

**AI For Book Analysis**

**20-12-2024**

Submitted by: *Rohan Singh* Supervised by: *Apoorav Mittal*

*Ankita Saha**Aashish Rana*

*Shreya Chatterjee Jaybrata Chakraborty*

*Sanghamitra Goswami*

*Jaykishan Padia*

**CANDIDATE’S DECLARATION**

We, the undersigned, hereby declare that the information provided in this report is true and accurate to the best of our knowledge and belief. The project entitled, “AI For Book Analysis” has been carried out by us during the six months industry training in Data Science at SABUDH FOUNDATION.

This work is entirely our own, and we have not copied or plagiarized any part of it. We further declare that the project was carried out under the supervision of Apoorav Mittal, Aashish Rana, Jaybrata Chakraborty

and the work described in this report is a genuine reflection of the work we have done.

We also understand that any discrepancy found in the information provided or any violation of academic integrity may lead to disqualification or other disciplinary action.

Date: 20/12/2024

Place: Kolkata, West Bengal, Rajkot, Gujarat

Candidate’s Name: Rohan Singh, Ankita Saha, Shreya Chatterjee, Sanghamitra Goswami, Jaykishan Padia

Signature: Rohan Singh, Ankita Saha, Shreya Chatterjee, Sanghamitra Goswami, Jaykishan Padia

**Acknowledgement**

We would like to express my heartfelt gratitude to the Sabudh Foundation and its esteemed founders, Dr. Sarabjot Singh Anand and Dr. Sukhjit Singh Sehra, for providing us with this incredible opportunity to work and learn. We are deeply thankful to my project mentors for their unwavering guidance and support throughout this journey.

A special thanks to the dedicated Sabudh team, our friends and everyone who has contributed, directly or indirectly, to the success of this project. Your encouragement and assistance have been invaluable, and we are truly grateful for your support.

**Abstract**

The increasing difficulty readers face in discovering books tailored to their preferences is a pressing challenge, exacerbated by the overwhelming volume of options and the limitations of traditional recommendation systems. These systems, often based on content-based or collaborative filtering, fail to adapt to evolving user preferences and overlook detailed book characteristics such as themes and genres.

This project proposes a hybrid AI-driven book recommendation system that integrates content-based and collaborative filtering methods to overcome these shortcomings. The system is designed to deliver highly personalized and dynamic recommendations by leveraging the strengths of each approach. A user-friendly interface enhances accessibility and interaction.

By addressing the limitations of existing systems and introducing a sophisticated hybrid approach, this project aims to transform the book discovery experience, offering readers more meaningful and personalized recommendations that adapt to their evolving preferences.

**Contents**

1. Introduction
2. Methodology
3. Libraries and Packages
4. Dataset Collection
5. Data Preprocessing
6. Project Specific Development
7. Future Prospects
8. Conclusion
9. References

**Introduction**

In an era where the abundance of choices often leads to decision fatigue, discovering books that resonate with individual preferences has become increasingly challenging for readers. Traditional recommendation systems typically rely on either content-based filtering, which emphasizes item features, or collaborative filtering, which focuses on user behavior. While these methods have made strides in providing recommendations, they often fall short in adapting to evolving user preferences and fail to capture nuanced book characteristics, such as themes, genres, and styles.

This project addresses these limitations by developing a hybrid AI-driven book recommendation system. By combining the strengths of content-based filtering and collaborative filtering, the system aims to deliver highly personalized and dynamic recommendations. The integration of these techniques allows for a more comprehensive understanding of user preferences and detailed book attributes.

To ensure an intuitive user experience, the system incorporates a user-friendly interface, designed for seamless interaction. This innovative approach not only enriches the reading journey for users but also sets a new benchmark in recommendation system design by addressing the intricate balance between personalization and discoverability.

**Methodology**

1. The dataset was sourced from a GitHub repository containing Goodreads book datasets. This served as the foundation for building the recommendation system.

2. To ensure data quality and relevance:

* Columns with more than 50% missing values and irrelevant information were removed.
* Retained columns: isbn, title, author, description, link, image\_url, and average\_rating.

3. Data Processing

* Language Standardization:
  + Used langdetect to identify the language of book descriptions.
  + Non-English descriptions were translated into English using googletrans for consistency.
  + Error Handling: Managed LangDetectException errors for cases with insufficient text, ensuring smooth processing.
* Synthetic Dataset Creation:
  + Books were distributed among users using a Gaussian distribution.
  + Updated image links to match Goodreads' new system.
  + Generated user ratings by creating a Gaussian distribution around the average rating for each book.

4. Data Storage and Retrieval

* Data was stored in MongoDB, chosen for its flexibility in handling large, unstructured datasets like book descriptions, user data, and ratings.
* MongoDB’s efficient querying capabilities allowed seamless data retrieval for the recommendation system pipeline.

5. Content-Based Recommendation System

* Initially, Word2Vec was used for embedding generation but later replaced by Sentence Transformers for better contextual understanding and memory efficiency.
* Semantic embeddings of book descriptions were generated using the "all-MiniLM-L6-v2" SBERT model to capture their meaning.
* Embeddings were indexed using FAISS, enabling efficient similarity searches through cosine similarity.
* A recommendation function was developed to retrieve books with semantically similar descriptions based on a given title.
* The system was demonstrated via a Gradio interface, allowing users to input a title and view relevant book recommendations.

6. Collaborative Filtering

* Initially used Word2Vec to create embeddings based on ISBNs of books read.
* Transitioned to Sentence Transformers ("all-MiniLM-L6-v2") for enhanced contextual understanding.
* Combined user-read book information into strings and generated embeddings, storing them in the FAISS vector store.
* Recommendations were made using cosine similarity to identify users with similar reading behaviors and suggest books liked by them.
* This method provided greater personalization and improved contextual relevance in recommendations.

**Libraries and Packages**

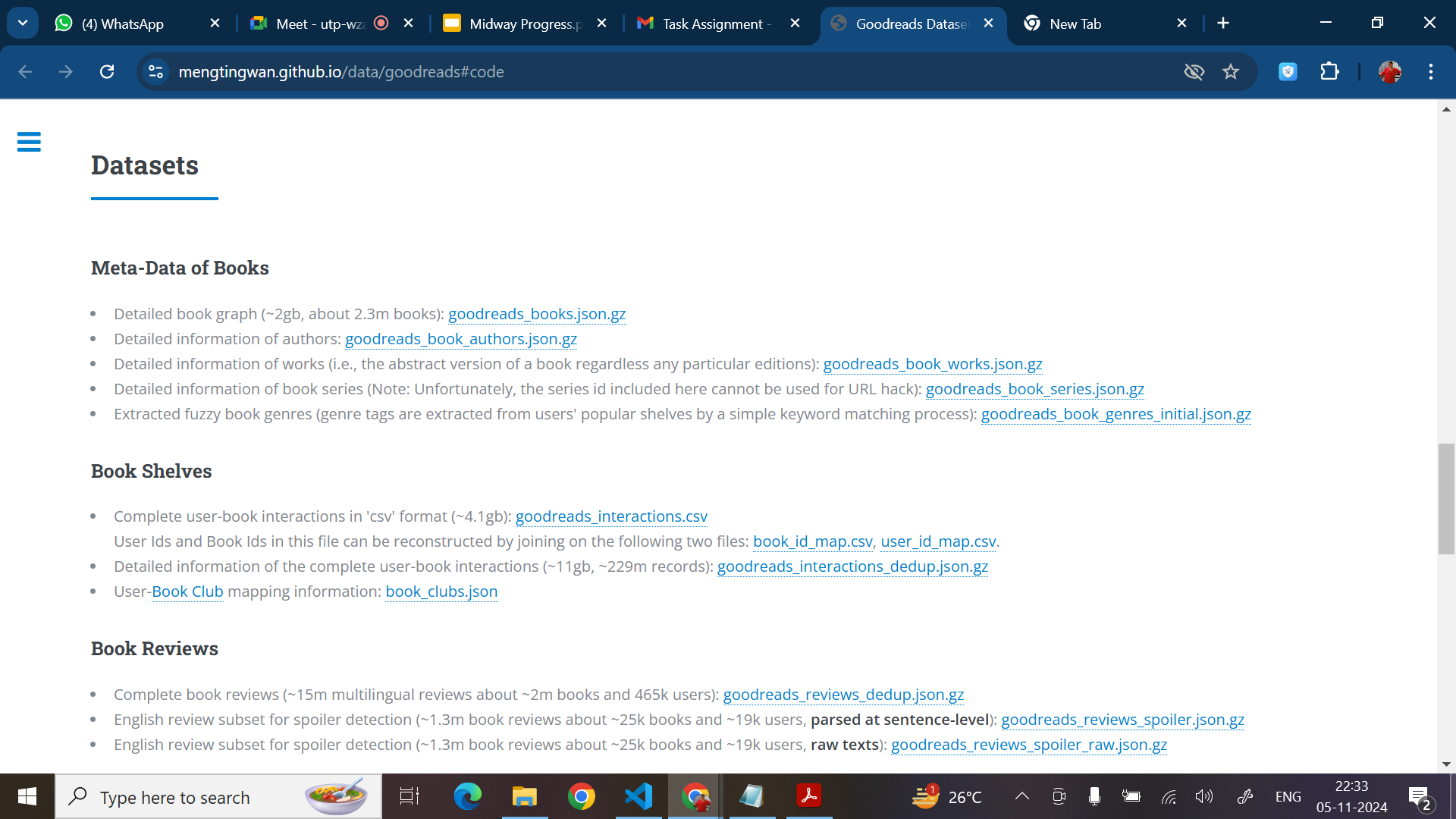
The following libraries and packages were utilized to facilitate the development and implementation of this project:

* **Pandas**: Used for data manipulation and analysis, particularly for handling structured datasets.
* **NumPy**: Facilitated numerical computations and array processing.
* **Seaborn** and **Matplotlib**: Employed for data visualization to create insightful plots and graphs.
* **PyMongo**: Utilized for interacting with MongoDB, a NoSQL database, to manage and query data effectively.
* **Word2Vec**: Implemented for generating vector representations of words to capture semantic meanings.
* **Googletrans**: Used for language translation, enabling the handling of multilingual text data.
* **LangDetect**: Aided in detecting the language of textual inputs.
* **Gradio**: Provided an intuitive user interface for deploying and interacting with machine learning models.
* **FAISS**: Used for efficient similarity search and clustering of dense vectors.
* **Sentence Transformer**: Enabled the generation of high-quality sentence embeddings for semantic similarity tasks.

**Data Collection**

The dataset used for this project was sourced from a publicly available GitHub repository. It consisted of a Goodreads book dataset, which provided comprehensive information about books, including titles, authors, description, ratings, and other metadata.

The dataset was downloaded and preprocessed to suit the requirements of the project.



**Data Preprocessing**

To ensure data quality, relevance, and consistency, the following preprocessing steps were undertaken:

1. Missing and Irrelevant Data: Columns with more than 50% missing values and those deemed irrelevant were removed.

* Columns Retained:
  + isbn
  + title
  + author
  + description
  + link
  + image\_url
  + average\_rating

2. Translation: Ensured all book descriptions are in English to maintain consistency for analysis.

* Language Detection:
  + Used the langdetect library to identify the language of each book description. Descriptions not detected as English were flagged for translation.
* Translation:
  + Used the googletrans library to translate non-English descriptions into English, ensuring all text data was standardized.

Handled LangDetectException errors from the langdetect library, which occur when text is insufficient for language detection. Implemented robust error handling to ensure uninterrupted processing of data.

3. Data Storage and Retrieval: Processed data was stored in MongoDB for efficient storage and easy retrieval during analysis and modeling.

To test the functionality of the recommendation system, a synthetic dataset was created. First, the number of users was initialized, and a subset of available books was selected. These books were distributed among the users following a Gaussian distribution to mimic realistic user-book interactions. Additionally, image links were updated to align with the new system implemented by Goodreads. User ratings were generated by creating a Gaussian distribution centered around the average rating of each book, ensuring a realistic variation in user preferences. This synthetic dataset served as a controlled environment to validate the performance and robustness of the system.

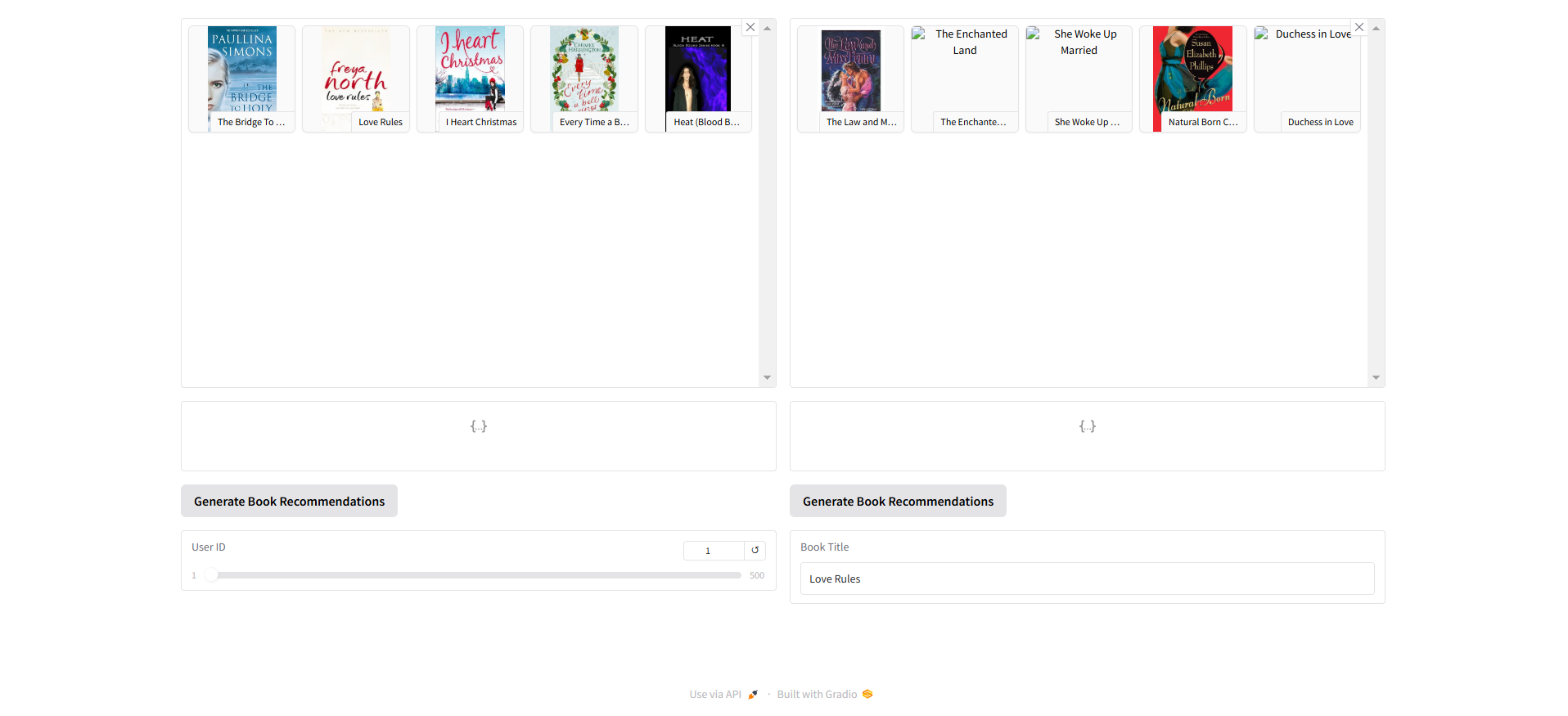
**Project Specific Development**

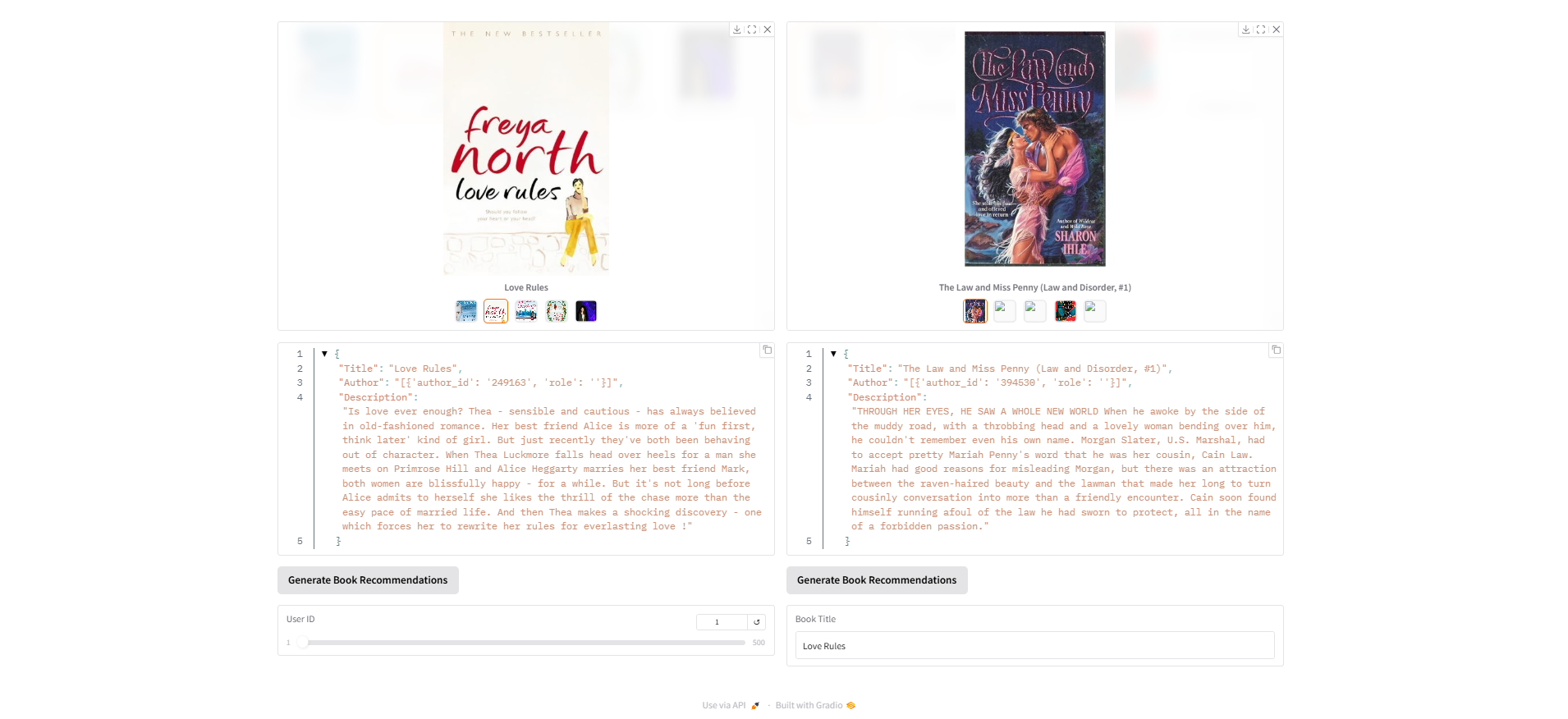
**Content-Based Recommendation System:**

1. Processed a book dataset by removing duplicates based on ISBN and title to ensure data quality and consistency.
2. Used the "all-MiniLM-L6-v2" SBERT model from Sentence Transformers to compute semantic embeddings for book descriptions. The embeddings effectively captured the semantic meaning of the book descriptions, enabling more relevant recommendations.
3. Indexed the generated embeddings using the FAISS library, which facilitated efficient similarity searches. This allowed for quick retrieval of books with similar semantic representations based on vector distances.
4. Developed a function to recommend books by searching for semantically similar descriptions given an input title. The function leverages cosine similarity to rank and retrieve the most relevant books.
5. Integrated the recommendation system with a Gradio interface to demonstrate its functionality. Users can input a book title and receive relevant book recommendations as output.

**Collaborative-Based Recommendation System:**

1. Implemented a collaborative filtering approach that identifies users with similar preferences to the target user, based on their past behaviors. Recommended books that were liked or interacted with by users similar to the target user.
2. Created a unique string for each user, combining all the books they have read along with relevant information. This representation allowed for effective modeling of user preferences.
3. Used the "all-MiniLM-L6-v2" SBERT model to generate embeddings for the user-book data. Stored the embeddings in a FAISS vector store to enable scalable and efficient similarity searches.
4. Utilized Cosine Similarity as the cost function to determine the closeness of user preferences and make personalized recommendations.

****

****

**Future Prospects**

* Enhance the recommendation system by incorporating features such as book genres, authors, user demographics, and user reviews. These additional data points can improve the system's accuracy and relevance by capturing a broader context of user preferences.
* Explore advanced techniques such as deep learning models, hybrid recommendation systems, and attention-based mechanisms to improve the quality of recommendations. Methods like matrix factorization or neural collaborative filtering could also be integrated to enhance performance.
* Develop a more intuitive and user-friendly interface for seamless interaction.
* Deploy the system on AWS or similar cloud platforms to ensure scalability and availability, allowing the system to handle real-world traffic and datasets efficiently.

**Conclusion**

The project successfully developed a book recommendation system capable of delivering personalized suggestions to users. By leveraging different representation techniques, such as Word2Vec and MiniLM, the system effectively captured semantic relationships and user preferences. The combination of collaborative filtering and content-based filtering ensured a comprehensive approach to recommendations, balancing both user behavior patterns and book content characteristics. This dual-methodology approach not only enhanced the accuracy of the recommendations but also provided a scalable foundation for further improvements and real-world deployment.

**References**

* Online Book Recommendation System using Collaborative Filtering (With Jaccard Similarity)

<https://iopscience.iop.org/article/10.1088/1742-6596/1362/1/012130/pdf>

* Book Crossing Dataset

<https://www.kaggle.com/datasets/somnambwl/bookcrossing-dataset>

* Personalized Recommendation Algorithm for books and its implementation

<https://iopscience.iop.org/article/10.1088/1742-6596/1738/1/012053/pdf>

* Personalized Book Recommendation System using Machine Learning Algorithm

[https://www.researchgate.net/publication/348968927](https://www.researchgate.net/publication/348968927_Personalized_Book_Recommendation_System_using_Machine_Learning_Algorithm)

* Online book recommendation using content based and collaborative filtering

<https://ieeexplore.ieee.org/abstract/document/7684166>

* A book recommendation system for using the matrix factorization and content-based filtering approach

<https://www.inderscienceonline.com/doi/abs/10.1504/IJBIDM.2020.104738>