



# Reading Recommendation Platform Using Machine Learning

Ms. Tejaswini Patil, Dr. Mamata S Kalas

Student, Professor

Kolhapur Institute of Technology's College of Engineering, Kolhapur  
INDIA

**Abstract:** With the exponential growth of digital content, reading recommendation systems have become essential tools for helping readers discover relevant and personalized content. This project involves building a reading recommendation system using Python, aimed at recommending books tailored to users' reading history and individual preferences. The system leverages collaborative and content-based filtering techniques to provide accurate and dynamic recommendations. A user-friendly interface enables readers to input preferences, receive suggestions, and explore related titles. This report details the system's design, implementation, and evaluation, highlighting its potential to enhance the reading experience, support libraries and online platforms, and promote diverse literature. The integration of intelligent recommendation technology represents a step forward in automating and personalizing the content discovery process.

**Index Terms** - book, model-based approach, collaborative filtering.

## I. INTRODUCTION

Reading recommendation systems have become increasingly important in the digital age, helping readers navigate vast libraries of content by offering tailored suggestions. These systems aim to enhance the user experience by understanding individual preferences and providing recommendations that align with their reading habits and interests.

This project focuses on developing a responsive and easy-to-use reading recommendation platform that leverages user interactions such as ratings and selected genres to generate personalized book suggestions. Inspired by leading platforms like Goodreads and Kindle, the system applies collaborative filtering techniques to identify and recommend books that match user profiles and interests.

Administrators are able to categorize and add new titles to the system, while users can browse recommended titles, provide ratings, and leave feedback. This user input refines the recommendation process over time, allowing the system to improve the quality and relevance of its suggestions.

By streamlining book discovery, this system aims to reduce the time and effort users spend searching for reading material. While the technology offers clear advantages, it also raises concerns related to data protection and ethical use, which must be addressed to maintain user trust and system transparency.

## II. LITERATURE SURVEY

According to study [1], an online book recommendation system was developed using an enhanced collaborative filtering approach that integrates cosine similarity for measuring user-item relationships and a quick sort algorithm to improve computational efficiency. The system was designed following the Object-Oriented Analysis and Design Methodology (OOADM) and implemented using the Django framework with a NoSQL database to support scalability and flexibility. Experimental results demonstrated improved recommendation accuracy and reduced processing time. Although content-based, collaborative, and hybrid recommendation techniques have been widely studied, challenges such as scalability, data sparsity, and privacy remain significant. The proposed approach addresses these issues by optimizing collaborative filtering techniques and employing scalable system architecture, thereby enhancing system reliability and performance.

According to study [2], the tourism industry plays a major role in the global economy, with increasing reliance on internet platforms for accessing travel-related information. However, the abundance of online content often leads to information overload, and the absence of user profiling in many tourism websites results in non-personalized recommendations. To overcome this limitation, the study proposes a Tourist Recommendation System (TRS) that combines item-based collaborative filtering and content-based filtering integrated with a Naïve Bayes classifier. User profiles and questionnaire-based inputs are utilized to capture individual preferences and interests, enabling personalized tourist spot recommendations. The study also addresses challenges such as cold-start and data sparsity by employing the Naïve Bayes classifier for prediction under limited data conditions. Experimental results indicate improved recommendation accuracy, and the authors suggest that hybrid recommender systems and advanced machine learning techniques could further enhance system performance in future work.

According to study [3], recommendation systems play a crucial role in suggesting relevant items to users by analyzing their behavior, preferences, and rating history. Among various recommendation techniques, collaborative filtering is identified as one of the most effective approaches due to its ability to leverage collective user behavior. The study proposes a novel user-based collaborative filtering algorithm that predicts item ratings through a two-phase process: identifying similar users using a mathematical similarity measure and computing predicted ratings based on these similarities. The similarity between users is calculated using rating differences and converted into a similarity percentage. Experimental evaluation conducted on a sample dataset demonstrates satisfactory prediction accuracy, achieving a Mean Absolute Error (MAE) of 0.80. The results indicate that the proposed method is effective for rating prediction, with potential for scalability and further validation on larger and real-world datasets.

According to this study [4], presents the development of an online book recommendation system that integrates cosine similarity and a quick sort algorithm within an enhanced collaborative filtering (CF) framework. The system architecture follows the Object-Oriented Analysis and Design Methodology (OOADM), and it is implemented using Django for the web framework along with a NoSQL database to ensure flexibility and scalability. The suggested model shows enhanced results, offering greater accuracy and improved efficiency. While traditional recommendation approaches such as content-based, collaborative filtering, and hybrid systems have been widely explored, they still face several limitations, including scalability, data sparsity, and privacy concerns. Addressing these challenges requires ongoing research to develop more robust methods that not only improve recommendation quality but also uphold data security and user trust. This work contributes by offering an optimized CF technique supported by scalable infrastructure and a methodological design that enhances the overall system reliability.

In [5], previous research highlights the value of book recommendation systems in various domains such as e-commerce, social media, and digital libraries, where users often struggle to find content that aligns with their interests. Traditional recommender systems based on collaborative filtering, content-based techniques, or hybrid models have contributed to simplifying the book discovery process. However, these systems frequently fall short in delivering personalized results due to their reliance on generalized group preferences or exact content similarity, leading to user dissatisfaction.

To address these limitations, a system known as PBRecS was proposed, emphasizing personalization through social context. This approach utilizes social interaction data, specifically the friendship networks formed on platforms like LibraryThing, to enhance recommendation quality. Instead of relying solely on global user data, PBRecS filters suggestions based on books liked or owned by a user's friends with similar interests. Additionally, the system incorporates word-correlation techniques to allow partial tag matching, enabling it to uncover books with related themes even when exact matches aren't present.

An empirical evaluation conducted using real-world data from LibraryThing demonstrated that PBRecS not only improved the personalization of recommendations but also surpassed the performance of existing systems such as Amazon's and LibraryThing's own recommenders. These findings support the potential of integrating social data and flexible content matching to create more effective and user-centric recommendation systems.

As per [6], traditional book recommendation systems typically generate suggestions based on similarities between users or the attributes of items. However, recent work introduces a temporal dimension to better reflect evolving user preferences. By tracking how frequently users engage with specific books over time, a dynamic counter is maintained for each item, influencing its likelihood of being recommended. This system avoids dependence on similar users' ratings and instead uses data from all users to enhance diversity. Content-based filtering and time-aware updates further ensure that recommendations remain relevant and aligned with the user's current interests, improving overall system effectiveness.

### III. PROBLEM DEFINITION

To design and implement a reading recommendation system using Machine Learning in Python, with the ability to:

- Recommend books by analyzing user interests and previous interactions.
- Recommend top-rated and category-specific books using collaborative filtering.
- Continuously improve recommendations through user feedback and ratings.

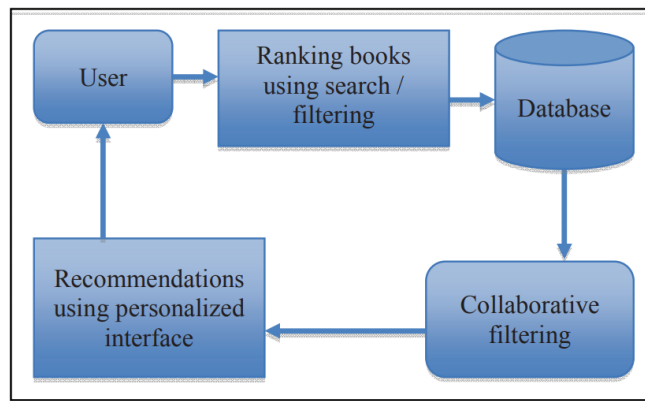
With the rapid expansion of digital libraries and online reading platforms, users often struggle to find content suited to their interests. Traditional browsing methods are inefficient and time-consuming. Integrating machine learning into recommendation systems offers a smarter, personalized, and faster way to discover relevant reading material.

### IV. OBJECTIVES

1. To study existing book recommendation methods.
2. To develop and implement a collaborative filtering algorithm using machine learning for reading recommendation.
3. To design an unsupervised machine learning approach for recommendation generation.
4. To perform a comparative analysis of different machine learning algorithms used for

### V. PROPOSED SYSTEM

After learning user preferences, the system generates personalized book recommendations using both collaborative filtering and machine learning techniques. Initially, a KNN-based collaborative filtering approach identifies books with similar user rating patterns. To improve the effectiveness of the recommendation process, additional machine learning techniques such as SVD, Random Forest, and XGBoost were incorporated and evaluated. The recommendations generated by these models are compared individually and in hybrid combinations to determine the most effective recommendation strategy. Performance is evaluated using Precision, Recall, F1-Score, and Confusion Matrix analysis.



### A. Collecting Data

The dataset used in this work was obtained from publicly available online sources and is composed of three files: user preferences, book ratings, and book information. The user preferences file represents users' reading interests, while the ratings file contains explicit feedback provided by users for different books. The book information file includes essential metadata such as book titles and categories. Prior to model training, the data was preprocessed to handle missing values and ensure consistency across all datasets for effective recommendation generation.

### B. Learning User Preferences

The system learns user preferences by examining how users interact with different books. Ratings provided by users are analyzed to identify reading interests and preference patterns. By comparing these patterns across users and items, the system determines similarities that help in understanding user behavior. This process enables the model to build a preference profile for each user, which is later used for recommendation generation.

### C. Providing Recommendations

After learning user preferences, the system provides recommendations.

In my work I decided to use collaborative filtering for generating the recommendations. Collaborative filtering is an approach to make recommendations on different items for users by collecting a bunch of information about their preferences. As it was mentioned before the idea of a collaborative filtering is that if two users have same preferences on a particular item then most likely that they will have same opinions on other items rather than with some other random user. A collaborative filtering has different types of realizations. One approach relies on neighborhood-based algorithms, which determine the similarity between items.

Collaborative filtering filters data using other people's suggestions. Here the idea is that the, people who liked some items in the past will probably like again in future.

### D. Hybrid Recommendation Framework

To improve the recommendation quality and evaluate the effectiveness of different machine learning approaches, the proposed system was extended beyond the traditional KNN-based collaborative filtering model. In addition to KNN, advanced machine learning techniques such as Singular Value Decomposition (SVD), Random Forest (RF), and XGBoost were implemented and analyzed.

The recommendation performance of individual algorithms as well as hybrid combinations including KNN+SVD, SVD+RF, KNN+RF, SVD+XGBoost, KNN+SVD+RF, and RF+XGBoost was evaluated using Precision, Recall, F1-Score, and Confusion Matrix analysis. The hybrid models combine the strengths of multiple algorithms, enabling the system to capture both similarity-based and predictive patterns present in user-book interactions.

The confusion matrix analysis was used to examine the classification capability of each model by measuring True Positives, True Negatives, False Positives, and False Negatives. This evaluation helped identify the most effective recommendation strategy and provided insights into the strengths and limitations of individual and hybrid models.

Experimental results demonstrated that machine learning-based hybrid approaches achieved improved recommendation performance compared to the traditional collaborative filtering method. The comparative analysis provides a comprehensive understanding of how different algorithms contribute to recommendation accuracy and user satisfaction.

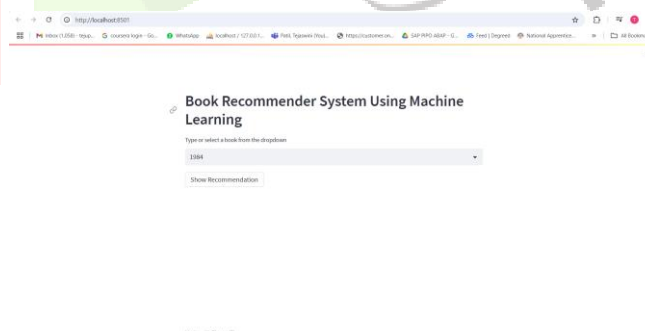
There are two approaches for implementing collaborative filtering:

Model based approach uses either item-based approach or user-based approach. Users perform the main role in the user-based approach. If some customers have similar tastes, then they join a group. Items are recommended to users based on the assessment of items by others in the same group who have common tastes. While implementing a methodology, the similarity between various items is known by using a similarity measure that is most suitable, to find the user ratings to the items that are not previously rated by him/her. Rather than depending on the person who is most similar, the predicted rating value is based on the weighted average of multiple people's recommendations. Weight given to the ratings of a person is known by the comparison between the person and the target user for whom the prediction should be made. The pearson correlation coefficient may be used as a correlation measure. In memory-based approach we build collaborative filtering models by using machine learning algorithms. In this we find the ratings of an item that user might give.

Model based approach uses either item-based approach or user-based approach. Users perform the main role in the user-based approach. If some customers have similar tastes, then they join a group. Items are recommended to users based on the assessment of items by others in the same group who have common tastes. While implementing a methodology, the similarity between various items is known by using a similarity measure that is most suitable, to find the user ratings to the items that are not previously rated by him/her. Rather than depending on the person who is most similar, the predicted rating value is based on the weighted average of multiple people's recommendations. Weight given to the ratings of a person is known by the comparison between the person and the target user for whom the prediction should be made. The pearson correlation coefficient may be used as a correlation measure. In memory-based approach we build collaborative filtering models by using machine learning algorithms. In this we find the ratings of an item that user might give.

## VI. IMPLEMENTATION RESULTS AND ANALYSIS

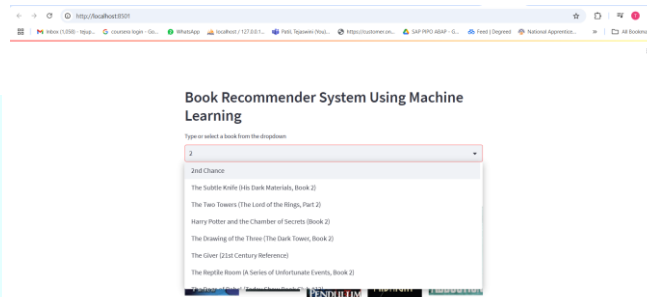
Below is the UI for the Book Recommendation Platform:



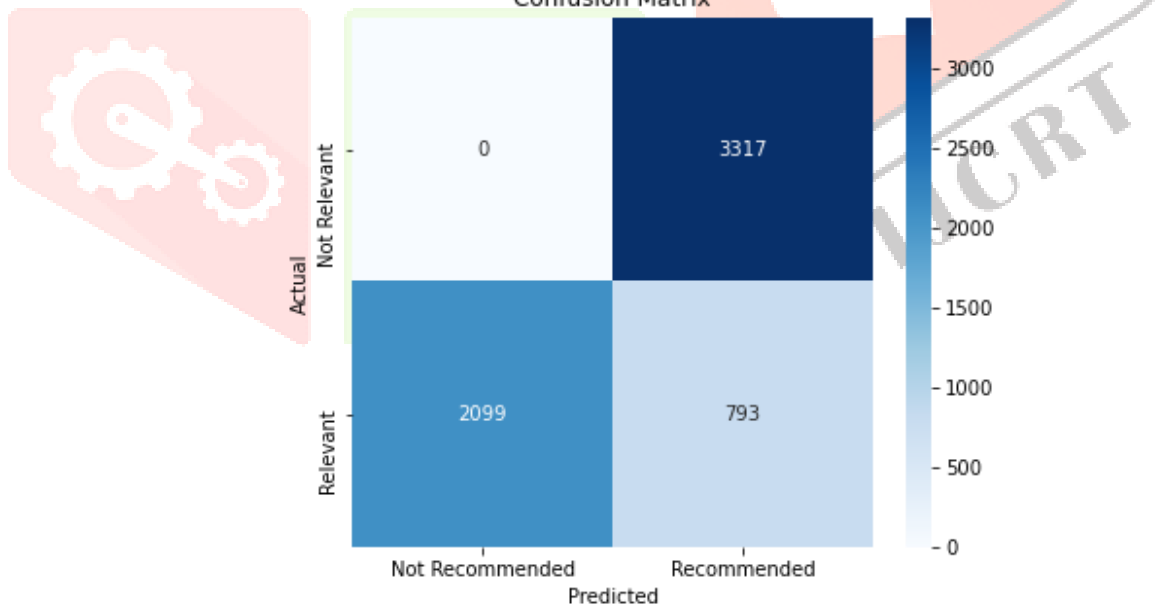
By clicking the "Show Recommendation" button, users can view the top 5 book recommendations.



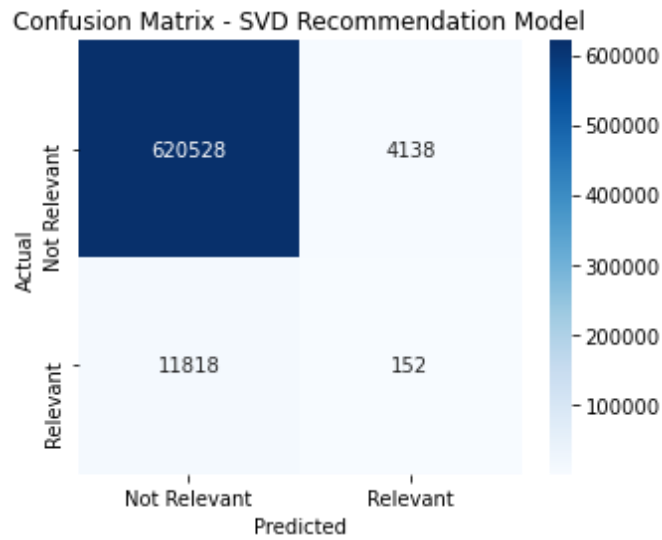
Additionally, users can type a book name in the dropdown menu, and matching titles will appear in the list for easy selection.



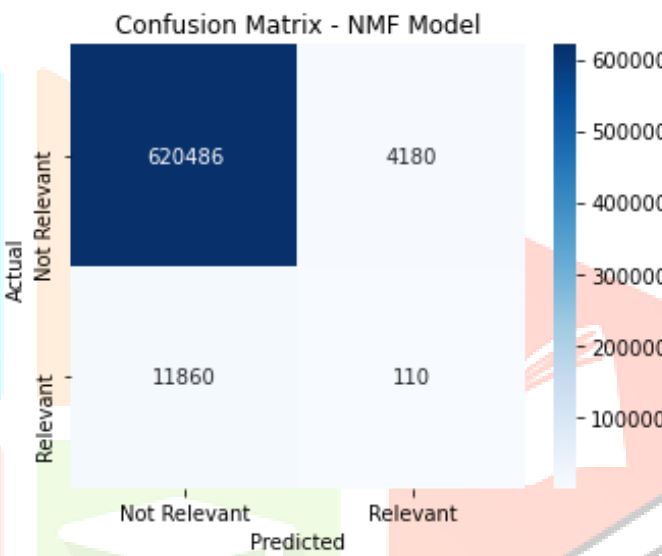
**Confusion Matrix:**



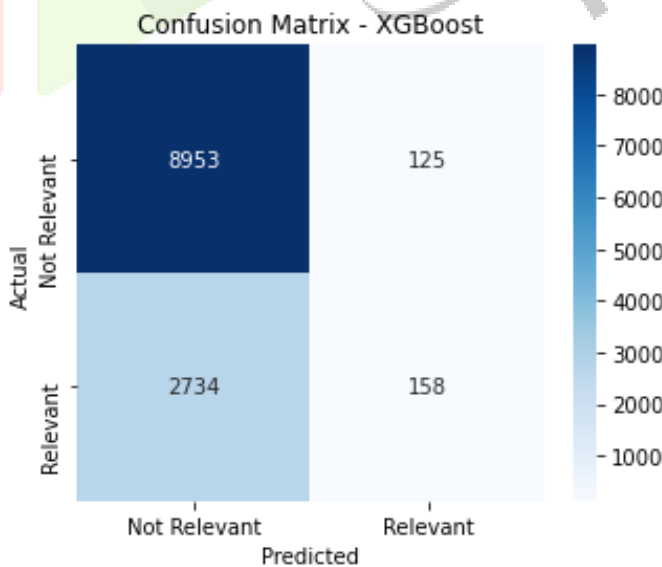
**Fig. Confusion Matrix- KNN**



**Fig. Confusion Matrix- SVD**



**Fig. Confusion Matrix- NMF**



**Fig. Confusion Matrix- XGBoost**

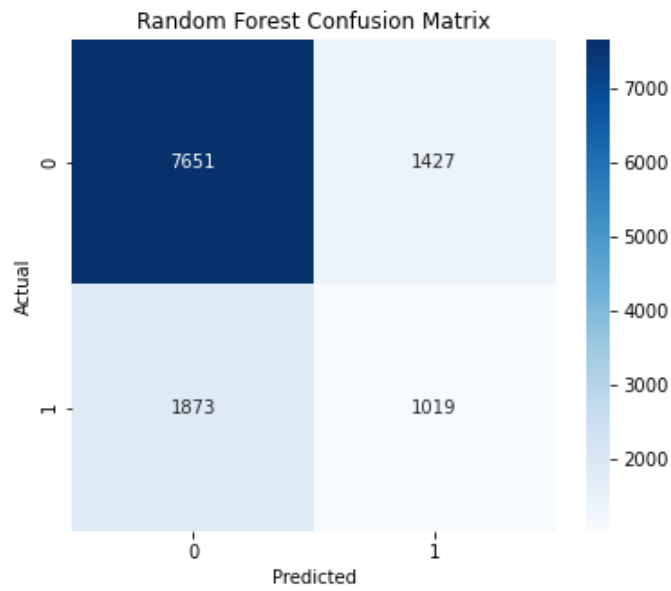


Fig. Confusion Matrix- RF

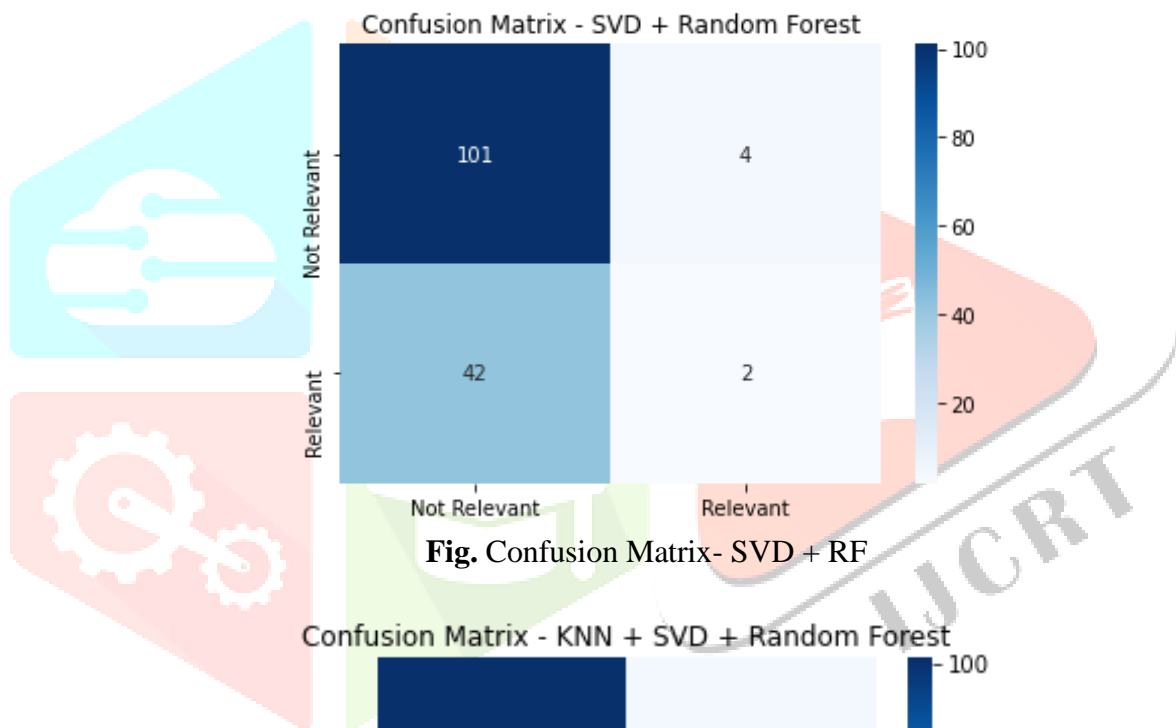


Fig. Confusion Matrix- SVD + RF

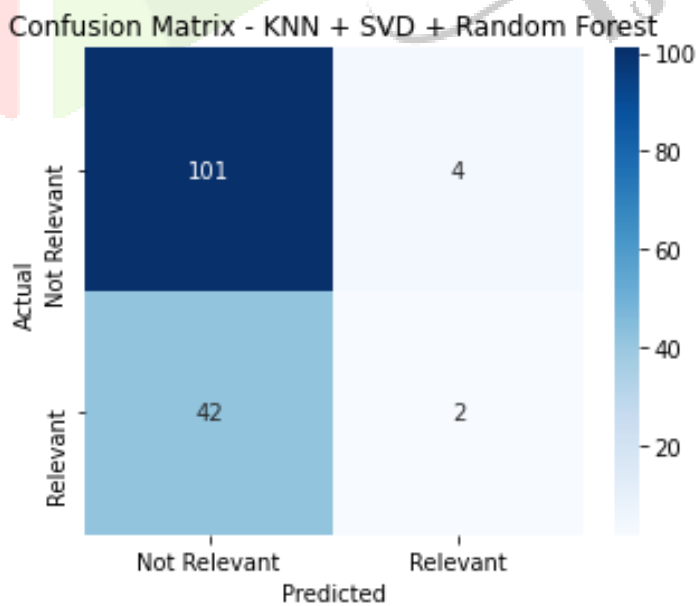


Fig. Confusion Matrix- KNN + SVD + RF

## VII. CONCLUSION

In this study, we proposed improved methods for recommending books by implementing user-based collaborative filtering to provide more accurate and efficient suggestions. Various similarity metrics were analyzed, and the most effective one was identified based on performance. The core of the user-based collaborative filtering approach lies in constructing a similarity matrix that evaluates how closely users relate to one another. This matrix serves as the foundation for generating recommendations by leveraging user behavior patterns. Looking ahead, future enhancements should focus on securing the system against potential threats and exploring alternative recommendation algorithms to further optimize results.

## REFERENCES

- [1] A Sheheeda Manakkadu, Srijan Prasad Joshi, Tom Halverson, Sourav Dutta, (2021) "Top-k User-Based Collaborative Recommendation System Using MapReduce"
- [2] Vignesh Thannimalai, Li Zhang, (2021) "A Content Based and Collaborative Filtering Recommender System"
- [3] Vinod Jain, Kriti Bansal, (2020) "A Novel Approach using user based Similarity for Recommendation Systems"
- [4] Nida Khairunnisa Kusumawardhani, Muhammad Nasrun, Casi Setianingsih, "Web Recommended System Library Book Selection Using Item Based Collaborative Filtering Method".
- [5] Vinayak Hegde, Praveena Mathew, Bincy Kuriakose, "Book Recommendation System through Content Based and Collaborative Filtering Method".
- [6] Sanjay Kumar Jain, Chhavi Rana, "Building a Book Recommender system using time based content filtering".
- [7] Bharti Joshi and Suhasini Parvatikar proposed an online book recommendation approach that integrates collaborative filtering with association rule mining techniques.
- [8] Ritu Rani, Rahul Sahu, "Book Recommendation System by using K-Mean clustering and Collaborative filtering".
- [9] Madhuri Komminen, P.Alekhya, T. Mohana Vyshnav, V.Aparna, K Swetha, V Mounika, "Machine Learning based Efficient Recommendation System for Book Selection using User based Collaborative Filtering Algorithm".
- [10] Nida Khairunnisa Kusumawardhani, Muhammad Nasrun, Casi Setianingsih, "Web Recommended System Library Book Selection Using Item Based Collaborative Filtering Method".
- [11] Vinayak Hegde, Praveena Mathew, Bincy Kuriakose, "Book Recommendation System through Content Based and Collaborative Filtering Method".
- [12] Suhasini Parvatikar and Dr. Bharti Joshi introduced a book recommendation system that combines collaborative filtering with association rule mining to enhance recommendation accuracy.