the Euclidean distance because of the size they might still have angle smaller between them, thus having the similar character.

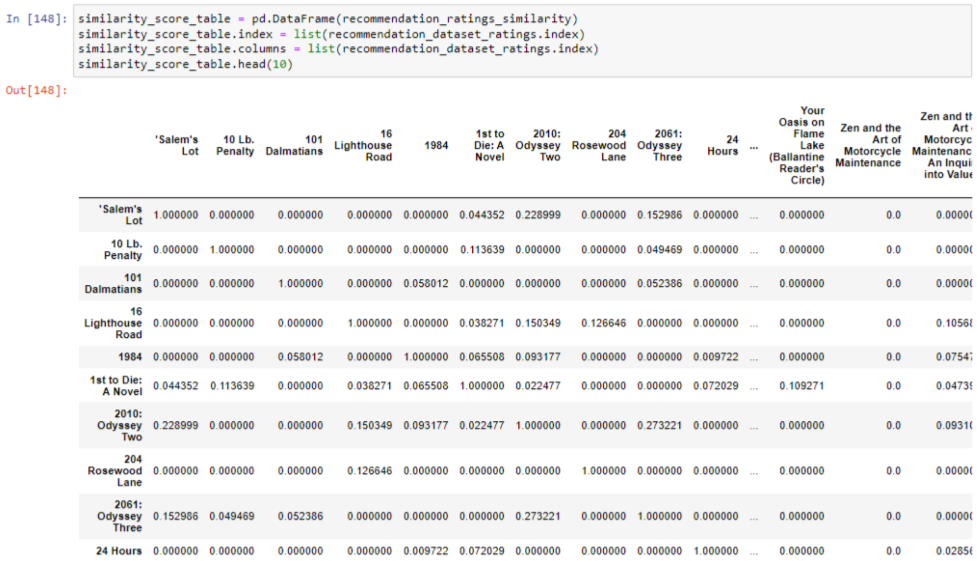
$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Further to get optimal results through cosine similarity we have used a function to sort the distance in descending order to publish the list of most similar books based on its ratings.

INSIGHT AND INTERPRETATION

Based on the above data mining approach for the book recommendation system, we have developed a cosine similarity matrix which shows us the similarity of books based on their ratings.

As per the sample matrix below, while comparing the first book – "Salem's Lot" to other books, since its cosine score is highest at 0.228999 with 'Odyssey: Two', we can interpret that ratings of "Salem's Lot" are more similar to "Odyssey : Two" than any other book displayed below.



In [148]:
```
similarity_score_table = pd.DataFrame(recommendation_ratings_similarity)
similarity_score_table.index = list(recommendation_dataset_ratings.index)
similarity_score_table.columns = list(recommendation_dataset_ratings.index)
similarity_score_table.head(10)
```

Out[148]:

	'Salem's Lot	10 Lb. Penalty	101 Dalmatians	16 Lighthouse Road	1984	1st to Die: A Novel	2010: Odyssey Two	204 Rosewood Lane	2061: Odyssey Three	24 Hours	...	Your Oasis on Flame Lake (Ballantine Reader's Circle)	Zen and the Art of Motorcycle Maintenance	Zen and th Art Motorcyc Maintenanc An Inqui into Value
'Salem's Lot	1.000000	0.000000	0.000000	0.000000	0.000000	0.044352	0.228999	0.000000	0.152986	0.000000	...	0.000000	0.0	0.0000
10 Lb. Penalty	0.000000	1.000000	0.000000	0.000000	0.000000	0.113639	0.000000	0.000000	0.049469	0.000000	...	0.000000	0.0	0.0000
101 Dalmatians	0.000000	0.000000	1.000000	0.000000	0.058012	0.000000	0.000000	0.000000	0.052386	0.000000	...	0.000000	0.0	0.0000
16 Lighthouse Road	0.000000	0.000000	0.000000	1.000000	0.000000	0.038271	0.150349	0.126646	0.000000	0.000000	...	0.000000	0.0	0.1056
1984	0.000000	0.000000	0.058012	0.000000	1.000000	0.065508	0.093177	0.000000	0.000000	0.009722	...	0.000000	0.0	0.0754
1st to Die: A Novel	0.044352	0.113639	0.000000	0.038271	0.065508	1.000000	0.022477	0.000000	0.000000	0.072029	...	0.109271	0.0	0.0473
2010: Odyssey Two	0.228999	0.000000	0.000000	0.150349	0.093177	0.022477	1.000000	0.000000	0.273221	0.000000	...	0.000000	0.0	0.0931
204 Rosewood Lane	0.000000	0.000000	0.000000	0.126646	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	...	0.000000	0.0	0.0000
2061: Odyssey Three	0.152986	0.049469	0.052386	0.000000	0.000000	0.000000	0.273221	0.000000	1.000000	0.000000	...	0.000000	0.0	0.0000
24 Hours	0.000000	0.000000	0.000000	0.000000	0.009722	0.072029	0.000000	0.000000	0.000000	1.000000	...	0.000000	0.0	0.0285

For the purpose of this Recommendation Engine, based on the input given by the user, the model would sort the cosine matrix in a descending order and then the top-5 highest cosine scored books would be displayed on the website of Crossword. This would induce the customer to explore more similar books which could be quantified from the KPIs listed in the scope section above (i.e. increase in clicks on the recommender rail, increase in sale numbers, etc)

The results of recommendation can be shown in a rail on website at a prominent position visible to users. This will increase the engagement of the users & ultimately sales of books thus solving the business objective.

Kindly refer Appendix for a screenshot of the top-5 book recommendations in case of the following books –

- Stardust
- Miss Smilla's feelings for snow

Operational Requirements & Constraints

With respect to operational requirements, we would need to make sure that the model is dynamic and takes into consideration the latest user ratings. This would ensure that the Recommendation Engine is up-to-date and gives the best possible results. The major constraint in this would be to segregate the genuine human ratings versus the automated bot ratings for the books on the website.

RECOMMENDATIONS

1) Adding other dimensions to the recommendation engine in addition to ratings

To further enhance the recommendation system developed above, we can further explore adding additional layers of dimensions based on additional book and user data, such that customer experience is enhanced and there is increase in success criteria listed in the scope.

One such example of this would be adding year of book publication as a dimension. This would recommend books closer to the year of the input given and hence give more accurate and latest book recommendations. This would be particularly beneficial if someone is looking for technology books for which year of publication is very critical. Kindly refer Appendix for Final Model Recommendation based on Publication Year.

In line with above, we can add following dimensions to enhance the accuracy of the book recommendation system –

(i) Genre - We can expand dataset to include the genre of the books and thus run the recommendation model on selected genre. This will give a more curated list of related books for the said customer.

(ii) Written Feedback - Further, we can even recommend Crossword, to implement a feedback or written review at the end of the sale, on which by conducting text analytics, we could understand specific reviews of users and hence add that as an additional dimension to the model.

(iii) User Age Group - Capturing age group of users recommending will also enable us to understand another dimension for the recommendation engine and the model would be more robust and useful in terms of relatable book searches.

2) Adopt Hybrid Filtering Approach
We could explore adopting a hybrid approach, that is, Content-based filter along with Collaborative, wherein the model would recommend books based on ratings of similar books, as well as user history. For this, Crosswords would need to capture additional user data and history from their database.

3) Matching Recommendation with available Inventory
We can also explore matching the user & book dataset with inventory at hand, and thus make sure that the recommendation engine is primarily showing recommendations of available books with Crossword. This would be beneficial for both, the business as well as the customer since the model would not show recommendation of books not in stock.

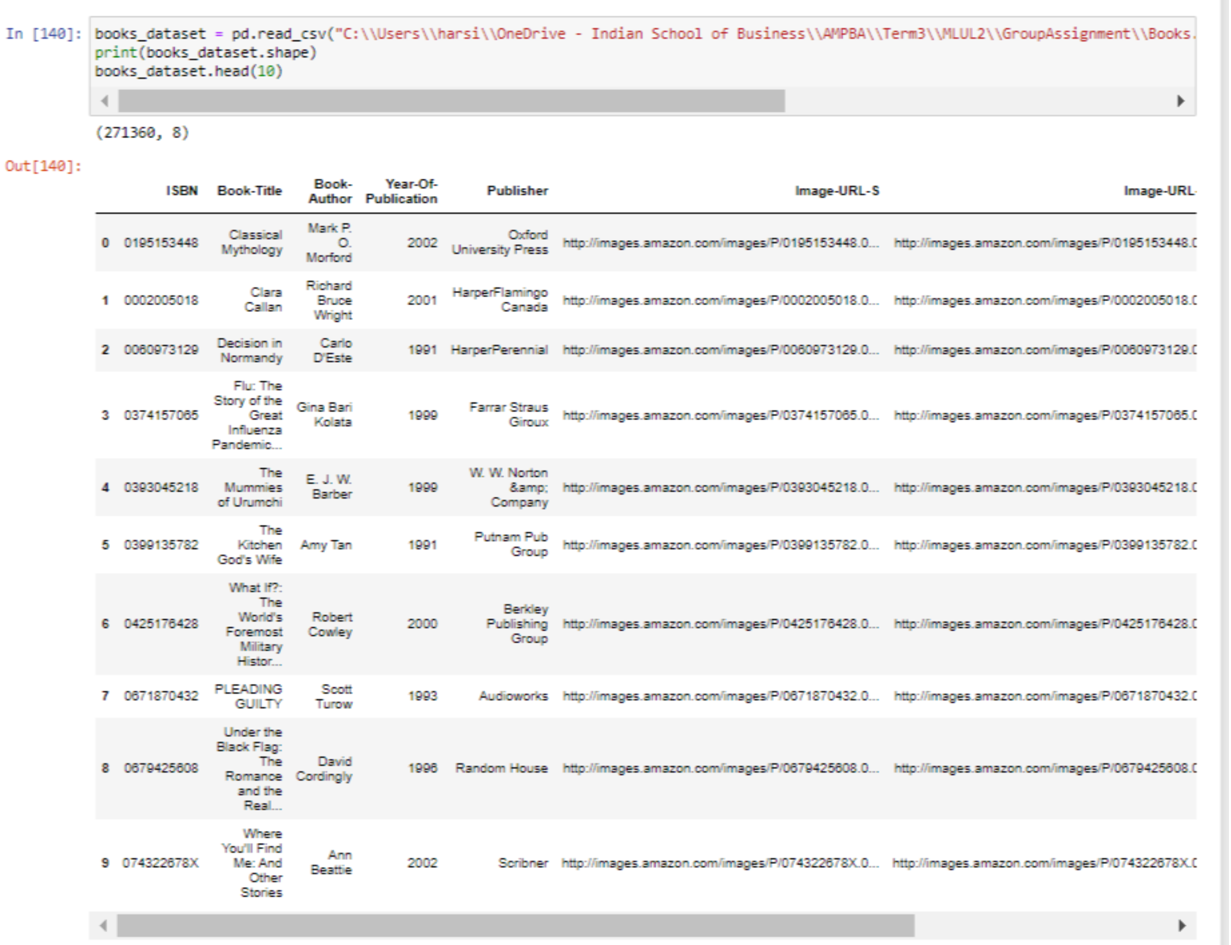
4) Enhancement in Business Offerings
Understand what books are bought together in the past and offer bundles / discounts on a set of 2 or 3 books as per the use attributes. We could also have a “Books to Read” for the month section based on book popularity to drive interaction and engagement.

5) Production & Deployment Constraints -
(i) Increase in Dataset Volume increases the complexity of the system
(ii) Sparsity issue of the data set may decrease the accuracy of the result

APPENDIX

Dataset Source - https://www.kaggle.com/code/alaminbhuyan/books-recommendation-using-collaborative-filtering/data

Sample View of Books Dataset -



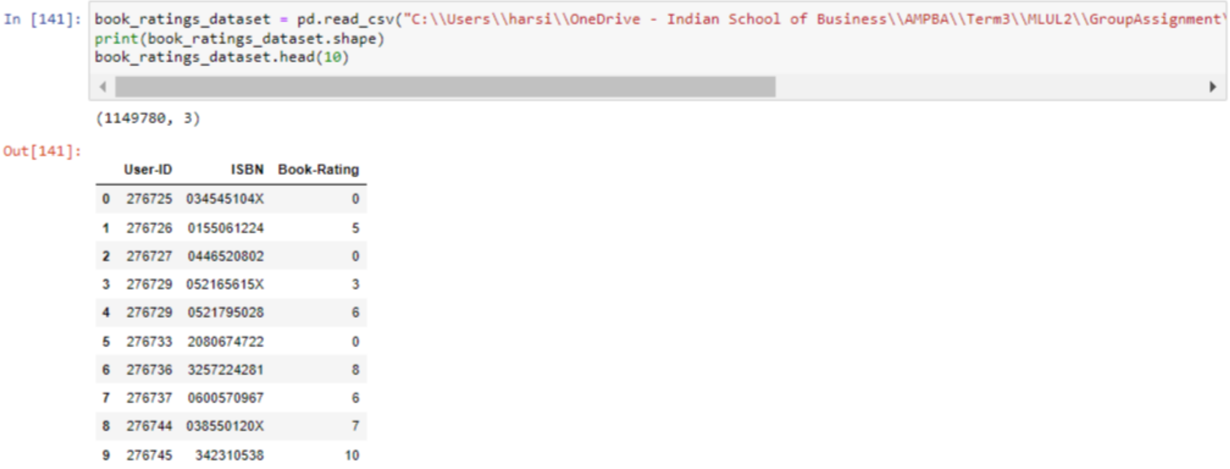
```
In [140]: books_dataset = pd.read_csv("C:\\Users\\harsi\\OneDrive - Indian School of Business\\AMPBA\\Term3\\MLUL2\\GroupAssignment\\Books.
          print(books_dataset.shape)
          books_dataset.head(10)
```

```
(271360, 8)
```

Out[140]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher	Image-URL-S	Image-URL-
0	0195153448	Classical Mythology	Mark P. O. Morford	2002	Oxford University Press	http://images.amazon.com/images/P/0195153448.0...	http://images.amazon.com/images/P/0195153448.0
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/0002005018.0
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/0060973129.0
3	0374157065	Flu: The Story of the Great Influenza Pandemic...	Gina Bari Kolata	1999	Farrar Straus Giroux	http://images.amazon.com/images/P/0374157065.0...	http://images.amazon.com/images/P/0374157065.0
4	0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton &amp; Company	http://images.amazon.com/images/P/0393045218.0...	http://images.amazon.com/images/P/0393045218.0
5	0399135782	The Kitchen God's Wife	Amy Tan	1991	Putnam Pub Group	http://images.amazon.com/images/P/0399135782.0...	http://images.amazon.com/images/P/0399135782.0
6	0425176428	What If?: The World's Foremost Military Histor...	Robert Cowley	2000	Berkley Publishing Group	http://images.amazon.com/images/P/0425176428.0...	http://images.amazon.com/images/P/0425176428.0
7	0671870432	PLEADING GUILTY	Scott Turow	1993	Audioworks	http://images.amazon.com/images/P/0671870432.0...	http://images.amazon.com/images/P/0671870432.0
8	0679425608	Under the Black Flag: The Romance and the Real...	David Cordingly	1996	Random House	http://images.amazon.com/images/P/0679425608.0...	http://images.amazon.com/images/P/0679425608.0
9	074322678X	Where You'll Find Me: And Other Stories	Ann Beattie	2002	Scribner	http://images.amazon.com/images/P/074322678X.0...	http://images.amazon.com/images/P/074322678X.0

Sample View of Ratings dataset



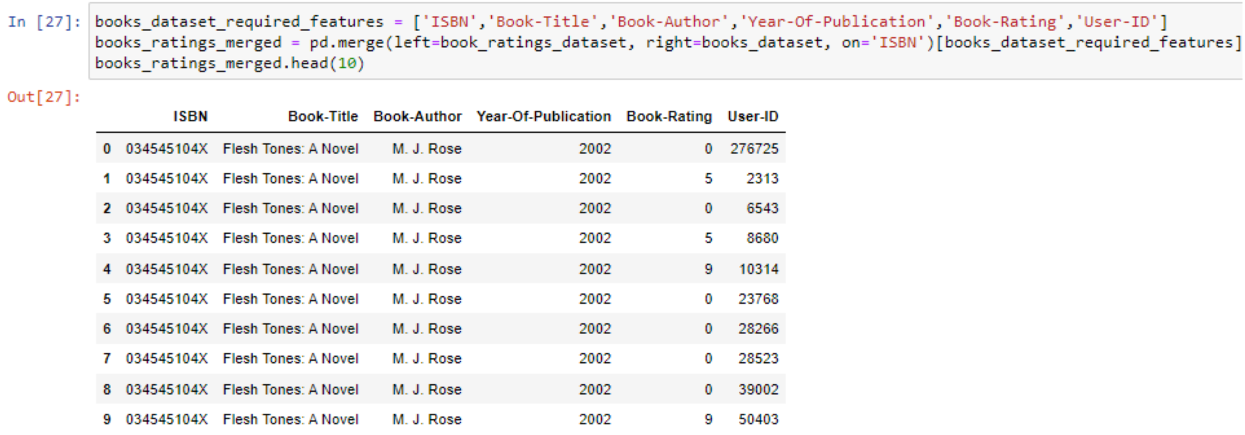
```
In [141]: book_ratings_dataset = pd.read_csv("C:\\Users\\harsi\\OneDrive - Indian School of Business\\AMPBA\\Term3\\MLUL2\\GroupAssignment\
          print(book_ratings_dataset.shape)
          book_ratings_dataset.head(10)
```

```
(1149780, 3)
```

Out[141]:

	User-ID	ISBN	Book-Rating
0	276725	034545104X	0
1	276726	0155061224	5
2	276727	0446520802	0
3	276729	052165615X	3
4	276729	0521795028	6
5	276733	2080674722	0
6	276736	3257224281	8
7	276737	0600570967	6
8	276744	038550120X	7
9	276745	342310538	10

Merged Dataset

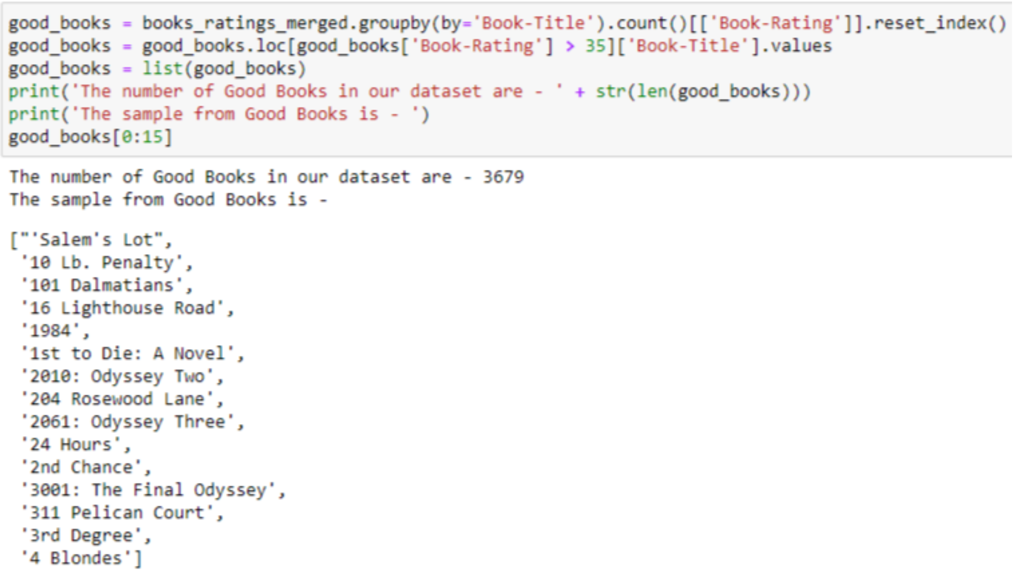


```
In [27]: books_dataset_required_features = ['ISBN','Book-Title','Book-Author','Year-Of-Publication','Book-Rating','User-ID']
         books_ratings_merged = pd.merge(left=book_ratings_dataset, right=books_dataset, on='ISBN')[books_dataset_required_features]
         books_ratings_merged.head(10)
```

Out[27]:

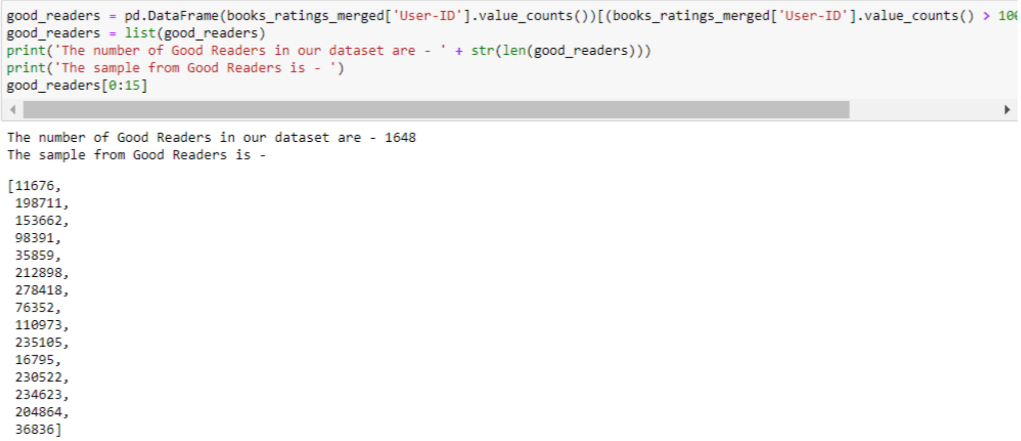
	ISBN	Book-Title	Book-Author	Year-Of-Publication	Book-Rating	User-ID
0	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	276725
1	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	5	2313
2	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	6543
3	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	5	8680
4	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	9	10314
5	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	23768
6	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	28266
7	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	28523
8	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	0	39002
9	034545104X	Flesh Tones: A Novel	M. J. Rose	2002	9	50403

Sample View for Top Books & Users as per Crosswords instructions



```
good_books = books_ratings_merged.groupby(by='Book-Title').count()[['Book-Rating']].reset_index()
good_books = good_books.loc[good_books['Book-Rating'] > 35]['Book-Title'].values
good_books = list(good_books)
print('The number of Good Books in our dataset are - ' + str(len(good_books)))
print('The sample from Good Books is - ')
good_books[0:15]
```

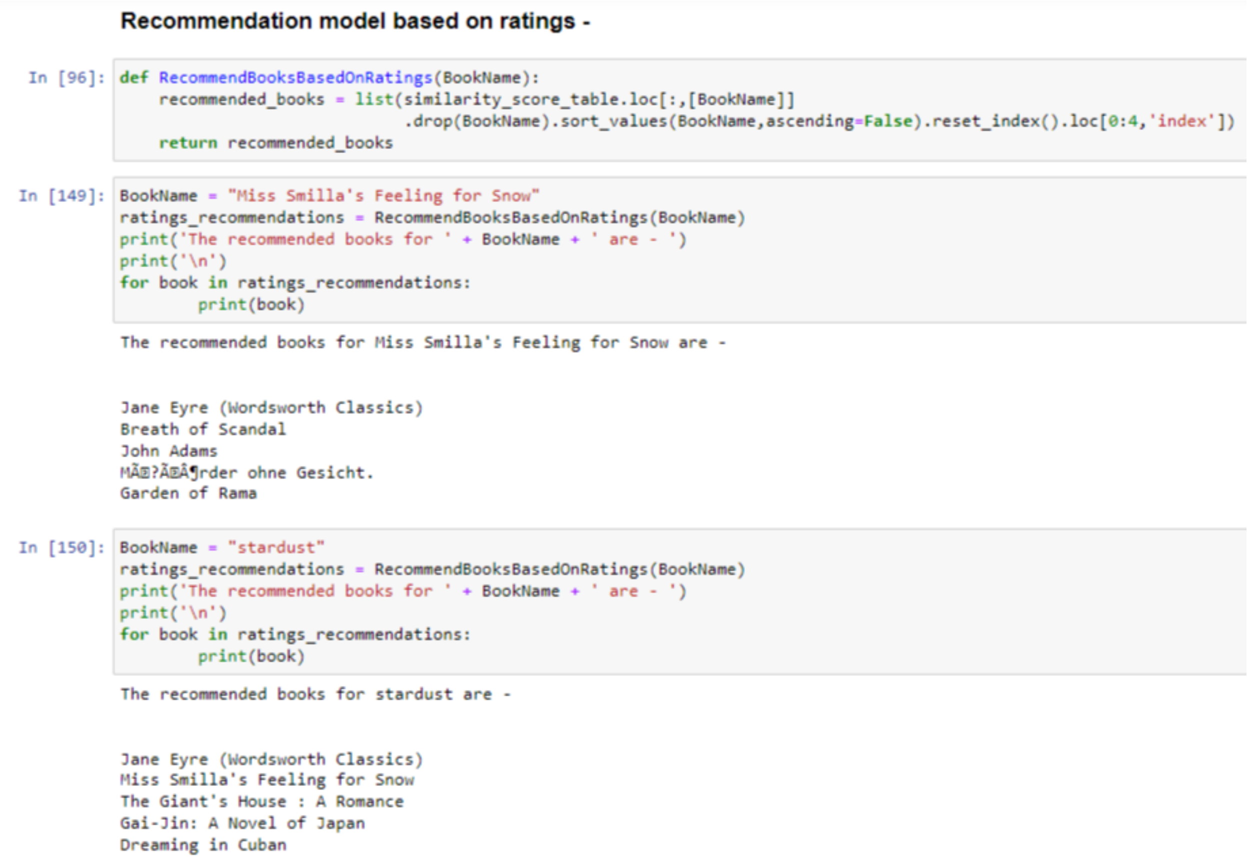
```
The number of Good Books in our dataset are - 3679
The sample from Good Books is - 

["'Salem's Lot",
 '10 Lb. Penalty',
 '101 Dalmatians',
 '16 Lighthouse Road',
 '1984',
 '1st to Die: A Novel',
 '2010: Odyssey Two',
 '204 Rosewood Lane',
 '2061: Odyssey Three',
 '24 Hours',
 '2nd Chance',
 '3001: The Final Odyssey',
 '311 Pelican Court',
 '3rd Degree',
 '4 Blondes']
```



```
good_readers = pd.DataFrame(books_ratings_merged['User-ID'].value_counts())[(books_ratings_merged['User-ID'].value_counts() > 10
good_readers = list(good_readers)
print('The number of Good Readers in our dataset are - ' + str(len(good_readers)))
print('The sample from Good Readers is - ')
good_readers[0:15]
```

```
The number of Good Readers in our dataset are - 1648
The sample from Good Readers is - 

[11676,
 198711,
 153662,
 98391,
 35859,
 212898,
 278418,
 76352,
 110973,
 235105,
 16795,
 230522,
 234623,
 204864,
 36836]
```

Final Recommendation Screenshot



Recommendation model based on ratings -

```
In [96]: def RecommendBooksBasedOnRatings(BookName):
             recommended_books = list(similarity_score_table.loc[:,[BookName]]
                                      .drop(BookName).sort_values(BookName,ascending=False).reset_index().loc[0:4,'index'])
             return recommended_books
```

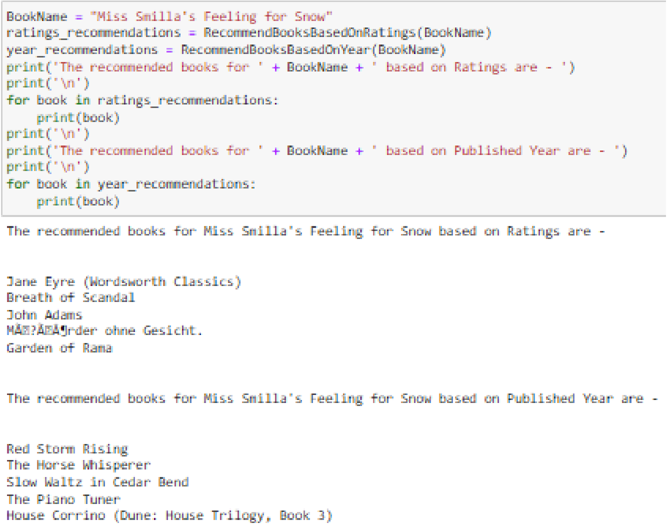
```
In [149]: BookName = "Miss Smilla's Feeling for Snow"
          ratings_recommendations = RecommendBooksBasedOnRatings(BookName)
          print('The recommended books for ' + BookName + ' are - ')
          print('\n')
          for book in ratings_recommendations:
                  print(book)
```

```
The recommended books for Miss Smilla's Feeling for Snow are -


Jane Eyre (Wordsworth Classics)
Breath of Scandal
John Adams
MÃ□?Ã□Â¶rder ohne Gesicht.
Garden of Rama
```

```
In [150]: BookName = "stardust"
          ratings_recommendations = RecommendBooksBasedOnRatings(BookName)
          print('The recommended books for ' + BookName + ' are - ')
          print('\n')
          for book in ratings_recommendations:
                  print(book)
```

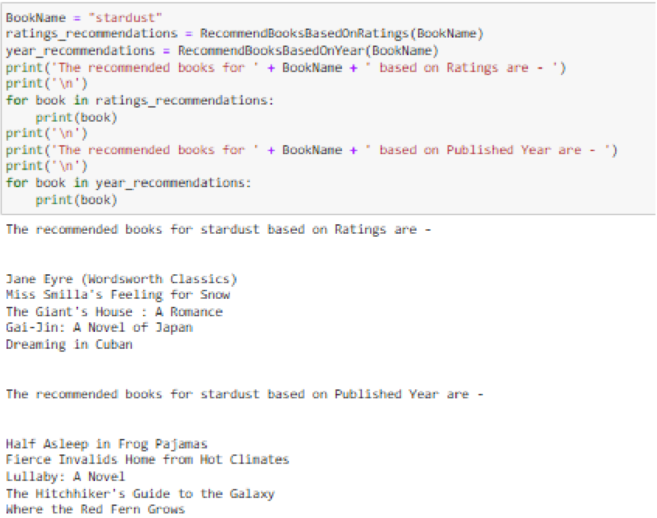
```
The recommended books for stardust are -


Jane Eyre (Wordsworth Classics)
Miss Smilla's Feeling for Snow
The Giant's House : A Romance
Gai-Jin: A Novel of Japan
Dreaming in Cuban
```

Recommendation Model inculcating Publication Year as additional dimension



```
BookName = "Miss Smilla's Feeling for Snow"
ratings_recommendations = RecommendBooksBasedOnRatings(BookName)
year_recommendations = RecommendBooksBasedOnYear(BookName)
print('The recommended books for ' + BookName + ' based on Ratings are - ')
print('\n')
for book in ratings_recommendations:
    print(book)
print('\n')
print('The recommended books for ' + BookName + ' based on Published Year are - ')
print('\n')
for book in year_recommendations:
    print(book)
```

```
The recommended books for Miss Smilla's Feeling for Snow based on Ratings are -


Jane Eyre (Wordsworth Classics)
Breath of Scandal
John Adams
MÃ□?Ã□Â¶rder ohne Gesicht.
Garden of Rama


The recommended books for Miss Smilla's Feeling for Snow based on Published Year are -


Red Storm Rising
The Horse Whisperer
Slow Waltz in Cedar Bend
The Piano Tuner
House Corrino (Dune: House Trilogy, Book 3)
```



```
BookName = "stardust"
ratings_recommendations = RecommendBooksBasedOnRatings(BookName)
year_recommendations = RecommendBooksBasedOnYear(BookName)
print('The recommended books for ' + BookName + ' based on Ratings are - ')
print('\n')
for book in ratings_recommendations:
    print(book)
print('\n')
print('The recommended books for ' + BookName + ' based on Published Year are - ')
print('\n')
for book in year_recommendations:
    print(book)
```

```
The recommended books for stardust based on Ratings are -


Jane Eyre (Wordsworth Classics)
Miss Smilla's Feeling for Snow
The Giant's House : A Romance
Gai-Jin: A Novel of Japan
Dreaming in Cuban


The recommended books for stardust based on Published Year are -


Half Asleep in Frog Pajamas
Fierce Invalids Home from Hot Climates
Lullaby: A Novel
The Hitchhiker's Guide to the Galaxy
Where the Red Fern Grows
```