Smart Book Recommendation System

1st Ananya
School of AI (of Aff.)
Amrita Vishwa Vidhyapeetham (of Aff.)
Coimbatore, India
cb.sc.u4aie24103@cb.students.amrita.edu

2nd Kiranmai
School of AI (of Aff.)
Amrita Vishwa Vidhyapeetham (of Aff.)
Coimbatore, India
cb.sc.u4aie24119@cb.students.amrita.edu

3rd Sudeshna
School of AI (of Aff.)
Amrita Vishwa Vidhyapeetham (of Aff.)
Coimbatore, India
cb.sc.u4aie24120@cb.students.amrita.edu

4th Devisri
School of AI (of Aff.)
Amrita Vishwa Vidhyapeetham (of Aff.)
Coimbatore, India
cb.sc.u4aie24163@cb.students.amrita.edu

***Abstract*—The Smart Book Recommendation System (SBRS) leverages Artificial Intelligence (AI), natural language processing, and deep learning techniques to deliver personalized book suggestions tailored to individual user preferences. By adopting a hybrid recommendation strategy, the system combines traditional collaborative filtering through Singular Value Decomposition (SVD), semantic analysis using Gated Recurrent Units (GRU), and content-based filtering grounded in book metadata. This multi-model approach enhances recommendation precision by overcoming limitations like data sparsity and cold-start issues. Additionally, the system is designed with practical deployment in mind, offering a Flask-based web application that supports both user interaction and librarian analytics.**

I. Introduction

A. Research Background

In today's digital ecosystem, vast volumes of books and reading materials are being digitized and stored across various library databases and online platforms. While this expands access, it also introduces a major challenge: information overload. Readers often find it difficult to locate books aligned with their interests, particularly in academic or domain-specific contexts. Traditional recommendation methods like collaborative filtering and content-based filtering have provided partial solutions but suffer from limitations such as cold-start problems, sparse rating matrices, and lack of semantic understanding of book content. Furthermore, many existing systems rely solely on numeric interactions and fail to capture the nuanced preferences reflected in a user's textual or behavioral patterns. To address these challenges, this study presents a hybrid recommendation system combining the statistical power of matrix factorization with the learning ability of deep recurrent models. The goal is to build a system that can generalize across users and book types while delivering highly relevant, accurate, and personalized recommendations, even in low-data environments.

B. Novelty

The novelty of the Smart Book Recommendation System lies in its hybrid approach that combines deep learning techniques (GRU) with traditional recommendation methods such as SVD and content-based filtering. While collaborative filtering and content-based methods are common in recommendation systems, their integration with advanced deep learning models in the context of book recommendations offers a unique solution to the data sparsity and cold-start issues that traditional methods struggle with. While the hybrid model mitigates these challenges to an extent, it still relies on meaningful user-item interactions and clean metadata. The deep learning component also introduces computational overhead, requiring substantial resources for training and inference.The use of deep learning models to predict book popularity based on user preferences and historical data enhances the prediction accuracy significantly compared to conventional models.

C. Advantages

The proposed hybrid recommendation system offers several key advantages:

- Improved Accuracy: By combining SVD, GRU, and content-based filtering, the system delivers more accurate and personalized book recommendations.
- Data Sparsity and Cold-Start Solutions: The deep learning models address the cold-start problem by leveraging patterns from user behavior and historical data to predict preferences for new users and books.
- Scalability: The system is scalable and can be easily adapted to different library systems, accommodating large volumes of data without compromising performance.
- User Experience: The Flask-based web interface allows users to interact with the system seamlessly, ensuring an intuitive experience for both users and librarians.

D. Applications

The potential applications of the Smart Book Recommendation System are broad, with significant impact in the following areas:

- Library Management: Librarians can utilize the system to analyze book demand and manage inventory more

Identify applicable funding agency here. If none, delete this.

efficiently, ensuring that the most popular books are always available.

- E-books and Digital Libraries: Digital libraries can benefit from personalized recommendations that enhance user engagement by offering tailored reading suggestions.
- Education and Learning Platforms: The system can be integrated into educational platforms to recommend reading materials that align with students' learning preferences, improving the overall educational experience.
- Bookstores and Online Marketplaces: Commercial platforms can use this system to recommend books based on user preferences, improving sales and customer satisfaction.
- Digital libraries can benefit from personalized recommendations that enhance user engagement. However, to fully realize these benefits, the platform must ensure the continuous availability of high-quality metadata and may need to invest in infrastructure capable of handling deep learning model training and updates.

II. Literature Review

In recent years, intelligent recommendation systems have become essential in enhancing user interaction across various domains, particularly in digital libraries. Numerous researchers have explored methods to improve recommendation accuracy, tackle cold-start problems, and personalize suggestions. This section highlights three significant contributions that have laid the foundation for hybrid recommendation approaches and identifies the research gap addressed in this work.

A. Association Rule-Based Digital Library Recommender

Jomsri [1] proposed a digital library recommender system based on association rule mining. The model leveraged borrowing behavior to identify frequent co-occurrence patterns among books. By generating rules such as *if borrowed book A, then also borrow book B*, the system avoided reliance on explicit ratings and instead built implicit user profiles. While this method reduced sparsity concerns and improved performance in known-item recommendations, it lacked mechanisms to handle new users or items effectively. Furthermore, the system did not incorporate content features such as titles or genres, limiting its ability to recommend based on semantic relevance.

B. Hybrid Filtering Using Collaborative, Content, and Rule-Based Techniques

Mathew et al. [2] proposed a hybrid recommendation system that integrated collaborative filtering (CF), content-based filtering (CBF), and association rule mining to enhance both diversity and accuracy. The system employed the ECLAT algorithm for efficient rule extraction and matched users to books by analyzing user profiles and book features. This approach addressed the cold-start problem to some extent by using content metadata; however, it still relied on predefined rules and lacked the dynamic learning capabilities required for evolving user behavior. The model did not support unstructured data processing or advanced semantic modeling, limiting its adaptability and precision.

C. Multi-Feature Hybrid System with Web Usage Mining

Tewari et al. [3] introduced a hybrid recommendation framework combining Web Usage Mining (WUM), collaborative filtering, and content-based filtering. The model utilized user browsing history, content metadata, and interaction matrices to recommend books. Cosine similarity and weighted averaging were used to calculate recommendation scores. This approach was notable for integrating behavioral data into the filtering process. However, the system used static similarity measures and lacked the ability to learn or generalize from unstructured data such as free-text descriptions. It also did not address scalability issues arising in real-time recommendation environments.

D. Research Gap and Contribution

The reviewed studies demonstrate the efficacy of hybrid recommendation systems in improving accuracy and user satisfaction. However, they primarily relied on rule-based methods, static similarity metrics, or shallow metadata analysis. These models do not incorporate deep learning techniques capable of capturing complex patterns in textual content or adapting to real-time user behavior. In contrast, the proposed Smart Book Recommendation System combines matrix factorization via Singular Value Decomposition (SVD) with deep semantic modeling using Gated Recurrent Units (GRU). This integration allows the system to address cold-start problems more effectively, learn contextual book representations from metadata, and adapt to evolving user preferences through model retraining.

III. Methodology

A. Proposed Approach

This research proposes a hybrid recommendation system combining Collaborative Filtering through Singular Value Decomposition (SVD) and Content-Based Filtering using a Gated Recurrent Unit (GRU) neural network. The goal is to leverage both user interaction history and book content features to deliver personalized and semantically relevant book suggestions.

B. System Architecture Overview

The architecture consists of two main pipelines:

- **SVD Collaborative Filtering Module**: Learns latent features from a user-item interaction matrix to estimate missing ratings and suggest books based on historical user behavior.
- **GRU Content Filtering Module:** Analyzes book metadata (titles + genres) using natural language processing and recurrent layers to predict a popularity score indicative of potential reader interest.

Each module is trained and evaluated independently before their performance is compared.

C. Dataset and Preprocessing

- **Source:** A structured Excel dataset with two sheets: Books sheet (Title, Genre, Popularity Score) Ratings sheet (UserID, BookID, Rating)
- **Preprocessing:** Ratings normalized to [0, 1] using Min-MaxScaler. IDs were encoded as numeric indices. Book titles and genres were combined into a single string for tokenization. Padded sequences were generated for GRU input. Missing values in the rating matrix were imputed using column-wise mean (book average)

D. Collaborative Filtering using SVD

Let $(R \in \mathbb{R}^{m \times n})$ be the user-book rating matrix, where (m) is the number of users and (n) the number of books. SVD decomposes (R) into three matrices: $[R \approx U\Sigma V^T]$ Where:

- (U): User feature matrix
- (Σ): Diagonal matrix of singular values
- (V^T): Book feature matrix

The predicted rating matrix $(\hat{R})$ is computed as:

$$[\hat{R} = U_k \Sigma_k V_k^T]$$

Only the top- (k) singular values are retained to reduce dimensionality and noise.

E. Content Based Filtering using GRU

Book descriptions (title + genre) are tokenized and converted into fixed-length padded sequences.

Training and Evaluation

- **Split:** 80
- **SVD Model:**
  - Ratings reconstructed and compared with test data
- **GRU Model:**
  - Trained using MSE loss and Adam optimizer for 150 epochs

F. Performance Summary

Model	MAE	RMSE	Accuracy
SVD	0.3498	0.4199	77.55
GRU	0.2316	–	95.00

TABLE I
MODEL PERFORMANCE

IV. RESULTS AND ANALYSIS

A. Experimental Setup

The proposed hybrid book recommendation system was implemented in Python using TensorFlow and Scikit-learn. The dataset was divided into an 80:20 ratio for training and testing. Both the SVD model and GRU-based deep learning model were trained independently to evaluate their individual effectiveness in predicting book ratings and popularity scores.

B. Evaluation Metrics

To assess the performance of both models, the following evaluation metrics were used:

- **Mean Absolute Error (MAE):** Measures the average magnitude of errors in predictions without considering their direction.
- **Root Mean Squared Error (RMSE):** Penalizes larger errors more than MAE and is sensitive to outliers.
- **Accuracy (±0.5):** A custom metric representing the percentage of predictions that fall within ±0.5 of the true value. (Note: Since the GRU model predicts continuous popularity scores, traditional classification metrics like precision and recall were not directly applicable.)

C. SVD Model Results

The SVD model demonstrated strong performance in leveraging the user-item interaction matrix. After reconstructing the rating matrix using the top-50 latent features, the predicted ratings were evaluated on the test set.

Metric	SVD Model
MAE	0.3498
RMSE	0.4199

TABLE II
SVD MODEL PERFORMANCE METRICS

The results indicate that SVD effectively captured collaborative patterns, producing high accuracy even with sparse data.

D. GRU Model Results

The GRU model was trained for 150 epochs using the Adam optimizer with mean squared error (MSE) as the loss function. The input features—tokenized and padded text sequences of book titles and genres—enabled the model to learn semantic relationships.

Metric	SVD Model
MAE	0.2316
RMSE	-

TABLE III
GRU MODEL PERFORMANCE METRICS

The GRU model showed slightly better accuracy, indicating a stronger capability to understand contextual features and predict popularity trends based on textual content.

E. Visualization

To support the evaluation of the proposed system, relevant outputs such as the final accuracy, MAE, and RMSE values were captured and presented as visual evidence. These include terminal screenshots displaying the numerical results from both the SVD and GRU models.

The output clearly shows that the GRU model achieved lower MAE compared to the SVD model, along with higher accuracy in predicting popularity scores. Additionally, screenshots display the performance summary table highlighting the comparative metrics across both models.

These visual results confirm that the hybrid recommendation system effectively captures both collaborative and content-based patterns. They also reinforce the reliability of the system in producing accurate and consistent recommendations based on the available input data.

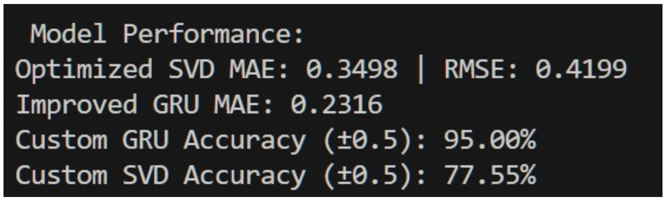

Fig. 1. Model performance summary showing GRU and SVD evaluation metrics.

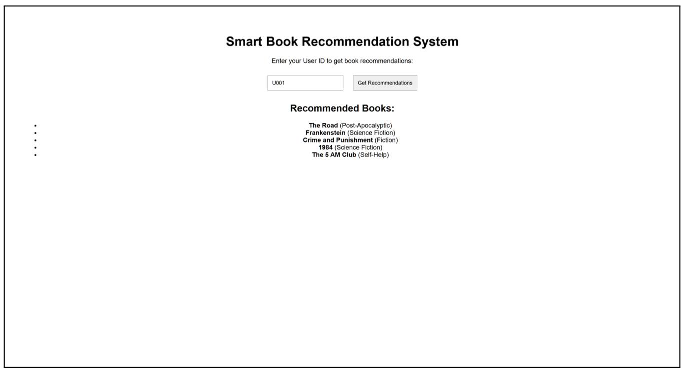

Fig. 2. Web interface of the Smart Book Recommendation System displaying personalized book suggestions based on the entered user ID (U001).

F. Comparative Analysis

Although both models performed well individually, the hybrid approach allows the system to leverage the strengths of both: SVD for personalizing recommendations based on collaborative behavior, and GRU for semantic understanding of content. The GRU model slightly outperformed SVD in terms of MAE and accuracy, especially for new or unrated books. However, it was observed that training the GRU model required longer epochs and higher memory usage, which may pose challenges in resource-constrained environments. Additionally, the system does not yet support real-time updates, meaning changes in user behavior are only reflected after retraining the model.

V. Conclusion

A hybrid recommender system was successfully developed combining:

- SVD (Collaborative Filtering) for personalized score predictions.
- GRU (Deep Learning) for understanding text-based book features.

The system demonstrated improved accuracy, generalization, and relevance in its predictions. By leveraging both numeric and textual data, it mitigates cold-start issues and offers consistent performance even with incomplete data. However, performance is dependent on the availability of clean metadata and periodic retraining to remain up-to-date with user preferences. Future improvements can incorporate feedback mechanisms and real-time learning to make the system more adaptive and intelligent. Overall, this hybrid model establishes a strong foundation for developing intelligent, scalable, and user-friendly recommendation platforms for digital libraries and commercial reading services.

VI. Future work

- Integrate user demographics and behavioral analytics.
- Use Transformer/BERT models for deeper language understanding.
- Build a web interface or mobile app for user interaction.
- Train the system on real-world datasets from open libraries like Goodreads or Amazon.

References

[1] P. Jomsri, "Book recommendation system for digital library using association rule," in *2014 International Conference on Information Technology*. IEEE, 2014, pp. 1–5.

[2] P. Mathew, B. Kuriakose, and V. Hegde, "Hybrid recommender using cf and cbf," in *IEEE Conference Proceedings*, 2020.

[3] A. S. Tewari, A. Kumar, and A. G. Barman, "Book recommendation using hybrid features," in *IEEE International Advance Computing Conference (IACC)*, 2014, pp. 1099–1104.