INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 10, Issue 6 - V10I6-1523)
Available online at: https://www.ijariit.com

Machine Learning-Based Collaborative Filtering Book Recommendation System

Vatsalya Maddu
vatsalyamaddu2003@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

P.Jusmitha
pathuri.jusmitha@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

S.Lilly
seelaboinalilly@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

K.Sai Sri
kilimisaisree@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

D.Janvitha Padma
damisettijanvithapadma@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

T.Devika
devikathokala@gmail.com
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

N.V. Muralikrishna Raja
muralikrishna1042@srivasaviengg.ac.in
Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh

ABSTRACT

The project aims to develop a book recommendation system tailored to support and inspire individuals with an interest in reading. Leveraging a collaborative filtering approach that incorporates collaborative filtering based on K-nearest neighbors (KNN) the system identifies similarities among users or items based on their book interactions and book ratings. Through meticulous dataset preprocessing, including feature extraction of genre, author, and user preferences, the system ensures high-quality recommendations. Evaluation metrics such as precision and recall gauge system performance, while a user-friendly interface provides easy access to personalized book suggestions. Continuous user feedback drives ongoing improvements, fostering a culture of reading discovery and habit cultivation. Ultimately, the deployment of this system aims to encourage individuals to explore new literary works and develop a lifelong passion for reading.

Keywords: *KNN, Collaborative Filtering, Book Recommendation system.*

INTRODUCTION

Rapidly expanding digital age, where information proliferates across the vast expanse of the Internet, the conventional methods of searching for relevant content have become increasingly challenging. This is where recommendation systems step in as indispensable tools, revolutionizing the way users access pertinent information by providing tailored suggestions. The present effort is primarily intended to support people who genuinely want to read and to assist people who are starting to establish a consistent study schedule. This initiative acknowledges the overwhelming abundance of books available on the market, recognizing the difficulty users face in selecting a specific book which matches their style and interests.

The fundamental purpose of recommendation systems lies in simplifying the user experience, allowing individuals to swiftly navigate the vast sea of data available and choose content that closely matches their interestand requirements. With the proliferation of digital content, users often find themselves inundated with choices, making it challenging to pinpoint the ideal book that resonates with their reading preferences. The recommendation framework within this project serves as a beacon of support, offering a personalized and curated list of books tailored to individual needs, thus alleviating the burden of decision-making.

This system integrates collaborative filtering. Collaborative filtering identifies patterns and preferences by considering the choices of similar users. Collaborative filtering, in particular, analyzes users' interactions with books to recommend items they may enjoy based on others with similar tastes. The merging for these techniques ensures that users receive suggestions based not only on their past interactions and preferences but also on the intrinsic features of the books they have enjoyed.

The crux of the project lies in its support for readers, both seasoned and those embarking on a literary journey. The recommendation system does not merely suggest books; it fosters a reading culture by presenting users with a diverse array of options that cater to their evolving interests. By encouraging users to explore new genres and authors, the system becomes a catalyst for expanding their literary horizons.

Additionally, the study highlights how important time-saving is. The system for suggesting products is made to effectively give consumers suggestions that suit what they want in a world where time is a valuable resource. By automating the procedure of discovering new and relevant books, users can dedicate more time to the actual act of reading, thereby fostering a habit that is both pleasurable and long-lasting.

RELATED WORKS

Throughout the previous Over the course of the last two decades, the field of recommendation systems has witnessed remarkable growth, driven by the pressing challenge of information overload faced by users in today's digital landscape. To effectively combat this issue, recommendation systems have evolved to propose personalized content and service suggestions using a wide range of sophisticated strategies and tactics. Recommendation frameworks various of approaches, frequently combining association rules, shared filtering, and filtering based on content to customize recommendations based on user preferences. Notably, a mobile application recommendation system has been introduced[14], emphasizing simplicity while preserving essential factors like rating, size, and permissions. Opinion mining and product feature extraction from Numerous studies are being conducted on reviews or opinions. The evaluations of several items by Morinaga et al. have been compared to ascertain the targetproduct's reputation; however, the reviewers' opinions have not been distilled from the reviews, nor has the product feature analysis taken into consideration[3].

Collaborative filtering, a key technique, involves constructing a dataset of user choices for items, taking into account unique attributes, and establishing similarities between user profiles[1],[3],[7]. Contrarily, content-driven recommendation uses user-provided information to create a comprehensive user profile, enabling the delivery of highly personalized suggestions. This picture is further enhanced by the extraction of association rules, which reveals interesting links among large item databases. But there is a significant drawback of previous recommendation systems lies in their narrow focus on existing user recommendations, often overlooking scenarios involving new users, the introduction of new books, or the provision of general recommendations.

In their research, Hu and Liu (2004) [8] compiled all of the customer reviews for a certain product and looked at its features to see if buyers had favorable or negative feedback. They have been working on extracting book features for recommendation in [1]. Our work and both of these are closely related [1][8]. Our method is different from theirs, though, in that we have customized a book's characteristics for a particular user. As part of our duties, we give the user a list of books with features that are similar to their own. has previously mentioned in reviews for other books. Consequently, we have tailored our work to each individual which provides recommendation systems with more relevant results.

Proposed System

The team's initiative, the "Book Recommendation System," aims to increase people's love of reading by assisting them in locating literature that speaks to their interests. To this end, we employ techniques such as KNN (K-Nearest Neighbors) to identify commonalities between different novels. The system supports three primary use cases: ratings for recently added books, suggestions for new users, and guidance for current users. Each use case is handled using a different approach. In order to forecast the interests of an individual according to ratings provided by people who have comparable tastes, the project primarily uses user-based collective filtering. The system also incorporates content-based filtering, which assesses user choices based on the particular content aspects of books. This dual strategy provides a more thorough and efficient way to filter content while also integrating user-based collaboration into the filtering process and accurate book recommendations. The workflow for making recommendations to an existing user is outlined in the flow chart, illustrating the systematic process implemented in our book recommendation system.

Datasets

In our dataset, we manage three tables: the "Book" dataset, whose 8 columns consist of ISBN, Book-Title, Book-Author, Year-of-Publication, Publisher, Image-Url-s, Image-Url-l, and Image-Url-m, the "Ratings" dataset consisting of 3 columns: User-ID, ISBN, Book-Rating, and the "Users" dataset with 3 columns: User-ID, location, and age. The initial phase involves data preprocessing, a pivotal method to transform raw data into a coherent format. During this phase, we merged datasets, eliminated records for which certain characteristics had values that were null, and removed columns with excessively missing values. Duplicate values were also identified and expunged from the dataset. Following the pre-processing, the dataset now features six columns: ISBN, Book Title, Book Author, Publisher, User ID, and Book Rating. This refined dataset lays the foundation for subsequent analyses.

Additionally, the second picture shows the rating frequency graph that we created to visually represent how books are distributed across different rating values ranging from 0 to 10. This graphical representation offers insights into the frequency of books receiving each rating, providing a comprehensive overview of the overall rating distribution within the dataset. The meticulous data pre-processing, coupled with the informative rating distribution graph, ensures that our dataset is well-prepared for further analysis, facilitating a clearer comprehension of the distribution of book ratings across the collection.

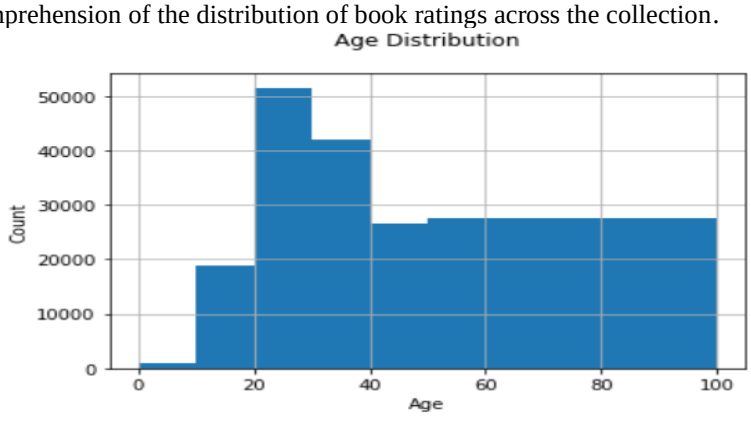

Fig. 1. Histogram

Algorithms

The k-Nearest Neighbor method:

This recommendation system uses a machine learning technique called KNN, or k-Nearest Neighbors, to find user categories that are related based on basic book assessments and to make suggestions derived from the mean ranking of the highest-ranking k closest neighbors. Using cosine similarity, this method finds comparable users in the beginning of the procedure.

This entails changing the rating values into a sparse matrix for more effective computations, converting the data table into a 2D matrix, and adding zeros to any missing values. Next, Using sklearn.neighbors unsupervised approaches, the Nearest Neighbors are determined. The rating matrices' cosine similarity is calculated by the method, and the result is a number between 0 and 1, where 0 denotes no relation and 1 represents 100% closeness.

The KNN suggestion is shown in Fig. 3, which illustrates the development of a collaborating filtering recommendation system. The five novels that are most similar to the selected book are listed by the algorithm using KNN in addition to the corresponding metrics. Since similarity is stated as a distance, a larger number denotes less similarity. The order of these novels is ascending. When a user selects their favorite book from the available data, the algorithm suggests the five titles that are most comparable to the selection. With the use of KNN, this method offers a personalized and effective recommendation system that makes relevant and accurate recommendations according to the interests of each unique user.

```
In [56]: # Now import our clustering algoritm which is Nearest Neighbors this is an unsupervised ml algo
         from sklearn.neighbors import NearestNeighbors
         model = NearestNeighbors(algorithm= 'brute')

In [57]: model.fit(book_sparse)

Out[57]: NearestNeighbors(algorithm='brute')

In [58]: distance, suggestion = model.kneighbors(book_pivot.iloc[237,:].values.reshape(1,-1), n_neighbors=6 )

In [59]: distance

Out[59]: array([[ 0.        , 68.78953409, 69.5413546 , 72.64296249, 76.83098333,
                 77.28518616]])

In [60]: suggestion

Out[60]: array([[237, 240, 238, 241, 184, 536]], dtype=int64)
```

Fig. 2. KNN Recommendation

Collaborative filtering Technique:

Collaborative filtering stands out as a pivotal approach to predicting book ratings for existing users by assessing similarities between users. The core concept involves identifying a cohort of users who share analogous preferences with a given user, achieved through the application of the K Nearest Neighbors algorithm. This algorithm was imported from the sklearn.neighbors library. In this context, the method looks for people who are similar by focusing on books that the user has already evaluated.

Three tasks are included in this implementation: finding the K closest neighbors, estimating the user's rating for a certain book, and suggesting the best books. The information is converted into an aggregate dataset, which is effectively a matrix with user IDs serving as the index, ISBNs as columns, and all zero ratings removed. The algorithm is then fed with the total number of books that each user has rated, as well as the user's distinct ID, ISBN, and ratings aggregating data.

Following the computation of anticipated ratings, the books are arranged in decreasing order, and the system suggests the top five books. If the estimated ratings are less than six, the user is prompted to look at broad suggestions based on Pearson correlation.This approach ensures users receive personalized recommendations based on collaborative filtering while offering a fallback option for general suggestions when necessary.

```
In [75]: trainset, testset = train_test_split(data, test_size=0.2)

         model = SVD(n_factors=80, n_epochs=20, lr_all=0.005, reg_all=0.2)
         model.fit(trainset)
         predictions = model.test(testset)

In [76]: df_pred = pd.DataFrame(predictions, columns=['user_id', 'isbn', 'actual_rating', 'pred_rating', 'details'])
         df_pred['impossible'] = df_pred['details'].apply(lambda x: x['was_impossible'])
         df_pred['pred_rating_round'] = df_pred['pred_rating'].round()
         df_pred['abs_err'] = abs(df_pred['pred_rating'] - df_pred['actual_rating'])
         df_pred.drop(['details'], axis=1, inplace=True)
         df_pred.sample(5)
```

Out[76]:

	user_id	isbn	actual_rating	pred_rating	impossible	pred_rating_round	abs_err
2293	174534	038531292X	10.0	7.634550	False	8.0	2.365450
17329	105517	0517556278	9.0	6.768408	False	7.0	2.231592
23549	171999	0446364193	5.0	7.677509	False	8.0	2.677509
21618	66680	0446521582	5.0	6.265271	False	6.0	1.265271
30683	258567	0553375407	8.0	8.403935	False	8.0	0.403935

Fig. 3. Collaborative Filtering output

Suggested Reading Books for New Users:

Nevertheless, the algorithm makes an effort to suggest books that are suitable for a beginner user, even if they don't show up in the dataset right away. In this instance, a list of the five books in the dataset with the highest average overall ratings is displayed to the viewer. The average ratings are determined by summing up all of the ratings from every book and dividing the total by the total number of ratings. After that, this list is sorted in descending order using the established averages as a reference.Once they have scored these recommended books, the new user's future suggestions are generated using the collaborative filtering approach. This methodology guarantees that the system customizes recommendations according to the user's explicit input, even in cases where the user hasn't been officially added to the dataset. Starting with a selection of well regarded books, the system establishes a foundation for refining and customizing future recommendations for the new user.

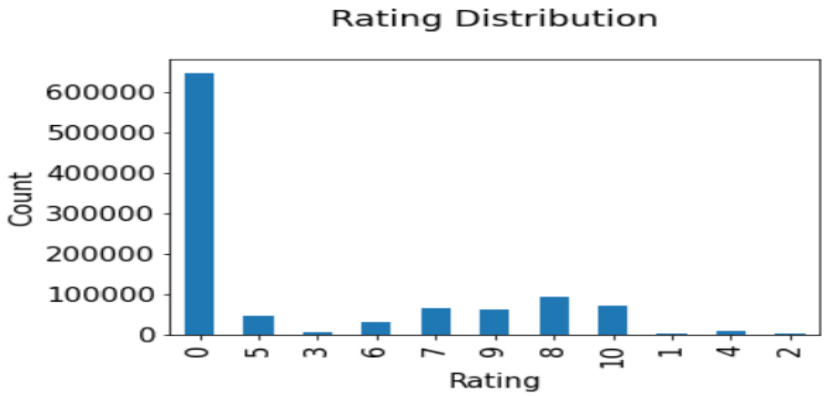

Fig. 4. Books with highest average rating

RESULTS AND ANALYSIS

In the frontend, we developed a web application using Streamlit, a Python package. This application acts as a Book Recommender system powered by machine learning. It takes the the book's title as input and provides a recommendation for a similar book as output. The recommendation is according to the user's ratings and reading patterns.

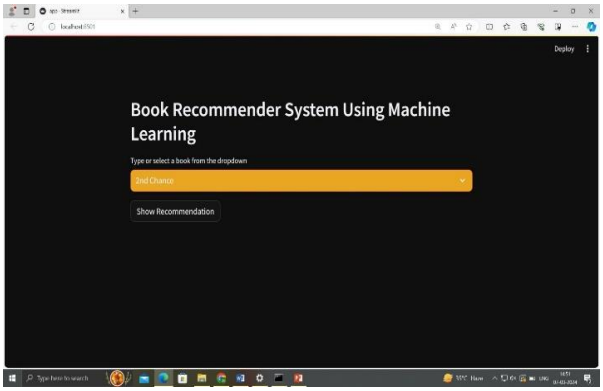

Fig. 5. Show Recommendations

Firstly, users select the title of the book they're interested in and then click on the "Show Recommendations" button. The system utilizes advanced algorithms to analyze the user's preferences and reading habits, generating personalized recommendations tailored to their tastes. By considering factors such as genre, author, and user ratings, the system suggests the top 5 books that closely match the selected title, ensuring a curated selection of relevant reading options for the user.

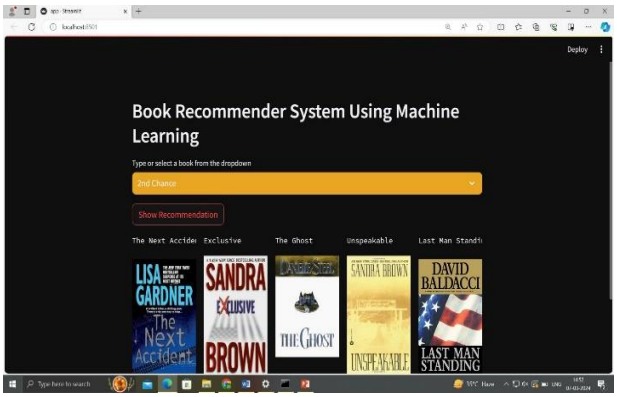

Fig. 6. Recommended Top 5 Books

CONCLUSION

This paper proposes a Collaborative Filtering Mechanism for a Book Recommendation System, aiming to enhance individuals' reading habits, vocabulary, expertise and abilities by offering quality book recommendations. The system utilizes user ratings from diverse books in a large dataset, considering a broad range of users and a vast number of books. It exploits distinctive dataset features, employing the Collaborative Filtering algorithm with cosine similarity measures to recommend the highest-rated books based on well-calculated reviews from various users. However, the system faces a primary limitation Regarding time consumption because of the
large dataset. This challenge could be addressed by implementing optimization techniques like parallel processing and caching. Moreover, a potential solution involves introducing a Hybrid Filtering approach, integrating Collaborative Filtering with other recommendation methods, potentially incorporating advanced machine learning models to optimize both efficiency and recommendation quality. Future work ought to concentrate on refining this hybrid model and leveraging innovative technologies to overcome time-related constraints, ensuring continuous improvement to obtain an additional effective and personalized user experience in the book recommendation system.

REFERENCES

[1] Book Recommendation System Devika PV , Jyothisree K , Rahul PV , S Arjun , and Jayasree Narayanan Department of Computer Science and Engineering Amrita Vishwa Vidyapeetham, Amritapuri, India 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) 2023.

[2] Kurmashov, Nursultan, Konstantin Latuta, and Abay Nussipbekov.”Online book recommendation System.” In 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO),pp.1-4. IEEE, 2015.

[3] Badriyah, Tessy, Erry Tri Wijayanto, Iwan Syarif, and Prima Kristalina.”A hybrid recommendation system for E-commerce based on product description and user profile.” In 2017 Seventh International Conference on Innovative Computing Technology (INTECH), pp. 95-100. IEEE,2017.

[4] Ahuja, Rishabh, Arun Solanki, and Anand Nayyar. ”Movie recommender system using K-Means clustering and K-Nearest Neighbor.” In 2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence), pp. 263-268. IEEE, 2019.

[5] Murali, M. Viswa, T. G. Vishnu, and Nancy Victor. ”A Collaborative Filtering based Recommender System for Suggesting New Trends in Any Domain of Research.” In 2019 5th International Conference on Advanced Computing Communication Systems (ICACCS), pp. 550553. IEEE, 2019.

[6] Mobius Data Scientist at Healthcare Melbourne, Victoria, Australia ”Book Recommendation Dataset Build state-of-the-art models for book recommendation system”, kaggle, 2016.

[7] Kurup, Ayswarya R., and G. P. Sajeev. ”Task recommendation in reward based crowdsourcing systems.” In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI),pp. 1511-1518. IEEE, 2017.

[8] Jisha, R. C., Ram Krishnan, and Varun Vikraman. ”Mobile applications recommendation based on user ratings and permissions.” In 2018 International Conference on Advances in Computing,Communications and Informatics (ICACCI), pp. 1000-1005.IEEE, 2018.