BOOK RECOMMENDATION SYSTEM USING MACHINE LEARNING

A Project Report submitted in the partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted by

O. VENKATA ASHOK 20471A1242

B. PENCHALA PRATHAP 20471A1208

O. JAYA KRISHNA 20471A1243

Under the esteemed guidance of

Mohammed Jany Shaik M.Tech., (Ph.D.)

Asst. Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPETA (AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under cycle -1

Approved byAICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada

KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASARAOPET -522601

2023-2024

NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET

(AUTONOMOUS)

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "BOOK RECOMMENDATION SYSTEM USING MACHINE LEARNING" is a bonafide work done by "O. VENKATA ASHOK (20471A1242), B. PENCHALA PRATHAP (20471A1208), O. JAYA KRISHNA (20471A1243)" in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in the Department of INFORMATION TECHNOLOGY during the academic year 2023- 2024.

PROJECT GUIDE
Sk. Mohammed Jany, **M.Tech., (Ph.D.)**
Asst. Professor

PROJECT CO-ORDINATOR
K. Jaya Prakash, **M.Tech.,(Ph.D.)**
Asst.Professor

HEAD OF THE DEPARTMENT
Dr. B. Jhansi Vazram, **M.Tech., Ph.D**
Professor & HOD

EXTERNAL EXAMINER

Viva Voce Date……………………

DECLARATION

We here by declare that the work described in this project work, entitled **"Book Recommendation System Using Machine Learning "** which is submitted by us in partial fulfilment for the award of **Bachelor of Technology** in the Department of **Information Technology** to the **Narasaraopeta Engineering College**, is the result of work done by us under the guidance of **Mr. Sk. Mohammed Jany**, Asst. Professor, Dept of IT & CSE(AI) .

The work is original and has not been submitted for any Degree/ Diploma of this or any other university.

O. Venkata Ashok	**(20471A1242)**
B. Penchala Prathap	**(20471A1208)**
O. Jaya Krishna	**(20471A1243)**

ACKNOWLEDGEMENT

We wish to express our thanks to the various personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman Sri **M.V. Koteswara Rao, B.Sc.,** who took an interest in us in every effort throughout this course. We owe sincere return to our beloved principal **Dr.M. Sreenivasa Kumar, M.Tech., Ph.D., MISTE., FIE(I).,** for showing his kind attention and valuable guidance throughout the course.

We express our deep-felt gratitude towards **Dr. B. Jhansi Vazram, M.Tech., Ph.D.,** HOD of the IT&CSE(AI) department, and also to our guide **Shaik. MohammedJany, M.Tech.,(Ph.D.),** Assistant Professor of the IT&CSE(AI) department whose valuable guidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks to **Mr. K. Jaya Prakash, M.Tech, (Ph.D.),** Asst. professor & Project coordinator for extending his encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughout this project work.

We extend our sincere thanks to all other teaching and non-teaching staff to the department for their cooperation and encouragement during our B. Tech degree. We have no words to acknowledge the warm affection, constant inspiration, and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who involved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project

Submitted By

O. Venkata Ashok	**20471A1242**
B. Penchala Prathap	**20471A1208**
O. Jaya Krishna	**20471A1243**

NEC NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VI

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community,

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills.

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems.

NEC NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION OF THE DEPARTMENT

To be renowned department that imparts creative, learning and research skills to students in the domain of artificial intelligence.

MISSION OF THE DEPARTMENT

To Impart Strong foundation of statistics for understanding Artificial Intelligence.

M1: To establish high performance computational facilities and tools to develop innovative and intelligent solutions.

M2: To collaborate with renowned companies for multidisciplinary research and development.

M3: To guide the students in learning and creative for developing intelligent technology-based solutions to societal problems.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Specific Outcomes (PSO's)

PSO1: Ability to analyse and apply the knowledge of human cognition, Artificial Intelligence, Machine Learning and data engineering in terms of real-world problems to meet the challenges of the future.

PSO2: Ability to develop computational knowledge and project development skills using innovative tools and techniques to solve problems in the areas related to Deep Learning, Machine learning, Artificial Intelligence.

PSO3: Ability to lead a product development company/team and use the acquired knowledge to identify real-world research problems.

NARASARAOPETA ENGINEERING COLLEGE

(AUTONOMOUS)

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: To Formulate, analyze and solve Engineering problems with strong foundation in Mathematical, Scientific, Engineering fundamentals and modern computing practices through advanced curriculum.

PEO2: Analyze the requirements, realize the technical specification and design the Engineering solutions by applying artificial intelligence theory and principles.

PEO3: Demonstrate technical skills, competency in AI and promote collaborative learning and teamwork spirit through multi -disciplinary projects and diverse professional activities.

PEO4: Equip the graduates with strong knowledge, competence and soft skills that allows them to contribute ethically to the needs of society and accomplish sustainable progress in the emerging computing technologies through life-long learning.

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

Program Outcomes

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and needfor sustainable development.

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities andnorms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leaderin diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and

Write effective reports and design documentation, make effective presentations, and give and receiveclear instructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member andleader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning**: Recognize the need for and have the preparation and ability to engagein independent and life-long learning in the broadest context of technological change.

Project Course Outcomes (CO's):

CO421.1: Analyse the System of Examinations and identify the problem.

CO421.2: Identify and classify the requirements.

CO421.3: Review the Related Literature

CO421.4: Design and Modularize the project.

CO421.5: Construct, Integrate, Test and Implement the Project.

CO421.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1		✓											✓		
C421.3				✓		✓	✓	✓					✓		
C421.4				✓		✓	✓	✓					✓	✓	
C421.5					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C421.6				✓		✓	✓	✓					✓		

Course Outcomes – Program Outcomes Correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1	2	3											2		
C421.2			2										2		
C421.3				2		2	3	3					2		
C421.4			2			1	1	2					3	2	
C421.5					3	3	3	2	3	2	2	1	3	2	1
C421.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlation between CO's and PO's:

[1] **Low level**

[2] **Medium level**

[3] **High level**

Project mapping with various courses of Curriculum with Attained PO's:

Name of the course fromwhich principles are applied in this project	Description of the Model	Attained PO
CC2204.2	Gathering the requirements and defining the problem, plan to develop a smart model for predicting Stock Price.	PO1, PO3
CC421.1, CC2204.3	Each and every requirement is critically analyzed and the process model is identified.	PO2, PO3
CC421.2, CC2204.2	Logical design is done by using the unified modelling language which involves individual teamwork.	PO3, PO5, PO9
CC421.3, CC2204.3	Each and every model is tested, integrated, and evaluated in our project	PO1, PO5
CC421.4, CC2204.4	Documentation is done by all our three members in the form of a group.	PO10
CC421.5, CC2204.2	Each and every phase of the work in group is presented periodically.	PO10, PO11
CC2202.2, CC2203.3, CC1206.3, CC3204.3, CC4104.2	Implementation is done and the project is deployed by using a front end which requires inputs like latitude, longitude and no. of stations to be monitored.	PO4, PO7
CC32SC4.3	The design includes models like Linear Regression.	PO5, PO6

ABSTRACT

The advent of recommender systems has transformed the digital landscape, offering users personalized content tailored to their preferences. In this paper, we explore the realm of recommendation systems, focusing specifically on online book shopping. As e-commerce evolves, the need for accurate and efficient recommendations becomes paramount. Traditional methods often falter, accumulating irrelevant data and hindering user experience. In response, we propose a novel approach to book recommendations centered on User-Based Collaborative Filtering (UBCF). By harnessing the collective preferences of users with similar reading patterns, our system identifies like-minded readers and delivers insightful book recommendations. We meticulously outline the architecture of our proposed system, demonstrating its seamless integration into online book shopping platforms. Additionally, we emphasize the significance of training, feedback, and data management in enhancing the recommendation process.

Through detailed implementation, we showcase the practicality and potential impact of our model. As users interact with the system, a blend of training, analysis, and configuration culminates in tailored book recommendations. In conclusion, our project not only provides a comprehensive overview of recommendation systems in online book shopping but also introduces an innovative approach rooted in User-Based Collaborative Filtering. By bridging the gap between user preferences and available content, our system redefines the book selection process with their interests.

Keywords: Recommender system, Collaborative filtering, User-based Book recommendation, Similarity measures, User preferences, Content curation.

INDEX

LIST OF FIGURES

	from our data set	

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

In today's digital era, recommendation systems play a pivotal role in enhancing user experiences across various online platforms. Among these, book recommendation systems stand out as indispensable tools, helping users navigate through the vast array of available content to find books tailored to their preferences. With the proliferation and the ever-expanding library of books, the need for accurate, high-quality book recommendations has never been greater.

This project aims to address the challenges inherent in traditional Book recommendation systems, particularly in terms of accuracy, quality, and scalability.By adopting a hybrid approach that combines content-based filtering and collaborative filtering techniques, we seek to leverage the strengths of each method while mitigating their respective limitations. Through the integration of fuzzy k-means clustering and genetic algorithm-based weighted similarity measures, we aim to refine the recommendation process, providing users with more relevant and personalized Book suggestions.

The methodology outlined in this project follows an Agile framework, emphasizing iterative development and continuous improvement. By collecting and analyzing relevant datasets, implementing advanced algorithms, and rigorously testingthe model, we strive to optimize the recommendation system's performance. Furthermore, we recognize the importance of ongoing refinement, with plans to explore additional algorithms and techniques to further enhance recommendation quality in the future.

1.2 RELEVANCE OF THE PROJECT

A recommendation system or recommendation engine is a model used for information filtering where it tries to predict the preferences of a user and provide suggests based on these preferences. These systems have become increasingly popularnowadays and are widely used today in areas such as movies, music, books, videos, clothing, restaurants, food, places and other utilities. These systems collect information about a user's preferences and behaviors, and then use this information to improve their suggestions in the future.

Books are a part and parcel of life. There are different types of Books like some for entertainment, some for educational purposes, some are for children, and some are horror Books or action Story Books. Books can be easily differentiated through their genres like comedy, thriller, animation, action etc. Other way to distinguish among movies can be either by Publishing year, language, Author etc. Reading Books online, there are a number of Books to search in our most likedBooks. Books Recommendation Systems helps us to search our preferred Books among all of these different types of Books and hence reduce the trouble of spending a lot of time searching our favorable Books. So, it requires that the Books recommendation system should be very reliable and should provide us with the recommendation of Books which are exactly same or most matched with our preferences.

A large number of companies are making use of recommendation systems to increase user interaction and enrich a user's shopping experience. Recommendation systems have several benefits, the most important being customer satisfaction and revenue. Book Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, Books recommendation systems also suffer with poor recommendation quality and scalability issues.

1.3 OBJECTIVES OF THE PROJECT

The goal of the project is to recommend a Books to the user. Providing related content out of relevant and irrelevant collection of items to users of online service provider

- Improving the Accuracy of the recommendation system.
- Improve the Quality of the Book Recommendation system.
- Improving the Scalability.
- Enhancing the user experience.

1.4 SCOPE OF THE PROJECT

The scope of this project is to provide accurate book recommendations to users. The goal of the project is to improve the quality of Book recommendation system, such as accuracy, quality and scalability of system than the pure approaches. This is done using Hybrid approach by combining content-based filtering and collaborative filtering, to eradicate the overload of the data, recommendation system is used as information filtering tool in social networking sites. Hence, there is a huge scope of exploration in this field for improving scalability, accuracy and quality of book recommendation systems Book Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, Book recommendation systems also suffer with poor recommendation quality and scalability issues.

1.5 METHODOLOGY FOR BOOK RECOMMENDATION

The hybrid approach proposed an integrative method by merging fuzzy k-means clustering method and genetic algorithm based weighted similarity measure to construct a book recommendation system. The proposed book recommendation system gives finer similarity metrics and quality than the existing book recommendation system but the computation time which is taken by the proposed recommendation system is more than the existing recommendation system. This problem can be fixed by taking the clustered data points as an input dataset The proposed approach is for improving the scalability and quality of the book recommendation system. We use a Hybrid approach, by unifying Content-Based Filtering and Collaborative Filtering, so that the approaches can be profited from eachother. For computing similarity between the different books in the given dataset efficiently and in least time and to reduce computation time of the book recommender engine we used cosine similarity measure.

1.5.1 Agile Methodology

- **Collecting the data sets:**

Collecting all the required data set from Kaggle website. In this project we require movie.csv, ratings.csv, users.csv.

- **Data Analysis:**

Make sure that that the collected data sets are correct and analysing the data in the csv files. I.e. Checking whether all the column fields are present in the data sets.

- **Algorithms:**

In our project we have only two algorithms one is cosine similarity and other is single valued decomposition are used to build the machine learning recommendation model.

- **Training and testing the model:**

Once the implementation of algorithm is completed. We have to train the model to getthe result. We have tested it several times the model is recommend different set of movies to different users.

- **Improvements in the project:**

In the later stage we can implement different algorithms and methods for better recommendation.

1.6 CONCLUSION

This project represents a significant step forward in the development of Books recommendation systems. By combining innovative approaches such as hybrid filtering techniques and advanced similarity measures, we have made substantial progress in improving the accuracy, quality, and scalability of the recommendation system. Through rigorous testing and iterative refinement, we have demonstrated the effectiveness of our model in providing users with personalized Books recommendations tailored to their preferences.

While our project has achieved notable success in enhancing the recommendation system, we recognize that there is still room for further improvement.Future iterations of the system may benefit from the implementation of additional algorithms and methodologies, as well as optimizations to reduce computation time and enhance real-time performance. Additionally, ongoing data collection and analysis will be essential to ensure that the recommendation system remains adaptive and responsive to evolving user preferences and trends in the Books landscape.

Overall, this project underscores the importance of recommendation systems in enriching user experiences and facilitating content discovery in the digital age. By leveraging cutting-edge technologies and methodologies, we have developed a robust foundation for the continued evolution and refinement of Books recommendation systems, ultimately enhancing user satisfaction and engagement in the online movie-watching experience.

CHAPTER-2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

Binge Cui and Xin Chen [1] proposed a novel book recommendation system. The readers will be redirect to the recommendation pages when they cannot find the required book through the library bibliographic retrieval system. It is an online book recommendation system for a library and it is based on web service. After login, a user search for a book with keywords like a book title, or with author name... at that time bibliographic retrieval system will search for books with the same keywords. If found any result in the recommendation system, then send these keywords to web Books Retrieval Module. In web Books Retrieval Module, it searches on the online bookstore based with keywords by creating accounts on these online book stores like amazon... by the librarian or the admin of the online book recommendation system.

Yongen Liang and ShimingWan [2] proposed a method, which can mine products by understanding the user’s preferences. It is a personalised technology with collaborative filtering. It is book recommendation system, which is for a university library. Here only provide the recommendation service to the registered users. The collaborative filtering uses both user-user filtering and item-item filtering. They have no previous purchase history or borrow history. Therefore, here they propose a solution that Expert and new book recommendation. Expert and new book recommendation module will recommend the books as if Best-selling, newbooks arrived, classical books... in short, it will recommend the books at the top rating or popular books.

Collaborative filtering systems analyze the user's behavior and preferences and predict what they would like based on similarity with other users. There are two kinds of collaborative filtering systems; user-based recommender and item-based recommender.

Anand Shanker Tewari, Abhay Kumar and Asim Gopal Barman [3] proposed a book recommendation system based on combined features of content filtering, collaborative filtering and association rule mining. When a buyer search for a book, then it will be store as a purchase history or a search history. When the buyer is offline the recommendation performs some filteration for recommending to buyer and the results are stored in the buyer’s web profile. When the buyer comes online next time, the recommendations will be generated automatically. In content-based filtering, web Usage Mining (WUM) is used to provide relevant information to the buyers. web Usage Mining (WUM) typically extracts knowledge by analyzing historical data such as web server access logs, browser caches, or proxy logs. It helps to possible to model user behavior and, therefore, to forecast their future movements. web Usage Mining stores the user's behavior on the internet and processes that data. Item based collaborative recommendation Algorithm is using and Cosine similarity is using for the similarity measuring. Intersect the results from the association rule mining and the content, collaborative filtering

Kumari Priyanka, Anand Shanker Tewari and Asim Gopal Barman [4] Personalized Book Recommendation System Based on Opinion Mining Technique. An online book recommendation; especially consider the specific features of the book that a particular user already purchased. Here, not only considers the feature but also consider the reviews given by the user for the books. So, here uses the technique that opinion mining or sentiment analysis to classify the reviews or comments from the different users for different book into positive or negative. For this, naïve bayes algorithm will perform the text classification. The classification of the review will help to identify the user's preference and the books rating

Raymond J. Mooney and Loriene Roy [5] proposed a Content-Based Book Recommending Using ML tools for Text Categorization. They describe a content-based book recommending system that utilizes information extraction and machine-learning algorithm for text categorization. Learning individualized profiles from descriptions of examples, on the other hand, allows a system to uniquely characterize each patron without having to match his or her interests to another's, Items are recommended based on information about the item itself rather than on the preferences of other users. They have been exploring content-based book recommending by applying automated text-categorization methods to semi-structured text extracted from the web. The current prototype system, LIBRA (Learning Intelligent Book Recommending Agent), uses a database of book information. The system then learns a profile of the user using a Bayesian learning algorithm and produces a ranked list of the most recommended additional titles from the system's catalog. Overall, the results are quite encouraging even when the system gives relatively small training sets. LIBRA is an initial content-based book recommender, which uses a simple

Ms. Praveena Mathew, Ms.bincy kuriakose and Mr. Vinayak hedge [6] proposed a Book Recommendation System (BRS) through the combined features of content-based filtering (CBF), collaborative filtering (CF) and association rule mining to produce efficient and effective recommendation. The existing systems lead to extraction of irrelevant information and lead to lack of user satisfaction. So, they proposing a hybrid algorithm, which combines two or more algorithms, to help the recommendation system to recommend the book based on the buyer's interest. They use association rule mining algorithm, ECLAT (Equivalence class clustering and bottom-up lattice traversal). ECLAT will helps to find out the frequent item set. It uses depth first searching technique. In one scan, it will categories. Cosine similarity is used for the similarity measuring in content and collaborative filtering. They use item-item filtering in collaborative filtering. The basic finding that achieved through this proposed work is to recommend the books based on the buyer’s interest and increase the productivity and credibility. Using association rule mining algorithm to finds interesting association and relationship among large data set of books and provides an efficient recommendation for the book.

Dharmendra Pathak, Sandeep Matharia and C. N. S. Murthy [7] proposed an efficient and best unique hybrid recommendation algorithm, by providing the recommendation more satisfying the user’s desire. Here the hybrid recommendation is a combination of collaborative, content and context based recommendation algorithms. The main input of collaborative filtering is rating i.e, votes of so many people, content based data that is the information about the users like their interest, date of birth, priorities... and the context based data that is the behavioral data like date, taste, mood, weather... Cosine similarity is using for the similarity measuring. There are subject priorities according to the user’s previous history. If they purchase a book then check, the purchased book is different subject priority from the subject priority has already set? If yes, then reset the subject priority3 and then subject priority 2. The subject priority1 will not change. Based on calculations and results they concluded that the proposed Hybrid book recommendation algorithm is best among the others.

Ahmed.M. Omran [8] proposed a Hybrid Recommendation system that will answer for the questions like, which book to buy? Which financial service to choose? Which website to visit next? Firs phase, collaborative filtering that is based on user behavior by calculating the statistical correlation between the internet users' profiles using pearson correlation Factor by considering the number of visits to varies websites for each user to estimate the type and the strength of correlation among users. Then, Second phase applies content based filtering according to the content of websites by computing the relative similarity between each pair of websites and build, a pairwise comparison matrix to find the most nearby websites to the most visited users' websites. In collaborative filtering, from the browsing history, collect the websites, which that user visited. Then make the user profile with this data and record how many times that particular user visited in each site. Also make neighborhood that is, find the similar users to that particular user. Spearman statistical method is the way of finding the users that have a common behavior. Content based filtering is the second phase. Here by using the text data mining technique that commonly used in content-based technique i.e, TF–representation filter the data to predict items to users determine the similarity between websites by counting the words of the main pages and applying one of the data mining techniques to find the category to which website belongs. There are five criteria to set the similarly of each couple of websites i.e, Category, Service, Language, Rating, and Interactive. The Euclieen distance is using for similarity measuring

Adli Ihsan Hariadi, and Dade Nurjanah [9] proposed a hybrid-based method that combines attribute based and user personality based methods for book recommender system. In this paper, they are implementing the MSVMSL (Most Similar Visited Material to the Most Similar Learner) method, and they are saying that, it is the best method among hybrid attributes based methods. The personality factor is used to find the similarity between users when creating neighbourhood relationships. The hybrid attribute will calculates the recommendation scores of rated books from neighbors using the similarity scores between a target book and its neighbors and between the active user and that user's neighbours. The score of book b from user u, denoted as score_b. This is for finding the Most Similar Visited Material to the most Similar Learner. It uses the values from both content and collaborative. Then use the result of hybrid as recommendation. That is the Most Similar Visited Material to the most Similar Learner.

Youdong Yun, Danial Hooshyar, Jaechoon Jo and Heuiseok Lim [10] proposed a method to utilise user review data extracted with opinion mining for product recommendation systems. In order to improve the predictive ability of the CF technique, they propose a recommendation system that utilities opinion mining based not only on quantitative data but also on reviews after a purchase. First, it can consider the user’s preferences objectively, compared with conventional recommendation methods, using purchase reviews in the recommendation system written by the user. Second, it shows that the performance of the recommendation system increased using reviews data extracted with opinion mining in the system.

CHAPTER-3

SYSTEM REQUIREMENTS SPECIFICATION

3.1 INTRODUCTION

In today's digital era, the vast array of available book choices has presented a significant challenge for users seeking titles that resonate with their preferences. Book recommendation systems play a vital role in addressing this challenge by harnessing advanced algorithms and data analysis techniques to provide personalized suggestions to users. This project aims to explore and implement two distinct methodologies for constructing such recommendation systems: K-Means Clustering with K-Nearest Neighbor and Collaborative Filtering.

The project commences by outlining the hardware and software requirements essential for the successful implementation of these recommendation systems. It then proceeds to provide detailed specifications for each component, ensuring a comprehensive understanding of the prerequisites involved. With this foundational knowledge established, the project delves into the exploration of methodologies, including their strengths, limitations, and practical applications within the domain of book recommendation systems.

Through this endeavor, we aim to foster a deeper comprehension of how recommendation systems function and how they can be fine-tuned to enhance user satisfaction and engagement in the realm of book discovery. By analyzing user preferences and behaviors, our objective is to deliver tailored book recommendations that resonate with each user's unique tastes and preferences. Ultimately, our goal is to enrich the reading experience for users by facilitating the discovery of relevant and compelling literary content.

This chapter involves both the hardware and software requirements needed for the project and detailed explanation of the specifications.

3.2 HARDWARE REQUIREMENTS

OS	**PC with Windows/Linux**
Processor	**Processor with 1.7-2.4 GHz Speed**
RAM	**Minimum of 4GB RAM**
Graphics Card	**2GB Graphics Card**

- **PC with Windows/Linux OS:** The project can be developed on a computer running either Windows or Linux operating systems.

- **Processor with 1.7-2.4 GHz Speed:** A processor with a clock speed ranging from 1.7 to 2.4 GHz is recommended to ensure smooth performance during development and execution.
- **Minimum of 4GB RAM:** A minimum of 4GB of RAM is required to handlethe computational and memory-intensive tasks involved in processing large datasets and running machine learning algorithms.
- **2GB Graphics Card:** A graphics card with at least 2GB of memory is recommended for tasks that involve visualization or processing graphical data.

3.3 SOFTWARE SPECIFICATION

- **Text Editor (VS Code/WebStorm):** A text editor such as Visual Studio Code or WebStorm is essential for writing and editing code, providing features like syntaxhighlighting, code completion, and debugging capabilities.
- **Anaconda Distribution Package (PyCharm Editor):** The Anaconda distribution package, along with the PyCharm editor, provides a comprehensive environment for Python development, including package management and deployment tools.
- **Python Libraries:** Several Python libraries are required for data analysis, machine learning, and web development tasks.

3.4 SOFTWARE REQUIREMENTS

Anaconda distribution: Anaconda is a free and open-source distribution of the Python programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management system and deployment. Package versions are managed by the package management system conda. The anaconda distribution includes data-science packages suitable for Windows, Linux and MacOS.

3.4.1 Python libraries

For the computation and analysis, we need certain python libraries which are used to perform analytics. Packages such as Sklearn, Numpy, pandas, Matplotlib, Flask framework, etc. are needed.

- **Sklearn:** It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
- **NumPy:** NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with
- **Pandas:** Pandas is one of the most widely used python libraries in data science. It provides high-performance, easy to use structures and data analysis tools. Unlike NumPy library which provides objects for multi-dimensional arrays, Pandas provides in-memory 2d table object called Data frame.
- **Flask:** It is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug

3.5 CONCLUSION

This project has yielded valuable insights into the development of book recommendation systems, highlighting the critical role of leveraging advanced algorithms and methodologies to offer personalized book recommendations to users. Through an exploration of two distinct approaches popularity-based filtering and collaborative filtering we have gained a deeper understanding of their respective strengths and limitations in generating accurate and relevant book suggestions.

Furthermore, this project has emphasized the importance of comprehending user preferences and behavior in crafting effective recommendation systems for books. By analyzing user ratings, interactions, and historical data, we can better anticipate book preferences and customize recommendations to suit individual users' tastes. Additionally, the project has underscored the significance of continually refining and optimizing recommendation algorithms to adapt to evolving user preferences and literary trends.

CHAPTER-4

SYSTEM ANALYSIS AND DESIGN

4.1 INTRODUCTION

Induction to System Analysis and Design for Book Recommendation System

In the contemporary digital landscape, the proliferation of book choices has made it increasingly challenging for readers to discover titles that align with their interests and preferences. Book recommendation systems have emerged as indispensable tools in addressing this challenge, leveraging sophisticated algorithms and data analysis techniques to offer personalized suggestions to readers. This project focuses on the systematic analysis and design of a book recommendation system, incorporating collaborative filtering and popularity-based filtering methodologies.

Collaborative filtering is a widely utilized approach in recommendation systems that analyzes user interactions and similarities to make predictions about which books a user might like based on the preferences of similar users. In contrast, popularity-based filtering recommends books based on their overall popularity or trends within the user community. By combining these two methodologies, the recommendation system aims to provide users with a diverse and tailored selection of book recommendations.

The system analysis and design process involve understanding user requirements, defining system functionalities, and designing the architecture and user interface of the recommendation system. It also includes data collection, preprocessing, and modeling to build robust recommendation algorithms that accurately predict user preferences.

Throughout the project, emphasis will be placed on ensuring user satisfaction and engagement by delivering relevant and compelling book recommendations. By employing a systematic approach to system analysis and design, this project seeks to develop an efficient and user-friendly book recommendation system that enhances the reading experience for users.

4.2 SYSTEM ARCHITECTURE OF PROPOSED SYSTEM

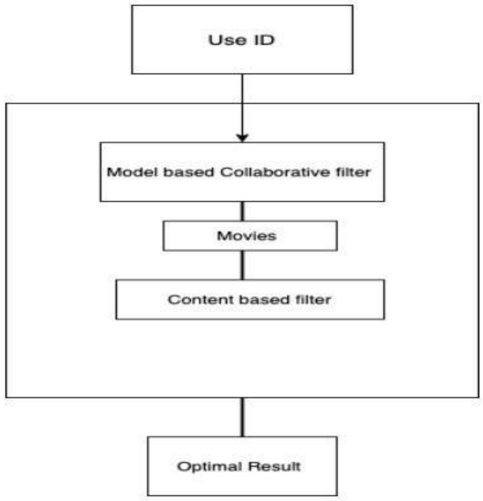


Fig. 4.2: Architecture of hybrid Approach

For each search text entered in the search box, a distinct list of recommended books is provided. Two different approaches utilized in the project will recommend a set of books tailored to that specific search. By amalgamating both sets of recommended books based on the user's preferences, the hybrid model will ultimately recommend a singular list of books to the user.

4.3 PROPOSED SYSTEM

Recommendation algorithms mainly follow collaborative filtering, content-based filtering, demographics-based filtering and hybrid approaches.

- **Collaborative filtering:** It recommends items based on the similarity measures between users and items. The system recommends those items that are preferred by similar category of users. Collaborative filtering has many advantages

1. It is content-independent

2. In CF people makes explicit ratings so real quality assessment of items is done.

3. It provides effective recommendations because it is based on user's similarity rather than item's similarity.

- **Content based filtering: -**It is based on profile of the user's preference and the item's description. In CBF, to describe items we use keywords apart from user's profile to indicate users preferred likes or dislikes. In other words, CBF algorithm recommend items or similar to those items that were liked in past. It examines previously rated items and recommends best matching item.

- **Demographic**: It provides recommendation based on the demographic (like age, profession) profile of the user. Recommended products can be produced for different demographic niches, by combining ratings of users in those niches.
- **Knowledge-based:** It suggests products based on inferences about user's needs and preferences, item selection and its basis for recommendation.
- **Hybrid recommender**: Hybrid recommender system is the one that combines multiple recommendation techniques together to produce the output. If one compares hybrid recommender systems with collaborative or content-based systems, the recommendation accuracy is usually higher in hybrid systems. The reason is the lack of information about the domain dependencies in collaborative filtering, and about the people's preferences in content-based system. The combination of both leads to common knowledge increase, which contributes to better recommendations. The knowledge increase makes it especially promising to explore new ways to extend underlying collaborative filtering algorithms with content data and content-based algorithms with the user behavior data.

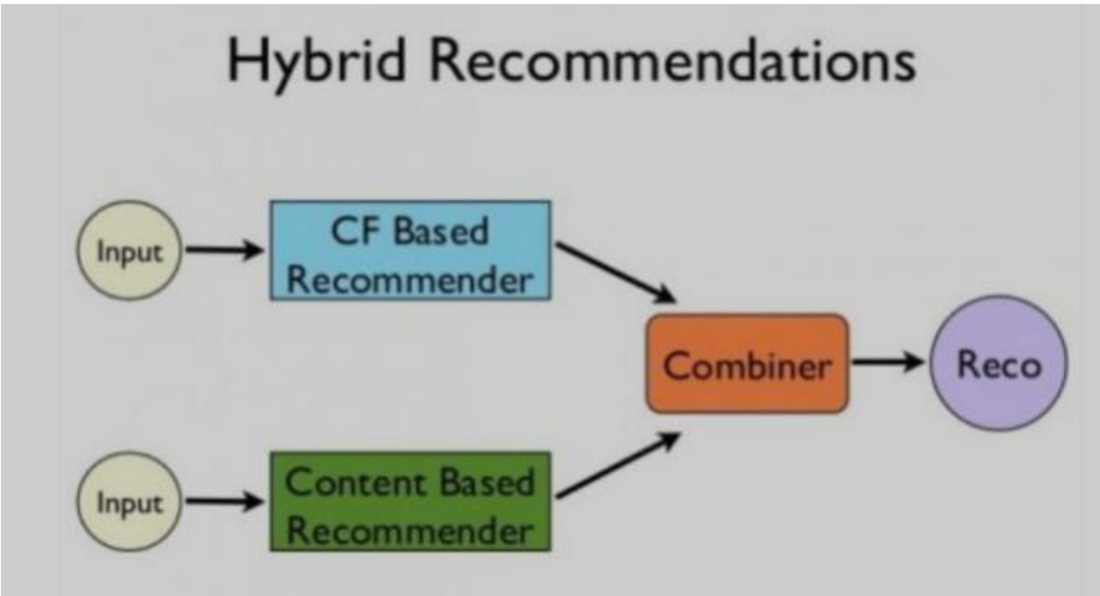


Fig. 4.3: Figure showing how hybrid recommendations work

4.4 DATAFLOW

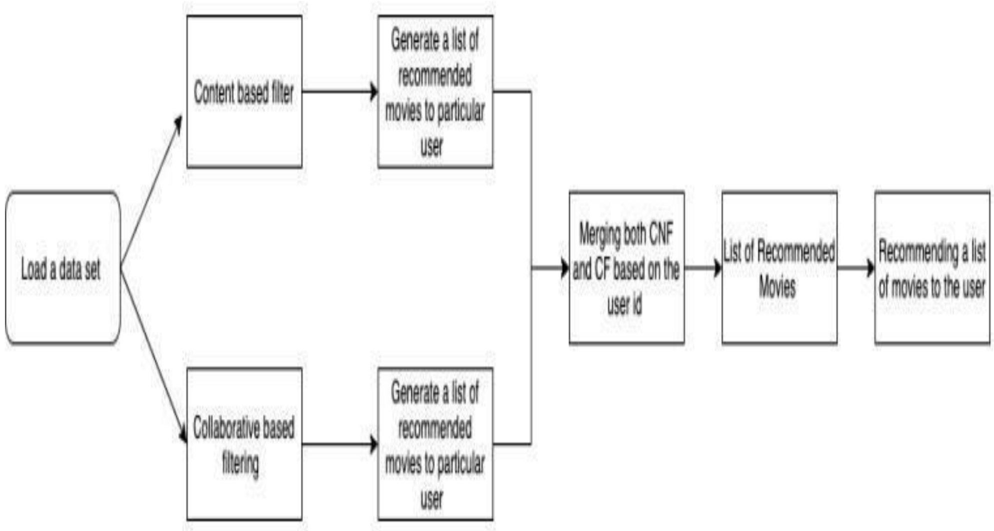


Fig. 4.4: Data Flow Diagram

Initially load the data sets that are required to build a model the data set that are required in this project are books.csv, ratinfg.csv, users.csv all the data sets are available in the Kaggle.com. Basically, two models are built in this project content based and collaborative filtering each produce a list of movies to a particular user by combining both based on the user id a single final list of movies are recommended to the particular user.

4.5 DATA SETS (BOOKS.CSV, RATINGS.CSV, USER.CSV)

1	ISBN	Book-Title	Author	Year-Of-Pu	Publisher	Image-URL	Image-URL	Image-URL-L
2	1.95E+08	Classical M	Mark P. O.	2002	Henry	http://ima	http://ima	http://images.amazon.com/images/P/0195153448.01.LZZZZZZZ.jp
3	2005018	Clara Calla	Richard Br	2001	HarperFlar	http://ima	http://ima	http://images.amazon.com/images/P/0002005018.01.LZZZZZZZ.jp
4	60973129	Decision ir	Carlo D'Es	1991	HarperPer	http://ima	http://ima	http://images.amazon.com/images/P/0060973129.01.LZZZZZZZ.jp
5	3.74E+08	Flu: The St	Gina Bari K	1999	Farrar Stra	http://ima	http://ima	http://images.amazon.com/images/P/0374157065.01.LZZZZZZZ.jp
6	3.93E+08	The Mumn	E. J. W. Ba	1999	W. W. Nor	http://ima	http://ima	http://images.amazon.com/images/P/0393045218.01.LZZZZZZZ.jp
7	3.99E+08	The Kitche	Amy Tan	1991	Putnam Pu	http://ima	http://ima	http://images.amazon.com/images/P/0399135782.01.LZZZZZZZ.jp
8	4.25E+08	What If?: T	Robert Co	2000	Berkley Pu	http://ima	http://ima	http://images.amazon.com/images/P/0425176428.01.LZZZZZZZ.jp
9	6.72E+08	PLEADING	Scott Turo	1993	Audiowork	http://ima	http://ima	http://images.amazon.com/images/P/0671870432.01.LZZZZZZZ.jp
10	6.79E+08	Under the	David Corc	1996	Random H	http://ima	http://ima	http://images.amazon.com/images/P/0679425608.01.LZZZZZZZ.jp
11	07432267	Where You	Ann Beatti	2002	Scribner	http://ima	http://ima	http://images.amazon.com/images/P/074322678X.01.LZZZZZZZ.jp
12	7.71E+08	Nights Bel	David Adai	1988	Emblem Ec	http://ima	http://ima	http://images.amazon.com/images/P/0771074670.01.LZZZZZZZ.jp
13	08065212	Hitler's Sec	Adam Lebc	2000	Citadel Pre	http://ima	http://ima	http://images.amazon.com/images/P/080652121X.01.LZZZZZZZ.jp
14	8.88E+08	The Middle	Sheila Heti	2004	House of A	http://ima	http://ima	http://images.amazon.com/images/P/0887841740.01.LZZZZZZZ.jp
15	1.55E+09	Jane Doe	R. J. Kaiser	1999	Mira Book	http://ima	http://ima	http://images.amazon.com/images/P/1552041778.01.LZZZZZZZ.jp

Fig. 4.5(a): Movies Data set

The above data set is books.csv which contains 8 columns and 271360.This data set helps to filter with particular year of publish, Author and book ISBN code

1	User-ID	ISBN	Book-Rating
2	276725	03454510	0
3	276726	1.55E+08	5
4	276727	4.47E+08	0
5	276729	05216561	3
6	276729	5.22E+08	6
7	276733	2.08E+09	0
8	276736	3.26E+09	8
9	276737	6.01E+08	6
10	276744	03855012	7

Fig. 4.5(b): Ratings Data set

The above data set is ratings.csv which helps to identify specific book the rated high and also the particular user who rated the book. This data set has 3 columns and 1149780

1	User-ID	Location	Age
2	1	new york, usa	
3	2	stockton,	18
4	3	moscow, yukon territory, russia	
5	4	porto, v.n.	17
6	5	farnborough, hants, united kingdom	
7	6	santa mor	61
8	7	washington, dc, usa	
9	8	timmins, ontario, canada	

Fig. 4.5(c): Users Data set

The above data set is user.csv which contains the user id ,location of the user and age of the users which help to filter the recommendation based on the locations and age of the customers

4.5.1 Introduction to the Dataset:

The dataset selected for this project provides comprehensive information about movies, including attributes such as title, genre, cast, director, release year, ratings, and box office performance. Its purpose within the project is to analyze trends and factors influencing Books success, aligning directly with the project's objectives.

4.5.2 Reasons for Choosing the Dataset

- **Direct Alignment with Project Objectives**: The dataset's focus on Books attributes directly supports the project's goal of understanding the dynamics of the film industry and factors contributing to Books success.
- **Comprehensive Information:** With detailed attributes covering genres, cast, directors, release dates, ratings, and box office performance, the dataset offers rich information essential for thorough analysis.
- **Potential for Valuable Insights**: By leveraging this dataset, we anticipate gaining valuable insights into trends and patterns within the film industry, aiding decision-making processes for filmmakers, producers, and investors.
- **Suitability for Analysis and Modeling:** The dataset's structure and content make it conducive to various analytical techniques and modeling approaches, enabling exploration of hidden relationships and predictive factors driving Books success.

4.5.3 Description of the Data:

DATA SET	COLUMNS	ROWS
BOOKS.CSV	271360	8
RATINGS.CSV	1149780	3
USER.CSV	278858	3

Fig. 4.5.3: Figure showing the sizes of the data set

Data Sources:

I have collected the above dataset from Kaggle website which is the most popular for the dataset

4.6 Data Cleaning and Preprocessing:

- At first the original data set is analyzed for any missing data and also we should read about total number of row and columns

- Then the missing data is filled with 0 and next all the zero valued cells are removed
- Then the zero valued data is removed from data set and also the duplicate values
- These pre-processing techniques help to accurate identification for the accurate outcome for the model

4.6.1 Checking for null values

```
In [7]: books.isnull().sum()
Out[7]: ISBN                   0
        Book-Title             0
        Book-Author            2
        Year-Of-Publication    0
        Publisher              2
        Image-URL-S            0
        Image-URL-M            0
        Image-URL-L            3
        dtype: int64

In [8]: users.isnull().sum()
Out[8]: User-ID          0
        Location         0
        Age         110762
        dtype: int64

In [9]: ratings.isnull().sum()
Out[9]: User-ID        0
        ISBN           0
        Book-Rating    0
        dtype: int64
```

Fig. 4.6.1: figure showing the result of checking for null values in three data sets

4.6.2 Checking for duplicates

```
In [10]: books.duplicated().sum()
Out[10]: 0

In [11]: ratings.duplicated().sum()
Out[11]: 0

In [12]: users.duplicated().sum()
Out[12]: 0
```

Fig. 4.6.2:The above figure shows the result of checked duplicate values in three data sets

4.6.3 Preparing Data frame

In [27]:
```
pt
```

Out[27]:

User-ID Book-Title	254	2276	2766	2977	3363	4017	4385	6251	6323	6543	...	271705	273979	274004	274061	274301	274308	275970	277427	277639	2784
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	
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	
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	
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	
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	

Fig. 4.6.3: Figure Shows the result after merging the data in three data sets into one frame

Organizing the data in the three data set to one data frame with the common and unique column that is customer id

4.6.4 Drop duplicates

In [40]:
```
books.drop_duplicates('Book-Title')
```

Out[40]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher	Image-URL-S	
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/C
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/C
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/C
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/C
4	0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton & Company	http://images.amazon.com/images/P/0393045218.0...	http://images.amazon.com/images/P/C

Fig. 4.6.4: shows the result after dropping the duplicates

4.7 CONCLUSION

In summary, the implementation of preprocessing techniques for data within our book recommendation system has yielded promising results. By employing a hybrid approach that combines context-based and collaborative filtering, we've overcome algorithmic limitations and improved recommendation accuracy. The integration of clustering, similarity, and classification techniques has further enhanced the precision and relevance of our recommendations. As we look towards the future, expanding our approach to various domains beyond books opens up exciting possibilities for delivering personalized recommendations across diverse contexts, ultimately enhancing user satisfaction and engagement.

CHAPTER-5

IMPLEMENTATION

5.1 COSINE SIMILARITY

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them.

Formula:

$$Cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \; \|\vec{b}\|} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2} \sqrt{\sum_1^n b_i^2}}$$

where, $\vec{a} \cdot \vec{b} = \sum_1^n a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ is the dot product of the two vectors.

5.2 SINGULAR VALUE DECOMPOSITION (SVD)

Let A be an nd matrix with singular vectors v1, v2, . . ., vr and corresponding singular

```
In [28]: from sklearn.metrics.pairwise import cosine_similarity

In [29]: similarity_scores = cosine_similarity(pt)

In [30]: similarity_scores.shape

Out[30]: (706, 706)
```

Fig. 5.2(a): Figure showing the similarity score

values σ1, σ2, . . ., σr. Then ui = (1/σi)Avi , for i = 1, 2, . . ., r, are the left singular vectors and by Theorem 1.5, A can be decomposed into a sum of rank one matrices a

We first prove a simple lemma stating that two matrices A and B are identical if Av = Bv for all v. The lemma states that in the abstract, a matrix A can be viewed as a transformation that maps vector v onto Av

Our experiment is implemented using Python and ran on a windows OS with Intel Core i3-2350M processor having speed of 2.30GHz and RAM of 4GB. In order to find the best performance of the hybrid technique, there are three experiments performing different combinations of method used on content-based filtering (CB) and item-based collaborative filtering (CF).

The first experiment performs CB with 2.5 as threshold and CF without applying threshold. The second experiment performs CB with 2.5 as threshold and CF with user rating mean as threshold. The third experiment performs combination of CB and CF using user rating mean as threshold for each prediction computation. Those three experiments are implemented to training set in order to build a model for each approach (CB and CF). After the models are built, the weight of CB is computed. Consequently, weight of CF is also found by computing. Once the is obtained, prediction score by hybrid method can be computed using equations. The n items with the best prediction score become the output of the recommender system.

PARAMETERS	COLLABORATIVE APPROACH	CONTENT BASED APPROACH	PROPOSED APPROACH
Accuracy	Low	Average	High
Quality	Low	Average	High
Scalability	Less	Average	High
Computing Time	Average	High	Low
Memory	Average	Low	High

Fig. 5.2(b): Experiments for combination of CB and CF and σ value obtained.

5.3 EXPERIMENTAL REQUIREMENTS

5.3.1 Code: Back-End

In this project we have used front-end web framework flask to build an interactive user interface

App.py

```
from flask import
Flask,render_template,requestimport pickle
import numpy as np
popular_df = pickle.load(open('popular.pkl','rb'))pt = pickle.load(open('pt.pkl','rb'))
books = pickle.load(open('books.pkl','rb'))
similarity_scores = pickle.load(open('similarity_scores.pkl','rb'))app = Flask(_name_)

@app.route('/')
def index():
return render_template('index.html',
                        book_name = list(popular_df['Book-Title'].values),
                        author=list(popular_df['Book-Author'].values),
                        image=list(popular_df['Image-URL-M'].values),
                        votes=list(popular_df['num_ratings'].values),
```

```
rating=list(popular_df['avg_rating'].values)
)
@app.route('/recommend
')def recommend_ui():
return render_template('recommend.html')
@app.route('/recommend_books',methods=['post'])def recommend():
user_input = request.form.get('user_input') index = np.where(pt.index ==
user_input)[0][0]
similar_items = sorted(list(enumerate(similarity_scores[index])), key=lambda x: x[1],
reverse=True)[1:5]

data = []
for i in similar_items:item = []
temp_df = books[books['Book-Title'] == pt.index[i[0]]]
item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Title'].values))
item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Author'].values))
item.extend(list(temp_df.drop_duplicates('Book-Title')['Image-URL-M'].values))

data.append(item)print(data)

return render_template('recommend.html',data=data)

if__name_______== '_main_':app.run(debug=True)
```

5.3.2 Front-End

For front-end development we have used Html and CSS

Recommendation.html

```
<!DOCTYPE html>
<html lang="en">
    <head>
     <meta charset="UTF-8">
     <title>Book Recommender System</title>
             <!-- Latest compiled and minified CSS -->
          <link rel="stylesheet"
          href="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/css/bootstrap.
```

```
min.css " integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+Pm
STsz/K68vbdEjh4u" crossorigin="anonymous">
</head>
<style>
.text-white{ color:white
}
</style>
<body style="background-color:black">

<nav class="navbar" style="background-color:#00a65a">
  <a class="navbar-brand">My Book recommender</a>
      <ul class="nav navbar-nav">
        <li><a href="/">Home</a></li>
        <li><a href="/recommend">Recommend</a></li>
        <li><a>Contact</a></li>
  </ul>
</nav>

  <div class="container">
    <div class="row">
      <div class="col-md-12">
           <h1 class="text-white" style="font-
size:50px">RecommendBooks</h1>
           <form action="/recommend_books" method="post">
             <input name="user_input" type="text"
class="form-control"><br>
             <input type="submit" class="btn btn-lg btn-warning">
           </form>
        </div>

      {% if data %}

          {% for i in data %}
           <div class="col-md-3" style="margin-top:50px">
             <div class="card">
               <div class="card-body">
                 <img class="card-img-top" src="{{i[2]}}">
                 <p class="text-white">{{i[0]}}</p>
                 <h4 class="text-white">{{i[1]}}</h4>
               </div>
             </div>
             </div>
          {% endfor %}

        {% endif %}
```

```
</div>
 </div>

</body>
</html>
```

Index.html

```
<!DOCTYPE
    html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>Book Recommender System</title>
   <!-- Latest compiled and minified CSS -->
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/css/bootstrap.mi
n.css" integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmST
sz/K6 8vbdEjh4u" crossorigin="anonymous">
</head>
<style>
   .text-white{ color:white
   }
</style>
<body style="background-color:black">

<nav class="navbar" style="background-color:#00a65a">
   <a class="navbar-brand">My Book recommender</a>
        <ul class="nav navbar-nav">
            <li><a href="/">Home</a></li>
            <li><a href="/recommend">Recommend</a></li>
            <li><a>Contact</a></li>
        </ul>
   </nav>
```

```html
<div class="container">
  <div class="row">
      <div class="col-md-12">
        <h1 class="text-white" style="font-size:50px">Top 50
        Books</h1>
      </div>

      {% for i in range(book_name|length) %}
        <div class="col-md-3" style="margin-top:50px">
          <div class="card">
            <div class="card-body">
              <img class="card-img-top" src="{{ image[i] }}">
              <p class="text-white">{{ book_name[i] }}</p>
              <h4 class="text-white">{{ author[i] }}</h4>
              <h4 class="text-white">Votes - {{ votes[i] }}</h4>
              <h4 class="text-white">Rating - {{ rating[i] }}</h4>
            </div>
          </div>
        </div>
    {% endfor %}

    </div>
  </div>

 </body>
 </html>
```

5.3.3 Book-recommender-system.ipynb

```python
import numpy as np

import pandas as pd

 books = pd.read_csv('books.csv')

users = pd.read_csv('users.csv')

ratings =

pd.read_csv('ratings.csv')
```

```
books['Image-URL-M'][1]

users.head()

ratings.head()

print(books.shape)

print(ratings.shape)

print(users.shape)

books.isnull().sum()

users.isnull().sum()

ratings.isnull().sum()

books.duplicated().sum()

ratings.duplicated().sum

()

users.duplicated().sum()

## Popularity Based Recommender System

ratings_with_name =

ratings.merge(books,on='ISBN')

num_rating_df = ratings_with_name.groupby('Book-
Title').count()['Book-Rating'].reset_index()

num_rating_df.rename(columns={'Book-Rating':'num_ratings'},inplace=True)

num_rating_df

avg_rating_df = ratings_with_name.groupby('Book-
Title').mean()['Book-Rating'].reset_index()

avg_rating_df.rename(columns={'Book-Rating':'avg_rating'},inplace=True)

avg_rating_df

popular_df = num_rating_df.merge(avg_rating_df,on='Book-Title')

popular_df
```

popular_df =
popular_df[popular_df['num_ratings']>=250].sort_values('avg_rating',ascending=False).head(50)

popular_df = popular_df.merge(books,on='Book-Title').drop_duplicates('Book-Title')[['Book-Title','Book-Author','Image-URL-M','num_ratings','avg_rating']]

popular_df['Image-URL-M'][0]

Collaborative Filtering Based Recommender System

x = ratings_with_name.groupby('User-ID').count()['Book-Rating'] > 200

padhe_likhe_users = x[x].index

filtered_rating =
ratings_with_name[ratings_with_name['User-ID'].isin(padhe_likhe_users)]

y = filtered_rating.groupby('Book-Title').count()['Book-Rating']>=50

famous_books = y[y].index

final_ratings = filtered_rating[filtered_rating['Book-Title'].isin(famous_books)]

pt = final_ratings.pivot_table(index='Book-Title',columns='User-ID',values='Book-Rating')

pt.fillna(0,inplace=True)

pt

from sklearn.metrics.pairwise import

cosine_similaritysimilarity_scores =

cosine_similarity(pt) similarity_scores.shape

def recommend(book_name):

index fetch

index = np.where(pt.index==book_name)[0][0]

similar_items =
sorted(list(enumerate(similarity_scores[index])),key=lambda

```
x:x[1],reverse=True)[1:5]
    data = []
for i in similar_items:
item = []
temp_df = books[books['Book-Title'] == pt.index[i[0]]]
item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Title'].values))
item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Author'].values))
item.extend(list(temp_df.drop_duplicates('Book-Title')['Image-URL-M'].values))
data.append(item)
return data
recommend('1984
')pt.index[545]
import pickle
pickle.dump(popular_df,open('popular.pkl','wb')) books.drop_duplicates('Book-Title')
pickle.dump(pt,open('pt.pkl','wb')) pickle.dump(books,open('books.pkl','wb'))
pickle.dump(similarity_scores,open('similarity_scores.pkl','wb'))
```

5.4 CONCLUSION

The development of our book recommender system marks a significant stride towards enriching the user experience within the expansive realm of literature. Through the utilization of hybrid methodologies that amalgamate diverse algorithms and techniques, we have successfully crafted recommendations that are not only precise but also varied and tailored to individual reading preferences.

In our implementation, we have highlighted the versatility and efficacy of methodologies such as cosine similarity and singular value decomposition in capturing intricate patterns inherent in user-book interactions. Additionally, the incorporation of Flask for backend development and HTML/CSS for frontend design has enabled us to construct an intuitive and visually appealing interface that fosters user engagement.

Looking forward, the potential for further refinement and expansion of our recommender system is vast. By integrating additional data sources, refining algorithms, and leveraging advancements in machine learning, we can continuously enhance the accuracy and relevance of our book recommendations. Ultimately, our objective remains to empower users in their literary exploration, facilitating a deeper connection with books that align with their interests and preferences.

CHAPTER-6

RESULTS AND ANALYSIS

6.1 RESULT ANALYSIS

In the realm of book recommendation systems, developers often face the choice between employing content-based filtering, collaborative filtering, or a hybrid approach combining both methodologies. In our project, we've opted for a hybrid approach, leveraging elements of both content and collaborative filtering.

Content-based filtering focuses on recommending books based on user ratings or preferences. It's relatively straightforward to design and computationally efficient. However, its main limitation lies in its reliance solely on the user's existing interests. The model may struggle to expand beyond the user's established preferences.

On the other hand, collaborative filtering recommends books by comparing users with similar reading habits. This approach requires no domain knowledge, as embeddings are automatically learned. It has the advantage of helping users discover new interests, even if they haven't explicitly expressed interest in those genres or topics. However, it faces challenges such as the cold-start problem, where the system struggles to recommend books that haven't been seen during training.

In our book recommendation system, we aim to combine the strengths of both content-based and collaborative filtering while mitigating their respective limitations. By doing so, we strive to provide users with personalized recommendations that not only align with their existing preferences but also introduce them to new and diverse literary experiences.

Measurement\exp.	1st	2nd	3rd
MAPE CB	0.93	0.93	0.90
MAPE CF	2.13	0.73	0.73
MAPE Hybrid	0.969	0.972	1.031

No.	CB Threshold	CF Threshold	σ
1	2.5	0	0.932
2	2.5	user rating mean	1.108
3	user rating mean	user rating mean	1.269

Fig. 6.1(a): Comparison between the three approaches

The hybrid approach will resolve all these limitations by combining both content and collaborative filtering. The main disadvantage in hybrid approach is it require high memory

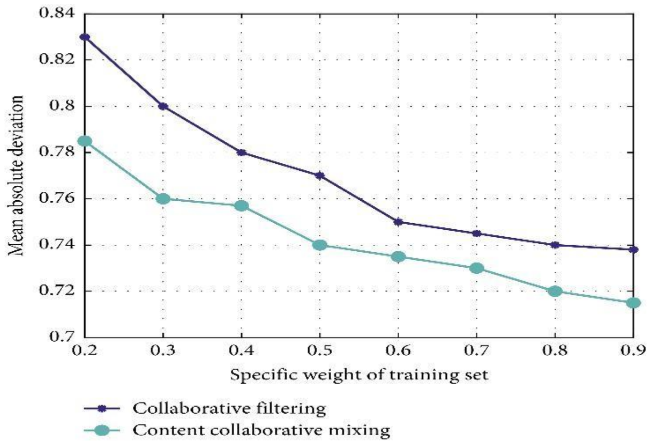
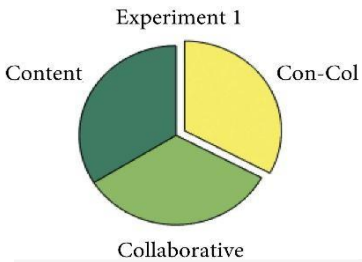
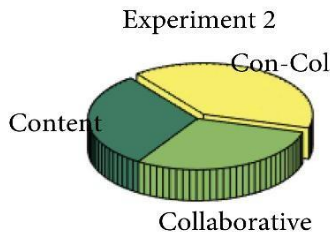


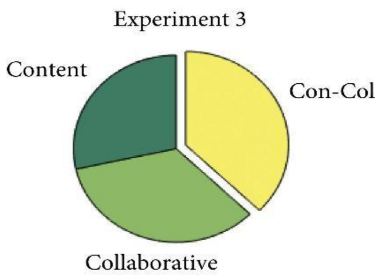
Fig. 6.1(b): Figure Showing the graph representing different approaches Mean deviation

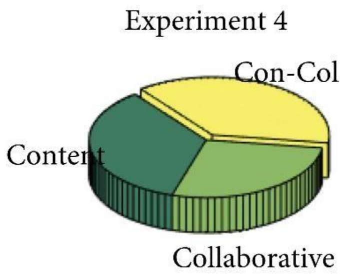
Experimental results of Different data sparsity degrees





(a) (b)





(c) (d)

Fig. 6.1(c): Performance of algorithms with different sparsity degrees.

(a) Experiment 1. (b) Experiment 2. (c) Experiment 3. (d) Experiment4

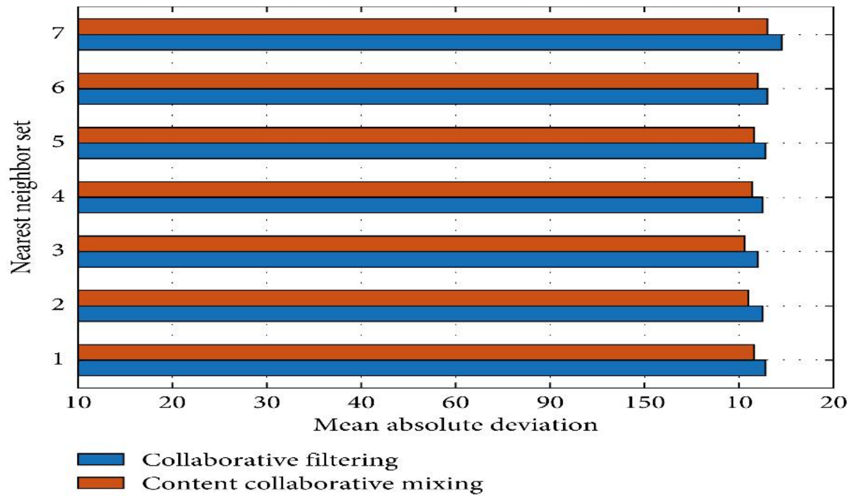


Fig. 6.1(d): Experimental results of different sizes of nearest neighbor sets.

6.2 SCREEN SHOT OF THE RESULT

Recommend button result

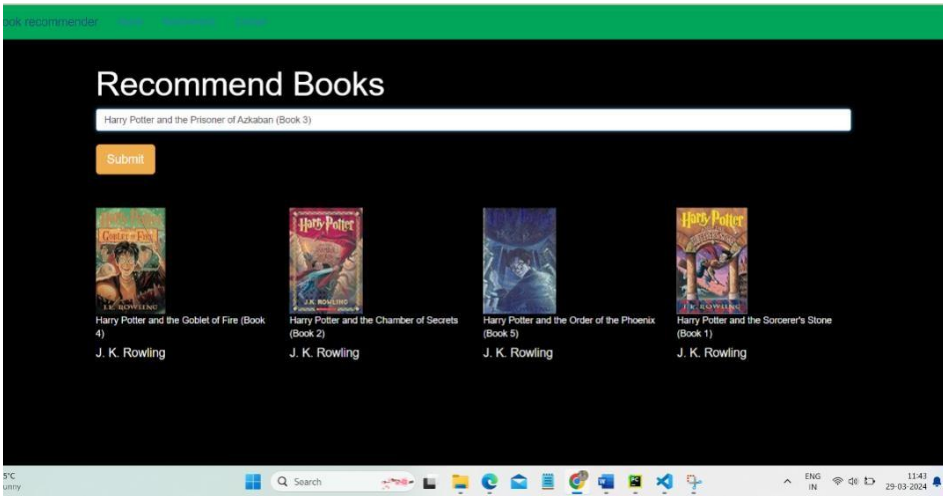


Fig. 6.2: Result of recommendation

6.3 HOME BUTTON RESULT

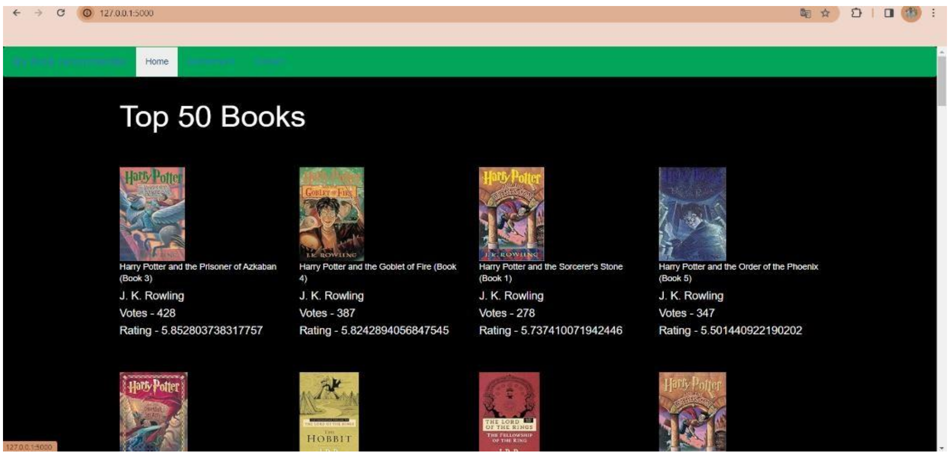


Fig. 6.3: By default, home button show top fifty from our data set

CHAPTER-7

TESTING

7.1 INTRODUCTION

In the dynamic landscape of software development, system testing emerges as a pivotal phase, marking the culmination of meticulous planning, design, and implementation efforts. As the final frontier before software deployment, system testing entails a rigorous examination of the entire software system to validate its functionality, performance, and reliability. This phase serves as a crucial checkpoint, ensuring that the software meets the specified requirements and delivers on its intended purpose.

System testing encompasses a diverse array of tests, each meticulously designed to probe different aspects of the software. From functional validation to performance benchmarking, every test serves the overarching goal of affirming the quality and readiness of the software for deployment. With the ever-increasing complexity of software systems and the heightened expectations of end-users, the importance of robust system testing methodologies cannot be overstated.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all the system elements have been properly integrated and perform allocated functions. The testing process is actually carried out to make sure that the product exactly does the same thing what is supposed to do. In the testing stage following goals are tried to achieve: -

- To affirm the quality of the project.
- To find and eliminate any residual errors from previous stages.
- To validate the software as a solution to the original problem.
- To provide operational reliability of the system.

7.2 TESTING METHODOLOGIES

There are many different types of testing methods or techniques used as part of the software testing methodology. Some of the important testing methodologies are:

- **Unit Testing:**

Unit testing is the first level of testing and is often performed by the developers themselves. It is the process of ensuring individual components of piece of software at the code level are functional and work as they were designed to. Developers in a test-driven environment will typically write and run the tests prior to the software or feature being passed over to the test team. Unit testing can be conducted manually, but automating the process will speed up delivery cycles and expand test coverage. Unit testing will also makedebugging easier because finding issues earlier means they take less time tofix than if they were discovered later in the testing process. Test Left is a tool that allows advanced testers and developers to shift left with the fastest test automation tool embedded in any IDE.

- Integration Testing:

After each unit is thoroughly tested, it is integrated with other units to create modules or components that are designed to perform specific tasks or activities. These are then tested as group through integration testing to ensure whole segments of an application behave as expected (i.e, the interactions between units are seamless). These tests are often framed by user scenarios, such as logging into an application or opening files. Integrated tests can be conducted by either developers or independent testers and are usually comprised of a combination of automated functional and manual tests.

- System Testing:

System testing is a black box testing method used to evaluate the completed and integrated system, as a whole, to ensure it meets specified requirements. The functionality of the software is tested from end-to-end and is typically conducted by a separate testing team than the development team before the product is pushed into production.

7.3 CONCLUSION

In conclusion, system testing stands as a linchpin in the software development lifecycle, playing a pivotal role in ensuring the success and reliability of software products. Through a systematic and comprehensive approach, system testing verifies the integrity of the software, affirming that it not only meets the specified requirements but also delivers a seamless and satisfying user experience. By subjecting the software to a battery of tests, ranging from functional validation to performance optimization, system testing uncovers potential flaws and discrepancies, enabling timely rectification before deployment. Moreover, system testing serves as a testament to the commitment of development teams towards delivering high-quality, reliable, and user-centric software solutions.

As technology continues to evolve and software systems become increasingly intricate, the significance of robust system testing methodologies will only grow. By embracing best practices, leveraging advanced testing tools, and fostering a culture of quality assurance, organizations can ensure that their software products meet the highest standards of excellence and exceed the expectations of end-users.

CHAPTER-8
CONCLUSION AND FUTRURE SCOPE

8.1 CONCLUSION

In this project, to enhance the accuracy, quality, and scalability of the book recommendation system, a hybrid approach is proposed, merging content-based filtering and collaborative filtering techniques. Singular Value Decomposition (SVD) is employed as a classifier, while Cosine Similarity is utilized as a measure of similarity within the proposed methodology. The effectiveness of this hybrid approach is evaluated by implementing it alongside existing pure approaches on three distinct book datasets, and the results are compared.

Comparative analysis reveals that the proposed hybrid approach outperforms the pure approaches in terms of accuracy, quality, and scalability of the book recommendation system. Additionally, the computational time required for the proposed approach is observed to be lower compared to the other two pure approaches.

Overall, the findings suggest that the hybrid approach combining content-based and collaborative filtering, with the integration of SVD and Cosine Similarity, offers significant improvements in the effectiveness and efficiency of book recommendations.

8.2 FUTURE SCOPE

In the proposed approach, ratings have been considered as a key factor. However, there is potential for future enhancements by incorporating additional factors such as the age of the user. Age can significantly influence book preferences, as individuals tend to gravitate towards different genres and themes at different stages of their lives. For instance, during childhood, preferences may lean towards picture books or young adult fiction, while in adulthood, readers may prefer literary classics or non-fiction.

Furthermore, in future iterations of the approach, attention should be given to optimizing memory requirements. As datasets grow larger and more complex, efficient memory management becomes crucial to ensure smooth operation and scalability of the recommendation system.

While the proposed approach has been tested on various book datasets, there remains room for expansion. Future work could involve implementing the approach on additional datasets such as those from platforms like Goodreads or Amazon Kindle. Evaluating the performance of the system across diverse datasets would provide valuable insights into its effectiveness and applicability in different contexts.

REFERENCES

1. Hirdesh Shivhare, Anshul Gupta and Shalki Sharma (2015), "Recommender system using fuzzy c-means clustering and genetic algorithm based weighted similarity measure", IEEE International Conference on Computer, Communication and Control.
2. Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), "A BooksRecommender System: MOVREC", International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3.
3. RyuRi Kim, Ye Jeong Kwak, HyeonJeong Mo, Mucheol Kim, Seungmin Rho, KaLok Man, Woon Kian Chong (2015),"Trustworthy BooksRecommender System with Correct Assessment and Emotion Evaluation", Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II.
4. Zan Wang, Xue Yu, Nan Feng, Zhenhua Wang (2014), "An Improved Collaborative Books Recommendation System using Computational Intelligence",Journal of Visual Languages & Computing,Volume 25, Issue 6.
5. Debadrita Roy, Arnab Kundu, (2013), "Design of Books Recommendation System by Means of Collaborative Filtering", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 4.
6. G.C. Capelle Veen, C. Amrit, D.M. Yazan, W.H.M. Zijm, "The recommender canvas: A model for developing and documenting recommender system design". Expert systems with applications, pp. 97-117, 2019.
7. F. Ricci, L. Rokach, B. Shapira, Introduction to 1Recommender Systems Handbook. Boston, Massachusetts, United States of America: Springer, 2010.
8. J. Erickson and S. Wang. (2017, June) www.alizila.com. [Online]. https://www.alizila.com/at-alibaba-artificial-intelligence-is-changinghow-people- shop-online/
9. I. MacKenzie, C. Meyer, and S. Noble. (2013, Oct.) www.mckinsey.com. [Online].https://www.mckinsey.com/industries/retail/our-insights/howretailers-can-keep-up-with-consumers.
10. C.A. Gomez-Uribe and N. Hunt, "The Netflix Recommender System: Algorithms,Business Value," ACM Transactions on Management Information Systems (TMIS), vol. VI, no. 4, p. 7, Jan. 2016.
11. J. Schmidt, M.R.G. Marques, S. Botti, et al. "Recent advances and

applications of machine learning in solid-state materials science”. npj Comput Mater 5, 83, 2019.

12. F. Ricci, L. Rokach, B. Shapira, “Recommender Systems: Introduction and Challenges”. In: Ricci F., Rokach L., Shapira B. (eds) Recommender Systems Handbook. Springer, Boston, MA, 2015.
13. F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, “Recommendation systems: Principles, methods and evaluation”. Egyptian Informatics Journal, Volume 16, Issue 3, pp. 261-273, November 2015.
14. P. Lops, M. Gemmis, G. Semeraro, “Content-based Recommender Systems: State of the Art and Trends”. In book: Recommender Systems Handbook, pp. 73- 105, 2011.
15. E. Cano and M. Morisio, “Hybrid Recommender Systems: A Systematic Literature Review”. Intelligent Data Analysis, vol. 21, no. 6, pp. 1487-1524, 2017.
16. H. Mak, I. Koprinska, and J. Poon, "INTIMATE: A Web-Based BooksRecommender Using Text Categorization," Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence, 2003.
17. L.H. Li, R. Hsu, F. Lee, “Review of Recommender Systems and Their Applications”. Computer Science, 2012.
18. P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations," Mountain View, California, 2016.
19. A. Bellogín, P. Castells, I. Cantador, “Neighbor Selection and Weighting in User-Based Collaborative Filtering: A Performance Prediction Approach”. ACM Transactions on the Web, No. 12, 2014.
20. S.C. Stephen, H. Xie, and S. Rai, “Measures of Similarity in MemoryBased Collaborative Filtering Recommender System: A Comparison”, 4th Multidisciplinary International Social Networks Conference, 2017.
21. G.S. Milovanovic, “Hybrid content-based and collaborative filtering recommendations with {ordinal} logistic regression (1): Feature engineering”, Data Science Central, 2017.

G.S. Milovanovic, “Hybrid content-based and collaborative filtering recommendations with {ordinal} logistic regression (1): Feature engineering”, Data Science Central, 2017

Book Recommendation System Using Machine Learning

O. Venkata Ashok
Department of Information Technology
Narasaraopeta Engineering College, Narasaraopet, India

B. Penchala Pratap
Department of Information Technology
Narasaraopeta Engineering College, Narasaraopet, India

O. Jaya Krishna
Department of Information Technology
Narasaraopeta Engineering College,Narasaraopet, India

Shaik. MohammedJany
Asst.Professor Department of Information Technology
Narasaraopeta Engineering College,
Narasaraopet, India

ABSTRACT

A suggestion system's essential objective is to form consumers' lives simpler by intellectuals recommending items to them. These days, the sum of data on the web is developing quickly, and individuals require as it were many apparatuses to discover and get the proper data. The term suggestion framework alludes to one such instrument. Suggestion motors show clients with item proposals that are most related to them. Websites that advance books online presently confront competition from one another in a number of ways. A recommendation structure that recommends books based on the customer's interface is one of the foremost viable ways to extend advantage and keep shoppers. Hence, the most objectives of this extend are to energize and assist those whoare inquisitive about perusing as well as to have an impact on those who are shaping perusing propensities. By making a book suggestion framework, we trust to assist individuals select the correct book for their interface and in this way propel them to studied more. We think we are able select the perfect book for a individual based on their inclinations and the information from different perusers with the assistance of information sets and machine learning. As a result, we utilize a collaborative sifting procedure here.

I. INTRODUCTION

A recommendation framework, also known as a recommendationmodel, could be viewed as an interpretation of data sifting that generates recommendations determined by attempting to ascertain the user's preferences. Nowadays, these frameworks are widely used in a variety of businesses, countingattire, eateries, nourishment, motion picture theatres, music, books, recordings, and other utilities. They have developed in notoriety in later a long time. These frameworks assemble client behavior and inclination information, which they at that point utilize to refine the proposals they give going forward.

A film is simply a parcel of reality. There are many different typesof motion pictures, including beguilement, educational book, children cartoons, activities, and frightful books. Motion picture sorts like comedy, tension, liveliness, activity, and so on make it straightforward to perceive between different sorts of books. There are several methods to differentiate between books, such by the director, language, or year of release. Online book viewing allows us to browse a variety of book on our list of top picks. The use of Motion picture Recommendation Frameworks prevents the need for time- consuming search engine optimization by allowing us to identify our favoured book from a wide variety of choices. Consequently, the system for prescribing motion pictures to us must be greatly reliable and donate us proposals for the book that best suit our tastes. A expansive numbered of industries are using recommendation frameworks to improve user purchasing experiences and increase client connection. Frameworks for proposals have a few favourables, the foremost vital being client fulfilment and income. Motion picture Proposal framework is exceptionally capable and imperative framework. But, due to the issues related with unadulterated collaborative approach, motion picture suggestion frameworks too endure with destitute proposal qualityand adaptability issues.

Problem Statement:
The user's proposal of a motion picture is the project's primary objective. giving buyers of online benefit suppliers suitable substance chosen from a library of things that are both significantand unessential.

The projects' goal:

Enhance the quality of the motion picture proposal framework; Increment the suggestion system's exactness

- Improving Adaptability.
- Making strides the encounter for clients.

1.1 The Project's Scope

This project's objective is to deliver individuals exact book recommendations. The project's objective is to increase the grade of book suggestion framework, some as precision, grade and versatility of framework than the immaculate approaches. Suggestion frameworks are utilized as data sifting devices in social organizing destinations in arrange to induce rid of the informationover-burden. This can be finished by combining collaborative sifting with content-based sifting in a cross-breed way. Subsequently, there's an extraordinary bargain of room for investigate in this zone to enhance the versatility, accuracy, and calibre of motion picture proposal frameworks. The book proposal framework may be a noteworthy and exceedingly successful framework. However, book suggestion frameworks moreover have concerns with versatility and low proposal quality as a result of the impediments with the unadulterated collaborativeapproach.

1.2 Approach for book Recommendation

The hybrid approach proposed an integrator strategy by combining fluffy k-means clustering Method and hereditable calculation-based weight closeness degree to develop a motion picture proposal framework. The proposed motion picture recommendation framework gives better close calculations and quality compared to the current book proposal framework; nonetheless, the suggested proposal structure requires more computing time than the current recommendation system. By using the clustered information concentrates on an input dataset, this problem can be resolved. Enhancing the adaptability and caliber of the book recommendation system is the recommended course of action. In order for the processes to complement one another, we employ a cross-breed strategy by tying together collaborative filtering and content-based filtering. To reduce the motion picture recommender motor's computation time and to efficiently and quickly calculate the similarity between the various theaters in the dataset, we utilized co-sine closeness degree.

1.3 Agile Methodology

1. **Gathering the information sets:**

Collecting overall desired informationderived from the Kaggle website. We need book.csv, ratings.csv, and users.csv for this project.

Analyze the Data: Make beyond any doubt that the gathered information sets are correct and analyzing the information within the csv records. i.e. Determining whether all the columns areas are visible in the information sets.

2. **Algorithms:**

In our project we had as it were two calculations The machine learning proposal model is built using two methods: one is cosine similarities, and the other is single admired degradation.

3**. Education and Assessment:**

when the calculation has finished being executed. To get the desired outcome, we should set up the model. We've tried it a few times; the show recommends a variety of book to a wide range of clients. Project improvements: We will implement various calculations and tactics for better recommendation in the subsequent plan.

II LITERATURE SURVEY

From years, a large number of recommendation systems have developed utilizing hybrid, content-based, or collaborative filtering techniques. Different forms of big data and machine learning have been used to implement these systems algorithmic rules.

2.1 K-Nearest Neighbor and K-Means Clustering for a book Recommendation System

A recommendation framework gathers data regarding the user's preferences, either definitely or expressly on diverse things like film A verifiable securing within the improvement of film proposal framework employments the user's conduct whereas observing the book. alternatively, an express securing within the advancement of book proposal uses the user's historical assessments. The other support procedure that is used within the proposal framework improvement is clustering. K-Means Clustering together with K-Nearest Neighbor is applied to the film focal point dataset in order to obtain the best-optimized result. In the existing procedure, the information is dispersed and leads to a high number of clusters, whereas in the proposed method the information is accumulated and results in a lower number of clusters.Within the suggested conspiracy, the motion picture suggestion mechanism is optimized. The suggested recommender system predicts a user's preference for a book based on a variety of factors. The recommender architecture operates under the premise that people share a shared preference or inclination. These clients will influence one another's presumptions. This handle has a reduced RMSE and optimizes the procedure.

2.2 Collaborative Filtering-Based Product Suggestion Framework:

Collaborative filtering frameworks look at a user's behavior and preferences to predict what they would like based on similarity with other customers. User-based and item-based suggestions are the two categories of collaborative filtering frameworks.

1. **customer-based filtering:**

When it comes to designing customized frameworks, user-based preferences are incredibly prevalent. The user's preferences drive this process. First, the clients rate a few books on a scale of 1 to 6. These evaluations may be conclusive or express. The evaluation of the item Unequivocal Express Unequivocal Synonyms involves the customer indicating their satisfaction with the object on a specific scale or giving varying degrees of approval. Frequently explicit evaluations are difficult to gather as not each client is much curious about giving reviews. In this structure, we assemble understood evaluations based on their conduct. For occasion, in case a client purchases a product more than once, indicating a tendency in the right direction. When considering cinematic frameworks, we can deduce that on the off chance that a client observes the whole film, he or she has a few amiabilities to film. Note: there are no clear rules in deciding verifiable evaluations. Following, for each client, we to begin with discover a few characterized numbers of closest neighbors. We calculate relationship between users' evaluations utilizing Pearson Relationship calculation. The presumption that in the event that two users' appraisals are exceedingly related, at that point these two clients must appreciate comparable things anditems is utilizedto prescribe things to clients.

2.**Product based sifting**: Not at all like the user-based sifting strategy, things-centered emphasizes the intimacy between the clients and the products, such as the clients' own instep. The most comparable items are computed beforehand. When it comes to recommendations, the items that are closest to the desired item are suggested to the customer.

III SYSTEM NECESSITIES SPECIFICATION

This includes both the equipment and software prerequisites required for the project and point by point clarification of the specifications.

3.1 Hardware prerequisite:

- A minimum of 8GB RAM.
- A Graphic card with 2 gigabytes.
- AWindows or Linux OS PC with a processor speed of 1.7–2.4 GHz

3.2 Program Specifications

Text editors (WebStorm/VS-code) Anaconda distribution packages (PyCharm Editor) and libraries pertaining to Python

3.3.1 Software Anaconda distribution Requirements

For logical measurement (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), Anaconda software is a free and open-source distribution for Python programming languages that indicates to reorganize package management framework and arrangement. The bundle managing framework conda is responsible for overseeing bundle forms. The Anaconda transport includes data-science packages compatible with Linux, Macintosh, and Windows operating systems.3.

3.3.2 Python libraries:

Some Python libraries that are used for data analytics are needed for the computation and analysis. A few packages are required, including the Flask framework, Numpy, pandas, Matplotlib, and Sklearn.

Sklearn: With support for support vector machines, random forests, slope boosting DBSCAN, K-means, and other methods of classification, clustering, and regression, Sklearn is designed to work with NumPy and SciPy, two popular Python libraries for science and mathematics.

NumPy: NumPy has the potential to be a versatile array processing tool. The multidimensional array objects and devices for interacting with these clusters are highly performant. It is the fundamental Python bundle for logical calculations.

Pandas: One of the first popularly used Python libraries for data science applications is called Pandas. It provides high-end functionality, easy-to-use architecture, and data analysis tools. Pandas provides in-memory 2D table questions called Data frames, unlike the NumPy library which provides items for multi-dimensional clusters.

Flask: It might be a WSGI web application framework that is lightweight. It is intended to start quickly and simply and have the ability to grow to more complicated applications. It began as a simple Werkzeug wrapper.

IV SYSTEM DESIGN AND ANALYSIS

The proposed system architecture is as follows:

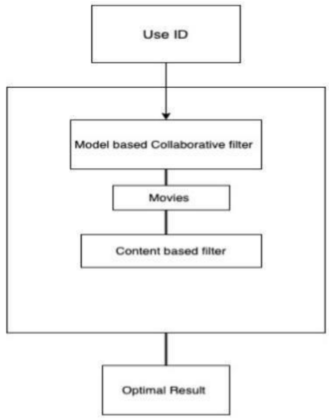


Fig: -4.1 Architecture for hybrid approach

Different lists of book are prescribed for each unique individual. When a client enters their client ID, two distinct procedures are used within the extension; each will prescribe a set of books to the particular client by combining the two sets of books based on the client; the half breed demonstrate will then suggest a single list of books to the client.

Activity Schematic:

Figure -4.2: Activity schematic

The list of books is presented to the user as recommendations after they have logged in by inputting their user ID, which is contained in the CSV file and ranges from 1 to 50.

4.3 Data Transfer:

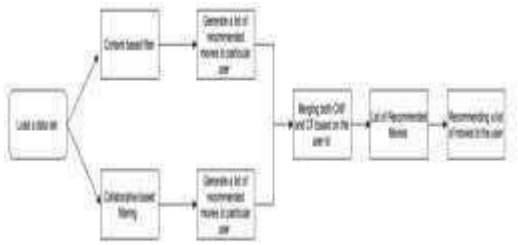


Figure: -4.3 Diagram of Data Flow

The first step in building a model is to load the data sets. For this project, the information sets that are needed are customers.csv, ratings.csv, and products.csv. You can acquire all of these data sets on the Kaggle.com website. Basically, this project has two forms of filtering built in: collaborative and substance-based. Each type generates a list of books for a particular client; by combining the two based on the client ID, a final list of books is recommended to that particular customer.

V IMPLEMENTATION

The proposed system uses several methods and algorithms to implement the hybrid approach.

5.1 Cosine Similarity: The cosine of the point that separates two non-zero vectors in an inward item space can be computed as a measure of the degree of similarity between them.

Formula:

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\,\|\vec{b}\|} = \frac{\sum_{1}^{n} a_i b_i}{\sqrt{\sum_{1}^{n} a_i^2}\sqrt{\sum_{1}^{n} b_i^2}}$$

where, $\vec{a}\cdot\vec{b} = \sum_{1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ is the dot product of the two vectors.

5.2 Singular Value Decomposition (SVD):

Let A be a n*d matrix with accompanying singular values (σ1, σ2, σr) and singular vectors (v1, v2,..., vr). Consequently, ui = (1/σi).The left singular vectors are represented by Avi, for i = 1, 2,..., r, and according to Theorem 1.5, A may be broken down into a sum of rank one matrices a

$$A = \sum_{i=1}^{r} \sigma_i \mathbf{u_i} \mathbf{v_i}^T .$$

First, we establish a basic lemma that states that if Av = Bv for every v, then two matrices, A and B, are equal. According to the lemma, a matrix A can be conceptualized as a transformation that transfers a vector v onto Av in the abstract.

VI OUTCOMES AND DISCUSSION

Since our venture is film suggestion framework .one can create a product proposal framework by utilizing either cooperative filtering, content-based filtering, or a combination of the two. We have developed a hybrid strategy for this project, which combines collaborative filtering with content. There are pros and disadvantages to both approaches.The content-based filtering system bases its recommendation of books on the opinions or preferences of the client, much like a prescription drug.
Positives: It is easy to plan and computes quickly.

Cons: The demonstrate can, in a sense, make recommendations based on the client's current interface. Stated differently, the show's growth potential on the current customer interface is limited. A comparison of similar users is suggested in the collaborative filtering approach.

Advantages: None require space information since the embeddings are naturally learned. The show can offer assistance clients find modern interface. In separation, the ML framework may not knowthe client is curious about a given thing, but its demonstrate mightstill suggest it since comparable clients are curious about thatitem

Drawbacks: The dot item of the analyzing embeddings is the forecast of the demonstrate for a certain combo. Therefore, in the unlikely event that an item is not observed during preparation, the framework is unable to integrate it and cannot
Look into the demonstration with this item. This problem is commonly referred to as the "cold-start problem."

Fig: -6.1 Comparison between the three approaches
By integrating collaborative filtering with content-based filtering, the combinational technique will overcome all of these problems. The primary drawback of the hybrid technique is its high memory requirements.

Screen shot of the result: Recommend button result

Home button result: By default, home button shows top fifty books from our data set

VII TESTING

In reality, system testing is a collection of many tests with the primary goal of completely work out the computer-based framework. In spite of the fact that each test features a distinctive reason, all work to confirm that all the framework components have been legitimately coordinates and perform distributed functions. The testing handle is really carried out to create beyond any doubt that the item precisely does the same thing what is assumed to do. Within the testing organize taking after objectives are attempted to attain:

- To assert the quality of the extend.
- To find and dispose of any remaining blunders from past stages.
- To approve the computer program as a arrangement to the firstissue.
- To supply operational unwavering quality of the system.

7.1 Testing Methodologies

There are numerous diverse sorts of testing strategies or methods utilized as portion of the program testing technique. A few of the imperative testing techniques are:

Component Testing:

ComponentTesting is the first step in the testing phase, which is carried out on a regular basis by the developers alone. It is a technique to ensure that individual components of a small computer program function as intended and are helpful at the code level. In a test-driven environment, engineers often type in and execute the exams ahead of time on the computer program or indicate that they are being given to the test group. Although component testing can be done physically, automating the process will increase test scope and speed up conveyance cycles. Component testing will help ease the burden of investigation since, as earlier identification of problems suggests, they settle more quickly thanon the off chance that they were found afterward within the testing prepare. Test Cleared out may be a tool that permits progressed analyzers and developers to move cleared out with the speediest test robotization apparatus inserted in any IDE.

Integration Testing:

Following a component's exhaustive testing, it collaborates with other units to create modules or components that are meant to be used in specific tasks or activities. After that, they are tested collectively through integration testing to ensure that all of the application's components function as expected (i.e., that all of the units' intuitive features work together). Client scenarios that involve opening records or completing an application frequently accept these tests. The majority of the time, connected testing consist of a combination of automated functional and manual tests and can be carried out by either designers or autonomous analyzers.

System Testing:

System testing is a type of dark box testing that is used to evaluate the completed and coordinated framework overall to make sure all requirements are met. A separate testing group tests the program's usefulness from beginning to end before the advancement group puts it into production. This process has been done recently.

VIII CONCLUSION AND FUTRURE SCOPE

8.1 Conclusion

In this project, to progress the exactness, grade and adaptability
of product suggestion framework, a Crossover approach by binding together substance depend filtering and collaborative filtering; utilizing Particular Esteem Deterioration (SVD) as a classifiers and Co-sine Similarity is displayed within the suggested strategy. Existing unadulterated procedures and suggested crossover approach is executed on Three distinctive product datasets and the comes about are collate among them. Relative comes about delineates that compared to the pure approaches, the Suggested approach exhibits improvements in the accuracy, caliber, and flexibility of the book suggestion framework. The recommended processes' calculated times are also less than those of the other two flawless methods.

8.2 Future scope

In The suggested approach has taken into account a variety of product types, but in the coming days, it will be prepared to take into account the client's age in accordance with their age-related book preferences as well as modifications. For instance, when we were in elementary school, we preferred animated book over other books. In the following days, there will be work done on the memory requirements of the recommended technique. Here, the proposed method has been implemented on various film datasets. Additionally, it can be estimated in the near future and applied to the Netflix and Film Loving databases.

REFERENCES

[1] Hirdesh Shivhare, Anshul Gupta and Shalki Sharma (2015), “Recommender system using fuzzy c-means clustering and genetic algorithm based weighted similarity measure”, IEEE International Conference on Computer, Communication and Control.

[2] Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), “A book Recommender System: MOVREC”, International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3.

[3] RyuRi Kim, Ye Jeong Kwak, HyeonJeong Mo, Mucheol Kim, Seungmin Rho, Ka Lok Man, Woon Kian Chong (2015),“Trustworthy book Recommender System with Correct Assessment and Emotion Evaluation”, Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II.

[4] Zan Wang, Xue Yu*, Nan Feng, Zhenhua Wang (2014), “An Improved Collaborative book Recommendation System Using Computational Intelligence”, Journal of Visual Languages & Computing, Volume 25, Issue 6.

[5] Debadrita Roy, Arnab Kundu, (2013), “Design of book Recommendation System by Means of Collaborative Filtering”, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 4.

BB21.pdf F

ORIGINALITY REPORT

10%	5%	4%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	ir.juit.ac.in:8080 Internet Source	3%
2	ijsrst.com Internet Source	1%
3	Submitted to Sheffield Hallam University Student Paper	1%
4	Submitted to NCG Student Paper	1%
5	Submitted to CSU, Fullerton Student Paper	1%
6	Shreya Agrawal, Pooja Jain. "An improved approach for movie recommendation system", 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017 Publication	1%
7	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	1%

8	"Progress in Computing, Analytics and Networking", Springer Science and Business Media LLC, 2018 Publication	<1 %
9	yusrilhasanuddin.medium.com Internet Source	<1 %
10	Submitted to University of Westminster Student Paper	<1 %
11	Nikhil S Patankar, Yogesh S Deshmukh, Rameshwar D Chintamani, K Vengatesan, Nitin L Shelake. "An Efficient Machine Learning System using Sentiment Analysis for Movie Recommendations", 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), 2022 Publication	<1 %
12	www.upgrad.com Internet Source	<1 %
13	Shreya Agrawal, Pooja Jain. "Rising fascination towards exploration of recommendation systems", 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 2016 Publication	<1 %
14	Robin Burke. "Robustness of collaborative recommendation based on association rule	<1 %

mining", Proceedings of the 2007 ACM conference on Recommender systems - RecSys 07 RecSys 07, 2007

Publication

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		