CSE508 - Information Retrieval Project
Book Recommendation System

Shivanshu Kumar
2019476

Karan Prasad Gupta
2020439

Rishav
2020569

Sarthak Dixit
2020574

D Likhith
2019038

Harsh Kashyap
2020434

1. Introduction

- We're developing a cutting-edge book recommendation system that is designed to address common user challenges and elevate their reading experience.
- Our system tackles the issue of information overload by intelligently combining multiple models to maximize accuracy and deliver highly relevant results tailored to the users' interests.
- With a focus on user engagement, our personalized recommender system delivers tailored book suggestions consistently, ensuring users remain captivated and invested in their reading journey. By implementing this system on large platforms, we aim to significantly impact user retention, satisfaction, and revenue generation.
- Our innovative approach provides a competitive advantage by simplifying content discovery and aligning recommendations precisely with user preferences. We prioritize ease of access, ensuring that users can effortlessly discover books that cater to their individual needs and interests.

2. Literature Review

1. https://ieeexplore.ieee.org/document/9579647

The authors discover methods including machine learning techniques like the K-nearest neighbors, Pearson's R Correlation Coefficient, Cosine Similarity through collaborative filtering, to be efficient in deciding the best books for the user based on the query that is provided as the input into their system.

2. https://cs.carleton.edu/cs_comps/1617/book_rec/final-results/paper.pdf

The project on book recommendation system works on the data collected from across the web (Amazon Books, GoogleReads, etc.) and makes use of the Machine Learning classifier models such as Naive Bayes Classifier and the Maximum Entropy Classifier for their Content based approach and the K-nearest neighbor and UV decomposition for their Collaborative filtering based approach for determination of appropriate books according to the user query.

3. Book Recommendation System using Association Rule Mining & Collaborative Filtering

The research paper discusses several Collaborative filtering algorithms like the Jaccard Distance and Pearson' Coefficient along with a novel technique known as Association Mining.

Among all the above mentioned literature, we came to the conclusion that our information retrieval system would make use of parts of each of them, like Cosine Similarity, few of the Machine Learning

approaches like Clustering (or K Nearest neighbors) and Pearson's coefficient, and all of the suggested evaluation methods would be considered in deciding which model works the best, which finally would be allowed in the model combination process where we integrate various models and their features.

3. Methodology

The model that we intend to build is an amalgam of various types of Recommender Systems, It's a hybrid recommender system that mainly includes the functionalities of Popularity based, Content-based and Collaborative-filtering based recommender systems. The overall idea of the approach is to find out the best of the features from the processed datasets (Books, Ratings and Users), performing matrix factorization and using various techniques including Cosine Similarity, Pearson's Coefficient and K-nearest neighbor algorithm.

4. Baseline Results

Data Overview:
Users Table:
The Users table has info about users like User-ID, name etc.
Ratings Table:
The Ratings table has user ratings for various books, identified by their ISBN.
Books Table:
The Books table has details about books, such as Book-Title, Book-Author, and Image-URL-M.

Data Cleaning:

We checked for missing values in the Books, Users, and Ratings tables. No missing values were found.

Data Processing:

- We then merged the Ratings table with the Books table based on the common column 'ISBN.'
- We calculated the number of ratings and the average rating for each book.
- Merged the number of ratings and average ratings dataframes.
- Filtered and selected the top 50 books with a minimum of 250 ratings.

1. Collaborative Filtering:

- Identified users who rated more than 200 books.
- Filtered and selected books with more than 50 ratings.
- Created a user-item matrix for collaborative filtering.

Similarity Techniques:
Cosine Similarity:
We computed the cosine similarity between every pair of books and then recommended the top 5 similar books based on cosine similarity.

K-Nearest Neighbors (KNN):
We used the Nearest Neighbors algorithm to find similar books and again recommended the top 5 similar books using KNN.

Pearson's Coefficient:
Calculated Pearson correlation coefficients between books and based on that recommended 5 books.

2. Content-Based Recommendation(using Word2Vec model):

We preprocessed the text data in the Books data frame by lowercasing, punctuation removal, stop-word removal, and stemming. We then concatenated the 'Book-Title,' 'Book-Author,' and 'Publisher' columns into a single text for each book in the Books dataframe. We then utilized the Word2Vec model to learn vector representations of words in the combined text data.

Matrix Factorization

User-ID / Book-Title	254	2276	2766	2977	3363	4017	4385	6251	6323	6543	...	271705	273979	274004	274061	274301	274308	275970	277427	277639	278418
1984	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	10.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1st to Die: A Novel	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2nd Chance	0.0	10.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4 Blondes	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A Bend in the Road	0.0	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Year of Wonders	0.0	0.0	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	9.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
You Belong To Me	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Zen and the Art of Motorcycle Maintenance: An Inquiry into Values	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	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	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	0.0	0.0	0.0	8.0	0.0	0.0	0.0	0.0	0.0

706 rows × 810 columns

Collaborative filtering-based methods:-

1. Cosine Similarity:

similarity scores:

```
similarity_scores
```

```
array([[1.        , 0.10255025, 0.01220856, ..., 0.12110367, 0.07347567,
        0.04316046],
       [0.10255025, 1.        , 0.2364573 , ..., 0.07446129, 0.16773875,
        0.14263397],
       [0.01220856, 0.2364573 , 1.        , ..., 0.04558758, 0.04938579,
        0.10796119],
       ...,
       [0.12110367, 0.07446129, 0.04558758, ..., 1.        , 0.07085128,
        0.0196177 ],
       [0.07347567, 0.16773875, 0.04938579, ..., 0.07085128, 1.        ,
        0.10602962],
       [0.04316046, 0.14263397, 0.10796119, ..., 0.0196177 , 0.10602962,
        1.        ]])
```

```
[ ] similarity_scores.shape
```

```
(706, 706)
```

Recommendation:

```
recommend_through_cs('1984')
```

```
[['Animal Farm',
  'George Orwell',
  'http://images.amazon.com/images/P/0451526341.01.MZZZZZZZ.jpg'],
 ["The Handmaid's Tale",
  'Margaret Atwood',
  'http://images.amazon.com/images/P/0449212602.01.MZZZZZZZ.jpg'],
 ['Brave New World',
  'Aldous Huxley',
  'http://images.amazon.com/images/P/0060809833.01.MZZZZZZZ.jpg'],
 ['The Vampire Lestat (Vampire Chronicles, Book II)',
  'ANNE RICE',
  'http://images.amazon.com/images/P/0345313860.01.MZZZZZZZ.jpg']]
```

2. K-Nearest Neighbors

Apply KNN, tryout for one book position

> Parameters: Row from Pivot Table, Num_of_Neighbors

```
[ ] distances, suggestions = model.kneighbors(pt.iloc[237, :].values.reshape(1,-1), n_neighbors=6)
```

Distances (Similarity -> Highest to Lowest)

```
[ ] distances
```

```
array([[ 0.        , 29.30870178, 30.01666204, 30.41381265, 30.47950131,
        30.87069808]])
```

Recommendation:

```
[ ] recommend_through_knn('Animal Farm')
```

```
Index(['Animal Farm', 'Exclusive', 'Hearts in Atlantis', 'Jacob Have I Loved',
       'Second Nature', 'Pleading Guilty'],
      dtype='object', name='Book-Title')
```

3. Pearson's coefficient

Recommendation:

```
recommend_through_pc('Animal Farm')
```

```
Similar books:
1. Book name: 1984, Similarity: 0.24863430168716266
2. Book name: Angus, Thongs and Full-Frontal Snogging: Confessions of Georgia Nicolson, Similarity: 0.2226534505381867
3. Book name: Midnight, Similarity: 0.21992580013893603
4. Book name: Second Nature, Similarity: 0.20647606147712394
5. Book name: Call of the Wild, Similarity: 0.19336425011186748
```