

COMP20008 Assignment 2

Executive Summary

Recommender systems have become an integral part of our daily lives. These systems are crucial in reducing the transaction costs associated with finding and selecting items. The two most common types of recommender systems are Content-Based and Collaborative Filtering. Content-based systems rely on item features to recommend similar items to what the user has already liked. However, collaborative filtering methods are more commonly used because they can provide better results and are straightforward to implement. This report will explore the models by analysing datasets from a Bookstore.

Introduction

Our research focuses on addressing the following research question:

1. Can the datasets be used to implement a recommendation system?
2. How content-based and collaborative filtering methods are used in recommendation systems?

The datasets that we use for analysis contain information on books (their author, ISBN, title, year of publication, and publisher), users (their ID, state, country, age), and ratings that users give books (user ID, book ISBN, and rating). We deduced that the feedback about a book falls under the explicit category, meaning users specify how much they liked a particular movie by providing a numerical rating. Based on the given datasets, our group conducted research to create 2 recommendation systems: content-based and collaborative filtering using matrix factorization.

Methodology

1. Data Preprocessing

1.1 The Users dataset (BX-Users.csv)

1.1.1 Handle missing data

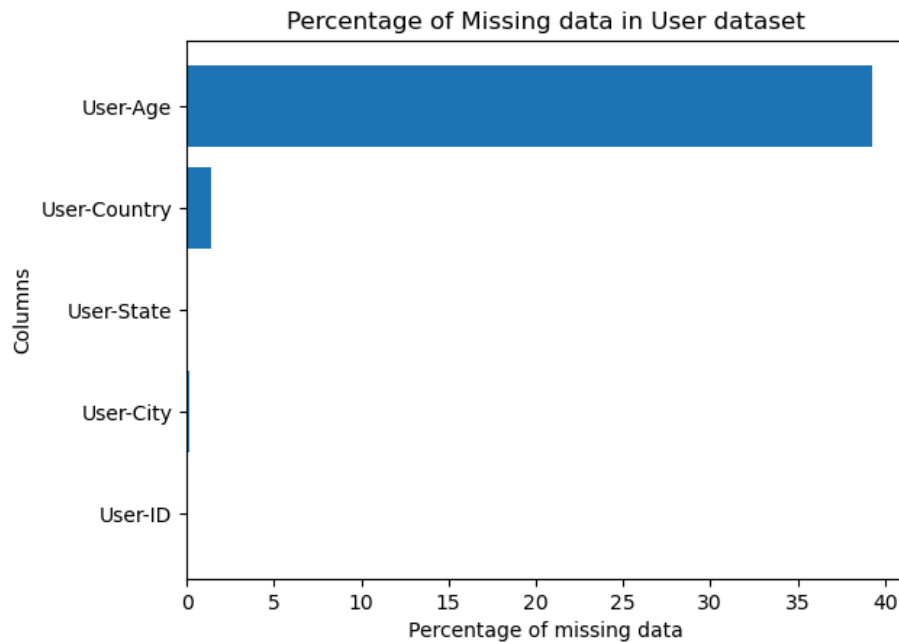


Figure 1. Percentage of missing data

As observed from the patterns of the missing data in the user database (Figure 1), 'User-Age' contains the largest number of missing values, nearly 40%, while other variables are significantly smaller in comparison. This suggests that 'User-Age' is Missing at Completely Random. Due to the large number of missing data, 'User-Age' is dropped from the dataset. The missing values in 'User-Country' are less than 5%, which is a low amount, and is filled with "other" as the placeholder.

1.1.2 Standardise 'User-Country'

This field has several noises, mostly quotation marks at the end of the field. This could stem from the parsing error while extracting the data. To fix the problem, we used REGEX to identify the pattern and removed the noises at the end of the data. Moreover, since the country's field has many derivations of the same country's name, we used PyCountry to standardise the name. Last but not least, the entries whose length is less than 3 and countries in which PyCountry cannot be found are invalidated, and replaced with Other.

1.2 The book dataset (BX-Books.csv)

1.2.1 Text Processing books' titles and author names

The provided code utilises the Natural Language Toolkit (NLTK) library to preprocess textual data effectively. The function encompasses key preprocessing steps, including case folding, removal of non-alphabetic characters, tokenization, elimination of stopwords, and lemmatization.

1.2.2 Discretize books' publication year

The code counts the occurrences of each unique year of publication and filters the dataset to include only entries with publication years up to 2024. Subsequently, it discretizes the years of publication into categorical groups based on predefined ranges, such as "Before 1950," "1950-1960," and so forth. Finally, the code calculates the frequency distribution of these discrete categories, offering insights into the distribution of publication years across different periods.

2. Recommendation system model

2.1 Content-Based method

Content-based approach utilises term frequency-inverse term frequency(TF-IDF), along with cosine similarities to compute the similarities between books within the dataset. This enables personalised recommendations based on the textual title of the books and users' preferences

2.2 Collaborative filtering method

Collaborative Filtering method uses Singular Value Decomposition (SVD) to extract latent factors from user-book interaction matrix, combined with k-fold cross-validation (with $k = 5$) to evaluate the predictions. Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) were utilised as loss functions, model optimization was achieved through stochastic gradient descent.

Data Exploration and Analysis

1. The user dataset:

1.1 Distribution of users' countries

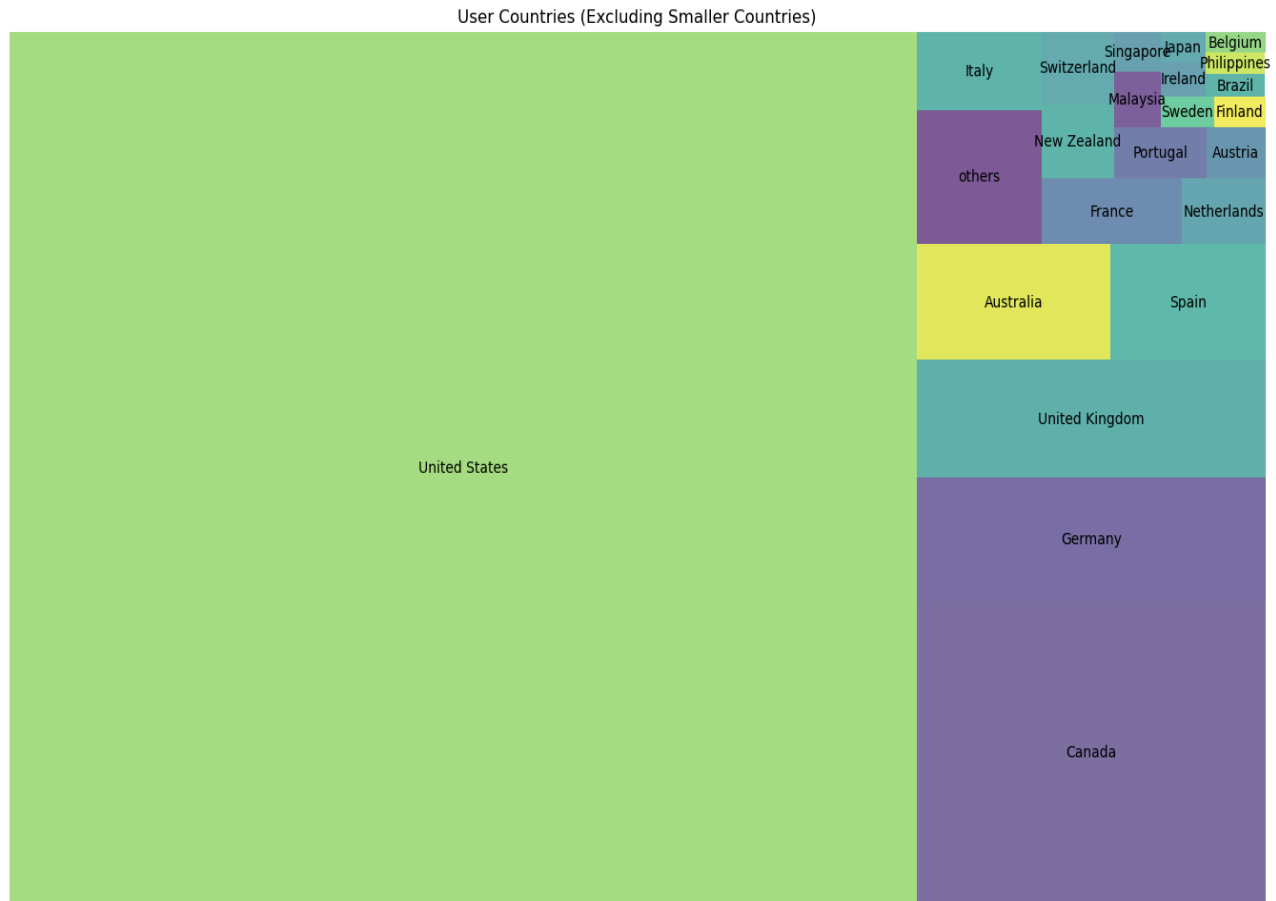


Figure 2. Countries distribution among users

Figure 2 shows the distribution of users by country. We use different colours to represent different countries. The United States has the largest proportion of users, it accounts for more than half of the total. The significant advantage of the United States in the distribution of book users is often due to the combination of multiple factors. First of all, as the core of the English-speaking world, American culture and language have a profound influence, which makes English readers more inclined to participate in online book-reading activities. Secondly, the United States has one of the largest book markets in the world, and its huge readership and rich book resources attract readers from all over the world to participate in grading and reviewing. With the popularity of the web and social media, the advanced network infrastructure in the United States has made it easier for American readers to participate more in reading and reviewing books online. Finally, the developed publishing industry chain in the United States, including publishers, retailers, and distribution channels, provides a solid foundation for the wide dissemination and reading of books, thus further enhancing the participation and influence of American readers.

2. Rating Dataset

2.1 Distribution of Book Rating

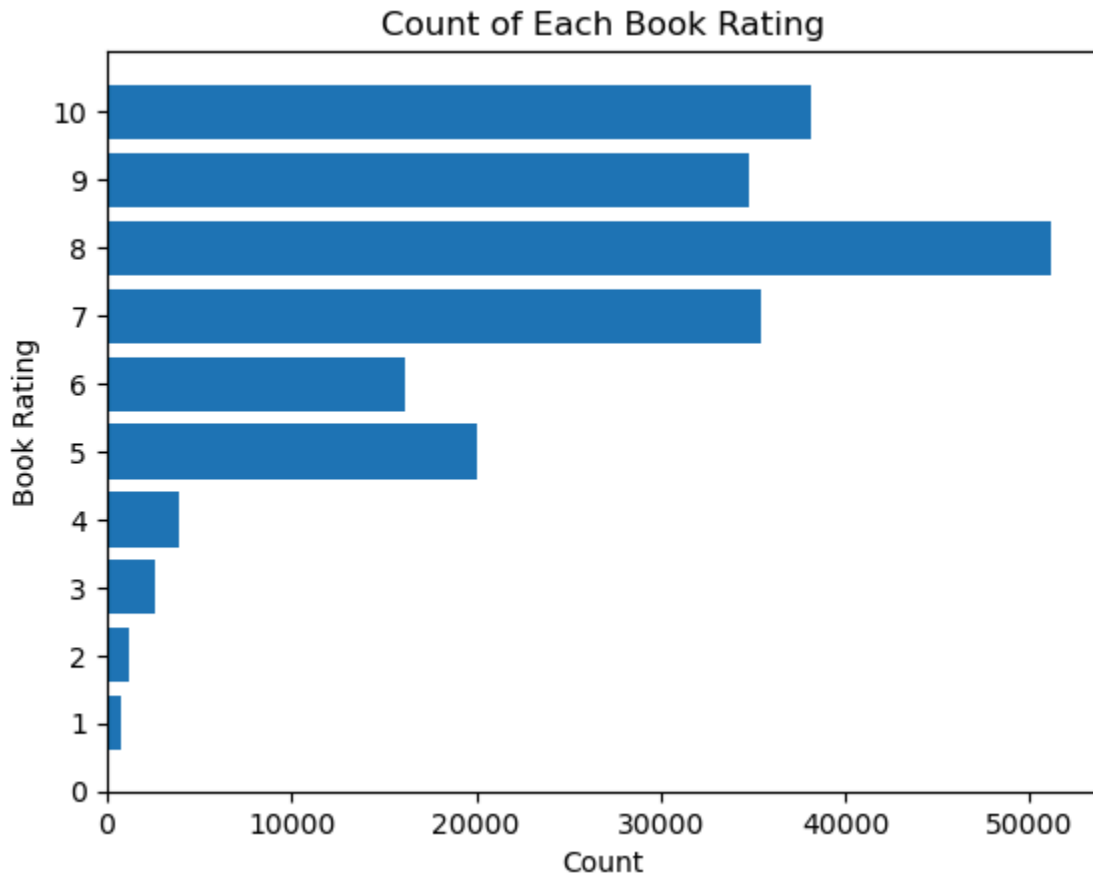


Figure 3. Rating counts

We use a bar chart to show the distribution of users' book ratings. In figure 3, we can clearly see that the vast majority of users rated the book at 8 points, and almost all users rated the book at more than 5 points. The reason why most users give the book a high score is probably because they have positive comments and recognition of the content, style, or other aspects of the book. At the same time, users will filter according to the aspects they are interested in before reading the book, so it is unlikely that they will completely dislike the book. The fact that most people score eight points suggests the book is excellent in some aspects, but there are also some shortcomings, leading some readers to slightly reserve its score, and eight points have become a consensus evaluation of the comprehensive performance of the book by most readers.

3. Book Dataset

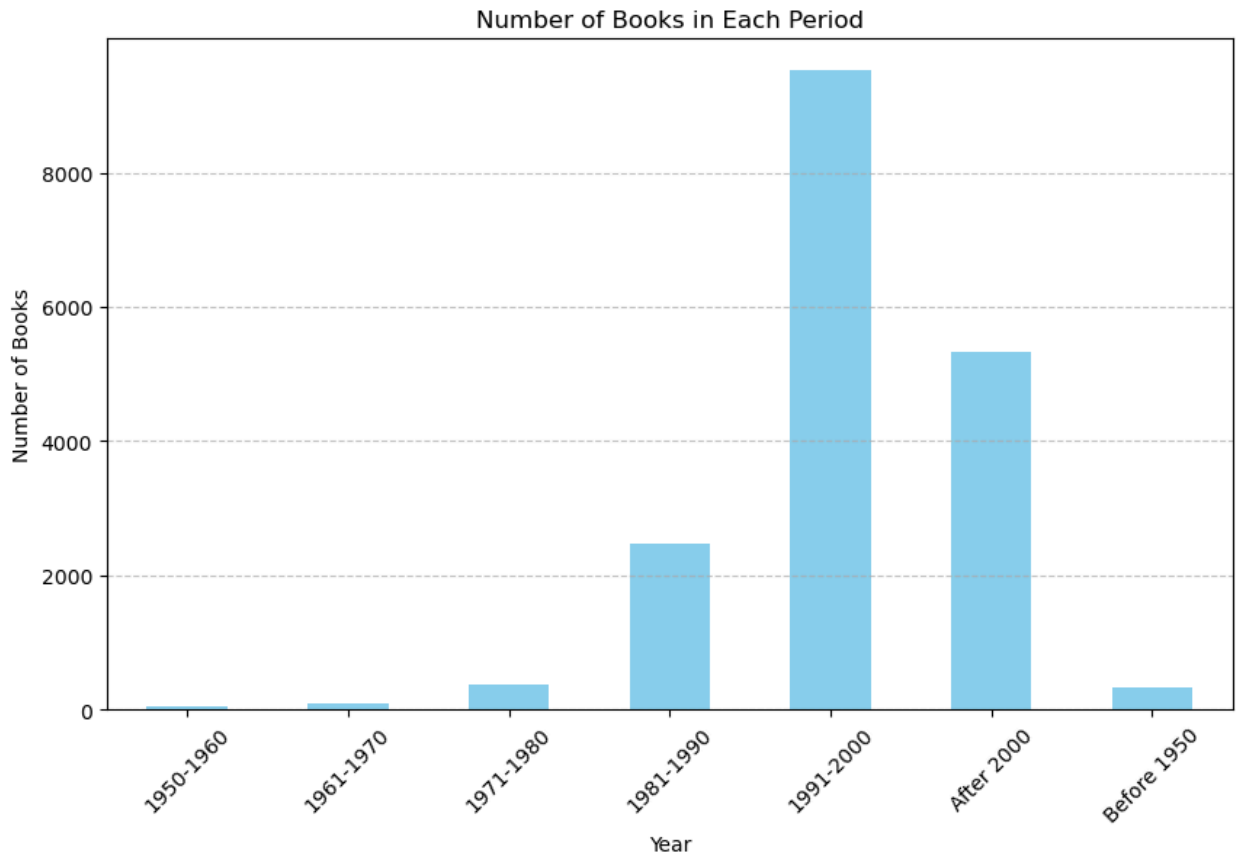
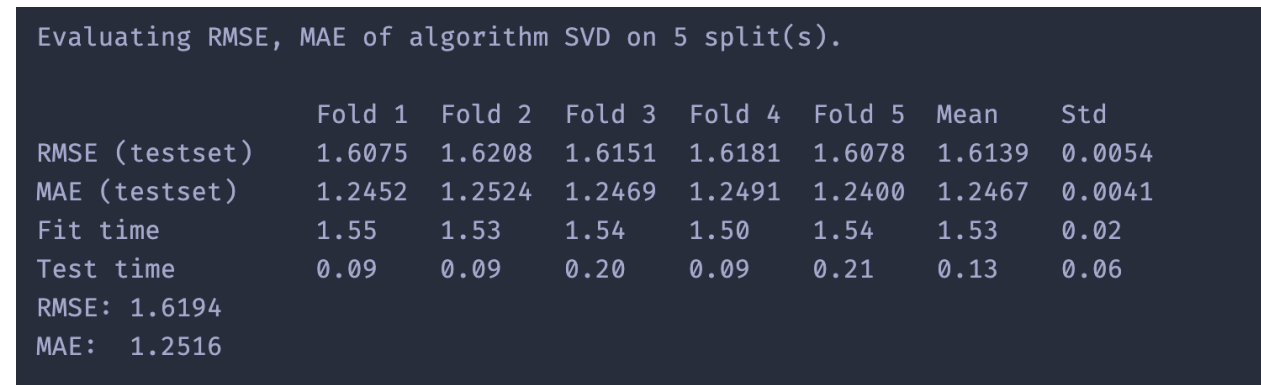


Figure 4. Number of books in its according periods

In terms of the number of books in each period, we select a histogram to represent the state of book publishing over a period of time (Figure 4). A considerable number of these books were published between 1991 and 2000, and after 1980, the number of books published increased dramatically, probably because the popularisation of digital technology made the process of book production and publication unprecedentedly convenient and efficient. From editing to typesetting to printing, a series of links accelerated to promote the surge in the number of books. At the same time, the expanding market demand, especially as people's pursuit of knowledge and entertainment becomes more intense, has prompted publishers to increase their publishing efforts.

Results

1. Collaborative filtering recommendation system model



	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean	Std
RMSE (testset)	1.6075	1.6208	1.6151	1.6181	1.6078	1.6139	0.0054
MAE (testset)	1.2452	1.2524	1.2469	1.2491	1.2400	1.2467	0.0041
Fit time	1.55	1.53	1.54	1.50	1.54	1.53	0.02
Test time	0.09	0.09	0.20	0.09	0.21	0.13	0.06
RMSE: 1.6194							
MAE: 1.2516							

Figure 5. Evaluation of SVD Algorithm

Evaluating the Singular Value Decomposition (SVD) algorithm on a dataset split into 5 folds reveals insights into the performance of the recommendation system. Across the folds, the algorithm demonstrates consistent rating predictions, as indicated by the small deviation between the RMSE and MAE. The average RMSE is 1.6139, with individual folds ranging from 1.6075 to 1.6208, while the average MAE is 1.2467, with fold values ranging from 1.2400 to 1.2524. This means on average, the model's prediction has an error rate of 1.6139 points out of 10 (by RMSE) or 1.2467 points (by MAE) (Figure 5). This suggests that the SVD model provides reasonably accurate predictions of user-item interactions. Additionally, the fitting time for the SVD algorithm ranges from 1.50 to 1.55 seconds, with an average of 1.53 seconds, indicating efficient model training. The testing time varies more, with an average of 0.13 seconds across folds, suggesting reasonable efficiency in generating recommendations. After inspecting the results from both of our models, we can address our 2 research questions. The Bookstore dataset can be used to implement a collaborative filtering algorithm using a matrix factorization recommendation system. The collaborative filtering model uses a user-item interactions matrix to identify patterns and similarities between users or items. By leveraging the similarities of users' preferences, collaborative filtering can provide personalised recommendations even in the absence of explicit item attributes or user profiles (More explanations are given under the Discussion and Interpretation section).

2. Content-based recommendation systems

Since collaborative filtering recommendation systems require data from users to train on and recommend to users, it raises a question: When the systems do not have many user interactions or new books are introduced, how can users get recommendations that incorporate those new data? To approach this problem, we decided to create another model that uses the book's title. After evaluation and testing, we concluded that the content-based recommendation systems using TF-IDF on the book's title failed to recommend books that actually interest the users. This is because the similarity matrix between titles consists most of 0, which is no correlation between them.

Discussion and Interpretation

1. Collaborative filtering recommendation system model

	test_rmse	fit_time	test_time
Algorithm			
SVD	1.614880	1.629364	0.131079
KNNWithMeans	1.794819	4.046509	1.331131
KNNWithZScore	1.800592	4.311492	1.325727
NMF	2.526160	3.987020	0.129513

Figure 6. Other methods performance

A few methods were trialled to find the best performance model to train the datasets. With kNN approaches, including kNN with means and kNN with z-score normalisation, the obtained Root Mean Square Error (RMSE) value is around 1.80. In contrast, the SVD model achieved a notably lower RMSE of approximately 1.61. This might be because kNN approaches rely on the similarity between users or items, making them susceptible to sparsity when there are insufficient overlapping interactions. In highly sparse interaction matrices like in the Bookstore dataset, this can lead to limited neighbourhood selection and lower-quality recommendations. As KNN method requires computing pairwise similarities between users or items, the computational time is 4 times higher than that of SVD, which scales reasonably well with medium size datasets due to its optimization technique. On the other hand, the test RMSE for NMF is 2.52, indicating a higher error compared to SVD. SVD typically achieves lower RMSE values due to its ability to capture both positive and negative user-item interactions, which may contain valuable information for recommendation accuracy. On the other hand, NMF enforces non-negativity constraints on the learned factors during the factorization process, which may limit its ability to model negative interactions effectively and lead to higher prediction errors.

1.1. Can the datasets be used to implement a recommendation system?

The model accurately predicts user-item interactions with low RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error) values across 5 folds of cross-validation. This indicates that the datasets contain sufficient information to capture the underlying patterns and preferences of users. Moreover, it consistently performs well across different splits of the data, which reinforces its robustness and suggests that it can generalise well to unseen data. The relatively low computational time required for model fitting and prediction tasks underscores the algorithm's efficiency. This makes it possible to scale up for larger datasets or real-time recommendation applications.

1.2. How is SVD used in recommendation systems?

Due to the nature of the Bookstore datasets, which contain explicit feedback data, along with having limited features that may not support a content-based approach/ regression approach, Single Value Decomposition (SVD) is what we opted to use. SVD excels in extracting latent factors from sparse user-item interaction matrices, enabling it to uncover underlying patterns in user preferences and item characteristics even when explicit ratings are limited. By reducing the dimensionality of the dataset and capturing latent factors, SVD can effectively address the sparsity issue inherent in recommendation datasets, providing accurate and personalised recommendations based on user-item interactions. The method decomposes the sparse user-book interaction matrix into three matrices: U , Σ , and V^T . U represents the users and their book ratings, with each row corresponding to a user and each column to a latent feature capturing the book ratings. The diagonal matrix Σ contains singular values, signifying the

importance of each latent feature. V^T represents the books and their characteristics, with each row representing a book and each column representing a latent feature indicating book characteristics. By factorising the matrices, SVD extracts latent features shared by users and books, thereby capturing underlying patterns in the data. Predictions for user-book interactions are generated by computing the dot product of the corresponding user and book vectors. The predicted scores generated by the model serve as a basis for recommendation generation. Specifically, books with the highest predicted scores, which the user has not yet explored, are prioritised for the recommendation.

2. Content-based:

2.1. Can the datasets be used to implement a recommendation system?

Upon investigating the failure to recommend books for users, we see that the titles similarity matrix using cosine similarity mostly consists of zero values. This observation suggests that there is a lack of correlations between the majority of the book titles. This could be explained by the limitation of words in the titles. Many book titles consist of only a few words, which restricts the potential for meaningful semantic overlap and similarity calculation. Additionally, the words used in book titles often vary widely in topics and themes. Consequently, the titles have diverse linguistic characteristics, making it challenging to establish connections between them. As a result, the calculated similarity score between titles tends to be nonexistent, with many pairwise comparisons resulting in zero similarity scores. Addressing this challenge may need exploring more features for the dataset or more suitable similarity measurements that better capture the book content, thereby enhancing the recommendation system's effectiveness.

2.2. How is TF-IDF used in a content-based recommendation system?

The recommendation system operates by assigning weights to the terms present in the book title. The weights are determined by analyzing the frequency of each term within the item description and its rarity across the entire set of books. When a user expresses interest in an item, the system retrieves other items with similar TF-IDF vectors, indicating a similarity in content. Essentially, the system uses the textual features of items to deliver personalized recommendations that match the user's preferences. This approach is particularly beneficial because it is content-based, meaning that it recommends items based on their textual similarity to other items rather than relying on user preferences or past behavior.

Limitations and improvement opportunities

Limitation of dataset:

The Book Store Dataset exhibits several limitations that impact its utility and scope for analysis. Firstly, the missing data poses a significant issue, particularly regarding the exploration of age-group demographics. The absence of age information deprives valuable insights into user age distributions and related patterns, which could potentially inform us more of users' preferences. Secondly, the dataset's feature set is constrained by its limited attributes. This not only impacts the performance of the recommendation systems model (see limitations of the content-based model) but also limits us from exploring discernible correlations between variables.

Limitation of Collaborative Filtering using SVD

While Collaborative Filtering using SVD produces a reasonable performance result, it should be taken into consideration of two problems.

The first drawback pertains to the model's inability to recommend new books to users ("cold-start problem"). This limitation arises because SVD relies solely on historical user-book interactions to generate recommendations. When new books are introduced into the system, they lack the knowledge of interaction history for the model to make informed recommendations. Consequently, until these new books accumulate sufficient interaction data, they remain excluded from the recommendation process, potentially leading to missed opportunities for personalised recommendations.

The second drawback concerns the neglect of user and book features in the recommendation model. SVD primarily focuses on capturing latent factors from user-book interactions, disregarding additional contextual information such as user demographics (e.g., age, country) or book attributes (e.g., title, author). Incorporating these side features could enhance the model's accuracy as it captures user preferences and item characteristics more comprehensively.

Another limitation that should be considered is how computationally expensive using SVD is. SVD involves decomposing the user-item interaction matrix into lower-dimensional matrices, a process that requires computing singular value decompositions and storing the entire matrix in memory. This computational complexity results in longer training times, increased memory requirements, and scalability issues, making it challenging to apply SVD to massive datasets efficiently. To mitigate these limitations approaches such as sampling can be employed.

Limitations of the Content-based model

The limitations of the dataset significantly impact the effectiveness of the content-based filtering model employed for book recommendations. Primarily, the model's reliance on word weight(TF-IDF) within book titles proves inadequate due to the dataset's inherent limitations.

With short titles and limited common words among them, the model struggles to find similarities, thereby impairing its ability to provide precise recommendations to users. Furthermore, the current model's restriction to English words and exclusion of non-English books from analysis and recommendation deprives users of diverse literary options but also disadvantages those interested in non-English literature due to the dataset's broken encoding for non-English titles.

Consequently, users with preferences for non-English books are not recommended their potential preferred books by the recommendation system, highlighting a significant shortfall in its inclusivity and scope. Moreover, the model's reliance solely on users' existing interests for recommendations further restricts its capacity to broaden users' literary horizons. By solely leveraging past interactions, the model overlooks opportunities to introduce users to new genres or authors, limiting its potential to foster exploration and discovery.

Conclusion

Our results have demonstrated that the **Collaborative Filtering Recommendation System using Singular Value Decomposition (SVD)** performed the best in our selections of models. With the model, it can reasonably predict the books that users are interested in.

The Content-based filtering Recommendation System using Terms Frequency - Inverse Document Frequency (TF-IDF) struggled to recommend books that users were interested in. However, it could be further improved with more text features such as an overview, genre, or content preview.

Reference

- Google. (2022, July 18). *Collaborative Filtering Advantages & disadvantages | machine learning | google for developers*. Google.
<https://developers.google.com/machine-learning/recommendation/collaborative/summary>
- Li, S. (2019, September 26). *Building and testing recommender systems with surprise, step-by-step*. Medium.
<https://towardsdatascience.com/building-and-testing-recommender-systems-with-surprise-step-by-step-d4ba702ef80b>
- Shah, S. (2021). *Introduction to Matrix Factorization for Recommender Systems*.
<https://doi.org/10.31219/osf.io/pnd5w>