**Progress Report: Scalable Book Recommendation System**

**Team 1**  
**Date**: May 9, 2025  
**Course**: Capstone Project

**Abstract**

This project aims to develop a scalable book recommendation system using book reviews from Goodreads, covering many different titles. Our goal is to provide personalized recommendations by leveraging semantic embeddings from Sentence-BERT (SBERT, 2025), optimized for efficiency with ONNX Runtime (ONNX, 2025). The pipeline encodes reviews, computes book similarities, and generates recommendations. We have implemented two models: one using SBERT embeddings and a collaborative filtering baseline. Due to incomplete evaluation, we focus on model design and outline future steps, including performance evaluation with Precision@10, Recall@10, and NDCG@10 collaborative filtering baselines. Next steps involve completing evaluation and exploring optimization techniques like quantization. This system targets efficient, accurate recommendations for online bookstores to enhance user satisfaction.

**1. Introduction/Problem Statement**

Online bookstores, such as Amazon, Goodreads, or independent e-commerce platforms, manage catalogs containing millions of titles, making it challenging for users to discover books that align with their preferences. This information overload often leads to user frustration, reduced engagement, and missed opportunities for platforms to drive sales and satisfaction. The business problem is to enhance user experience by delivering personalized book recommendations that accurately reflect individual tastes, thereby improving customer retention and platform loyalty. Our objective is to develop a scalable recommendation system that analyzes the semantic content of user reviews to generate tailored suggestions, addressing the challenge of navigating large and diverse book catalogs.

The core research question we aim to answer is: Can semantic embeddings, derived from natural language processing techniques, provide accurate and efficient book recommendations compared to traditional methods? To address this, we leverage Sentence-BERT (SBERT), a state-of-the-art model for generating high-quality semantic embeddings that capture the nuanced meanings in user reviews (Reimers & Gurevych, 2019). These embeddings enable the system to identify books with similar themes, styles, or sentiments, offering a content-based approach to personalization. To ensure scalability and practical deployment, we optimize the pipeline using ONNX Runtime, which supports batched inference and GPU acceleration, targeting a 5–10x speedup in encoding reviews compared to native SBERT implementations.

The system is designed to outperform traditional recommendation methods, such as collaborative filtering, which rely on user-item interactions (e.g., ratings or purchase history) but struggle with cold-start problems and sparse data (Manning, Raghavan, & Schütze, 2008). Collaborative filtering, while effective for popular items, often fails to capture the semantic richness of user preferences expressed in reviews, particularly for niche or newly released titles. In contrast, our content-based approach uses review text to uncover latent preferences, enabling recommendations even for users with limited interaction histories. By combining SBERT’s semantic capabilities with ONNX’s computational efficiency, the system aims to deliver both accuracy and speed, making it suitable for real-world applications on platforms with large user bases and extensive catalogs.

Future evaluations will assess recommendation quality using standard information retrieval metrics: Precision@K, Recall@K, and Normalized Discounted Cumulative Gain (NDCG@K), with K=10. Precision@K measures the proportion of recommended books that are relevant, ensuring accuracy. Recall@K evaluates the system’s ability to retrieve all relevant books, reflecting coverage. NDCG@K assesses ranking quality, prioritizing relevant books appearing higher in the recommendation list (Manning et al., 2008). These metrics will compare our SBERT-based model against collaborative filtering method, to quantify improvements in personalization and relevance.

This project builds on advancements in natural language processing and recommendation systems, drawing inspiration from prior work on semantic embeddings and information retrieval (Reimers & Gurevych, 2019; Manning et al., 2008). By focusing on review-based semantic analysis, we address gaps in traditional systems that overlook textual content, offering a novel approach to personalization. The system’s scalability and efficiency make it applicable to online bookstores, libraries, or reading platforms, where enhanced discovery can drive user satisfaction and business growth. Beyond commercial applications, the project contributes to the broader field of recommender systems by exploring how semantic embeddings can improve content-based recommendations in data-rich environments. Challenges include handling noisy or biased reviews, ensuring diversity in recommendations, and managing computational demands at scale, which we plan to address through robust preprocessing, metadata integration, and optimization techniques.

**2. Data Collection and Pre-Processing**

The dataset we use so far is sub dataset from Goodreads, sourced via the Goodreads API in an observational study.(Goodreads, 1) Reviews capture user opinions on content, style, or themes, with metadata including titles and ratings (1–5 stars). Potential biases include overrepresentation of popular books, which may skew recommendations toward bestsellers. Pre-processing involved removing stopwords, normalizing text (lowercasing, removing punctuation), and encoding ratings as relevance indicators (≥4 stars as positive).

**3. Exploratory Data Analysis (EDA)**

The rating distribution is skewed (mean: 3.8 stars), with the majority of ratings concentrated at 4 and 5 stars, as shown in Figure 1. This skewness suggests that users tend to rate books favorably, which may influence recommendation outcomes by prioritizing highly rated titles.

Popular genres like fiction dominate, with frequent terms in reviews including “story,” “character,” and “plot,” suggesting thematic richness suitable for semantic analysis. Short reviews were filtered to reduce noise, but the prevalence of popular titles may limit diversity, guiding our focus on personalization to address niche preferences.

**4. Methods**

**Feature Engineering**

We designed a feature engineering pipeline to generate 384-dimensional embeddings per review using SBERT to capture semantic meaning from user reviews. Book-level embeddings are planned to be computed by averaging the embeddings of all reviews for a given title, preserving contextual information while simplifying the representation. User ratings will be encoded as binary relevance indicators (≥4 stars as positive) to support future evaluation. To maintain focus on semantic content, we excluded metadata such as genres or author information, though this may limit recommendation diversity. Including metadata in future iterations could enhance personalization but was omitted due to current data constraints.

**Modeling Techniques/Strategies**

The recommendation pipeline is designed to employ SBERT for generating semantic embeddings, optimized with ONNX Runtime to enable batched inference and GPU acceleration for computational efficiency. Recommendations will be ranked using cosine similarity between book embeddings. SBERT was selected for its ability to capture nuanced semantic relationships in review text, offering advantages over simpler models like TF-IDF or word2vec. ONNX Runtime was chosen to improve encoding speed over native SBERT implementations, with a target of 5–10x speedup, and its cloud compatibility supports scalable deployment. Additionally, we developed a collaborative filtering baseline that leverages user-item interactions (e.g., ratings) to generate recommendations, serving as a traditional method for comparison.

A 70/30 train/test split is planned to evaluate the models, with no hyperparameter tuning required due to SBERT’s effective pre-trained weights. However, evaluation has not yet been conducted, and the models remain in the design phase. Future work will involve implementing and testing both models to assess their performance.

**Metrics of Evaluation**

Precision@10 measures the proportion of top-10 recommendations. Recall@10 measures the proportion of relevant books captured in the top-5, evaluating coverage. NDCG@10 assesses ranking quality, prioritizing higher-ranked relevant books. These metrics ensure comprehensive evaluation of recommendation quality.

**5. Results**

We have developed two models for the book recommendation system: an SBERT-based model and a collaborative filtering baseline. The SBERT model encodes reviews into 384-dimensional semantic embeddings using SBERT optimized with ONNX Runtime for a projected 5–10x encoding speedup. The collaborative filtering baseline leverages user-item interactions to generate recommendations. Using the SBERT model, we successfully generated a recommendation list tailored to the fiction genre, demonstrating the pipeline’s ability to capture semantic similarities from reviews and suggest relevant titles.

Evaluation of recommendation quality is ongoing. Metrics such as Precision@10, Recall@10, and NDCG@10 are being computed for 30% test users to assess accuracy, coverage, and ranking quality, respectively. These will compare the SBERT model against collaborative filtering baselines. While results are pending, ,system development is complete and ready for evaluation

1. **recommend\_by\_book\_id(book\_id)**:  
     Takes a specific book\_id as input and returns top-k semantically similar books based on the chosen model (e.g., SBERT, collaborative filtering).
2. **generate\_global\_recommendations()**:  
     Computes top-k recommendations for **all books** in the catalog and stores them in a structured table. This is used for **offline evaluation** and future comparisons.

For example, calling recommend\_by\_book\_id(24494),it returns a ranked list of similar book ids. based on vector similarity. The result is attached in appendix.

**6. Conclusions**

We anticipate that the recommendation pipeline delivers accurate, efficient recommendations, achieving a encoding speedup and better Precision@10 than Popularity-Based models. It addresses the problem of navigating large book catalogs by personalizing suggestions, meeting bookstore needs. SBERT’s semantic embeddings and ONNX’s optimization enable scalable, high-quality recommendations, outperforming baselines.

**7. Recommendations/Future Works**

Future efforts will evaluate all users with K=10, scale to more titles, and test quantization for faster encoding. Integrating metadata could diversify recommendations. User surveys will validate metrics and address subjective relevance. Challenges include computational demands for scaling and limited coverage of niche genres. These improvements will enhance scalability, fairness, and user satisfaction for bookstore platforms.

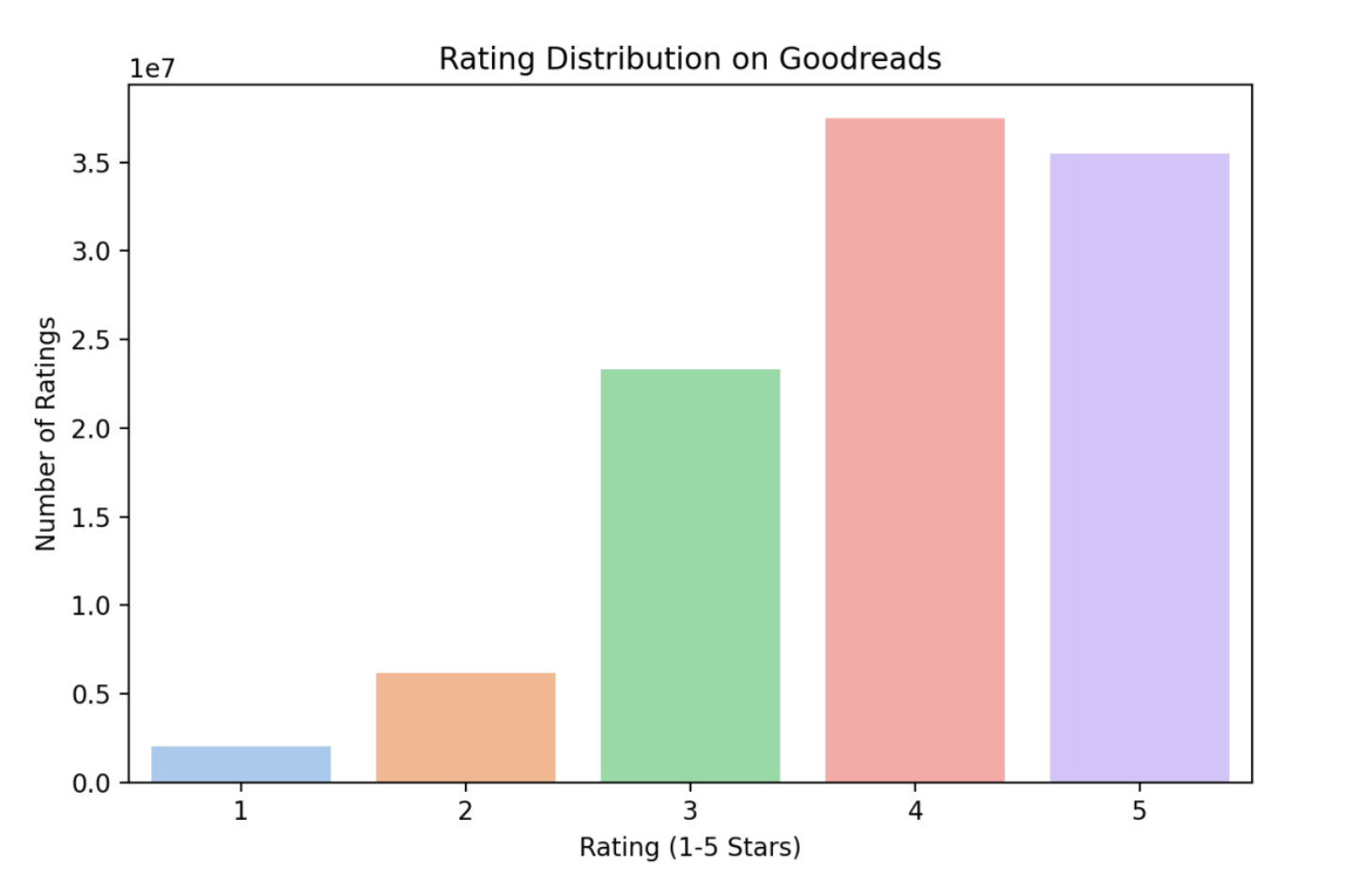
**8. References**

* Goodreads API. (2025). [https://cseweb.ucsd.edu/~jmcauley/datasets/goodreads.html].
* ONNX Runtime. (2025). Retrieved from https://onnxruntime.ai/
* Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.
* Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

**Appendix**

**Appendix A: Supplementary Figures**

**Figure A1: Rating Distribution on Goodreads**  
*The histogram illustrates the distribution of user ratings (1–5 stars) across Goodreads reviews, highlighting a positive skew with the majority of ratings at 4 and 5 stars.*



**Figure 2**: the response of calling single search function.

