REPORT: AI BOOK RECOMMENDATIONS

DATS 6312: Natural Language Processing

Team 1 - Caitlin Bailey, Nina Ebensperger, & Liang Gao

2024-5-1

Contents

List of Figures

List of Tables

1 Introduction

Choosing a good book from the millions available can be impossible. Moreover, bookstores need algorithms that can seamlessly and accurately match readers with the books they are most willing to purchase. In this project, we aimed to address this issue by testing the application of three different types of natural language processing (NLP) models designed to match readers with books: DeepLake, K-Nearest Neighbors (KNN), and Latent Semantic Analysis (LSA) using Singular Value Decomposition (SVD).

In the following sections, this report will describe the data set, NLP models, experimental setup, results, and conclusions of our work.

2 Dataset

This project utilized the Amazon Books Reviews dataset [1]. This dataset offers over 3 GB of data on book reviews and ratings. The data are available in two files linked by the variable Title (i.e., book title). The "books_data.csv" file includes title, description, authors, image, previewLink, publisher, published date, infoLink, categories, and ratings count. The "books_details.csv" file includes variables Id, Title, Price, User_id, profileName, review/helpfulness, review/score, review/time, review/summary, and review/text. More information about the Amazon Books Reviews dataset [1] can be found on Kaggle.com.

Considering the limitations of our GPU capacity, the original size of our dataset was too large, so we only utilized a subset of it. We removed users with fewer than 10 reviews, dropped rows with null values, and dropped categories with a count of less than 5000 and greater than 20,000. The number of raw data observations is 3 million. After filtering, the dataset for later modeling has 58,199 observations.

3 NLP Models

The NLP models we used for this project are as follows: 1) DeepLake, 2) KNN, and 3) LSA via SVD (hereafter referred to as the SVD model. See below for descriptions of each.

DeepLake. DeepLake is not just a transformer model but an advanced data management system optimized for handling high-dimensional vector data, such as embeddings generated by NLP models [2]. It utilizes the SentenceTransformer framework to convert textual data from book descriptions into semantic vector embeddings. These embeddings are managed efficiently within VectorStore, a component of DeepLake designed specifically for storing and querying vector data.

The SentenceTransformer model processes book descriptions to create embeddings that encapsulate the nuanced semantic meanings of the texts. These embeddings are then stored in VectorStore, enabling fast and scalable similarity searches. When a user inputs a query related to a book genre or content, DeepLake retrieves the most relevant book embeddings from VectorStore using cosine similarity. This measure computes the similarity by evaluating the cosine angle between two vectors, ensuring that the recommendations are contextually relevant and semantically aligned with the user's preferences.

SentenceTransformers. SentenceTransformers utilize transformer-based models, often derived from architectures like BERT, to produce sentence embeddings. The process involves the following steps:

Given a text input T, the model processes T through multiple layers of transformer networks, applying self-attention mechanisms to comprehend the context of each word relative to others. The mathematical representation is given by:

$$\textbf{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where Q, K, and V represent queries, keys, and values, respectively, and d_k is the dimensionality of the keys. The final embedding, $\mathbf{E}(T)$, is obtained by pooling the outputs of the last transformer layer:

$$\mathbf{E}(T) = \text{pool}(\text{Layer}_N(T))$$

This method ensures that the embeddings capture both syntactic and semantic nuances of the text, enhancing the efficacy of the recommendation system.

VectorStore Management. DeepLake's architecture supports real-time data processing and querying, critical for interactive applications like online bookstores. The integration of SentenceTransformer and VectorStore within DeepLake provides a robust solution for creating, managing, and utilizing embeddings to enhance user experience through personalized and contextually aware book recommendations.

KNN. The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used for both classification and regression tasks. KNN works by finding the nearest neighbors to a query data point and then basing its prediction on the properties of these neighbors. KNN calculates the distance between points using metrics such as Euclidean, Manhattan, Minkowski, or Hamming distance to determine which known instances are closest to the new one. In classification, KNN assigns a class to the query point based on the majority class among its nearest neighbors. In this task, the KNN algorithm calculates the distance between a user's new review text and the existing reviews in our dataset. It then identifies and outputs the indices of the reviews closest to the input. We can effectively recommend books from our dataset that align with the user's preferences using these indices.

SVD. The SVD model is a classical NLP model typically used for dimensionality reduction and latent semantic analysis. It was initially developed in the late 1960s to decompose matrices and extract meaningful patterns from high-dimensional data [3]. In the context of book reviews and recommendations, the SVD is a clear choice from the classical NLP model toolkit because it allows us to capture the latent semantic structure of the reviews and identify underlying topics or themes. This enables us to generate recommendations based on review similarities and identify related books. The model was implemented using the sklearn package, which efficiently implements various machine-learning algorithms. The SVD model architecture we developed for this task takes in user input as a short (3-5 sentence) book review. Using cosine similarity, the model outputs the top 10 most similar reviews (deduplicated) and relevant linked data from the merged Amazon Books Reviews datasets (i.e., book title, author, book review summary, review rating). Strengths of the SVD model include its ability to capture complex relationships in high-dimensional data, its interpretability, and its flexibility in handling different types of input data. Drawbacks of the model include the lack of CPU optimization for sklearn, which can lead to longer training times for large datasets. However, this limitation can often be mitigated by leveraging parallel processing or using optimized libraries for specific tasks. The SVD equation can be written as:

$$A = U\Sigma V^T \tag{1}$$

Where A is the original matrix (e.g., document-term matrix), U is the matrix of left singular vectors (e.g., a representation of the "concepts" or latent factors, capturing relationships between documents [i.e., rows]), Σ is the diagonal matrix of singular values (e.g., the representation of the significance of each latent factor), and V^T is the transpose of the right singular vectors matrix (e.g., a representation of the latent factors, capturing relationships between terms [i.e., columns]). SVD decomposes the original matrix A into orthogonal basis vectors (in U and V^T) and scaling factors (in Σ), collectively representing the data's latent structure. This decomposition enables dimensionality reduction and semantic analysis.

4 Experimental Setup

The project is designed to compare the functionalities of three models (DeepLake, KNN, and SVD) to the application of generating book recommendations for users based on user input of book reviews (i.e., natural language input). To do this, each model was individually developed, tested, and fine-tuned as needed to produce optimal model performance/accuracy. The development and testing process for each model is described below. Finally, once each model was completed, we compared their respective functionalities, metrics, and task-related outputs to identify each model's strengths and weaknesses for the task of book recommendation generation.

DeepLake. The experimental setup for DeepLake was focused on establishing a robust and responsive environment capable of handling natural language queries and generating real-time book recommendations. The primary components of this setup included the SentenceTransformer model for generating text embeddings and the VectorStore system for managing these embeddings efficiently.

First, the SentenceTransformer model 'all-mpnet-base-v2' was integrated to process and transform book descriptions into high-dimensional vector embeddings. This model was chosen for its ability to capture deep semantic meanings from the text, crucial for the nuanced understanding required in matching books based on thematic and contextual similarities.

Once the embeddings were generated, they were stored in VectorStore, a component of DeepLake optimized for the quick retrieval of vector data. VectorStore was configured to support efficient similarity searches, enabling the system to quickly find and suggest books that are semantically related to the user queries. The setup also included mechanisms for updating and maintaining the vector database as new books were added or existing entries were modified, ensuring the system remained current and accurate.

To ensure the system's effectiveness, a series of predefined book-related queries were used during development to iteratively test and refine the interaction between the SentenceTransformer embeddings and the VectorStore retrieval processes. These queries simulated typical user interactions, helping to fine-tune the system's responsiveness and the relevance of its recommendations before live deployment.

This setup not only facilitated the development of a highly functional recommendation system but also set the stage for a scalable solution that could adapt to increasing data volumes and evolving user needs.

SVD As described above, the SVD model architecture was designed to process natural language text as a short (recommended: 3-5 sentence) book review and output the top most similar book reviews (combined title and summary review data) using cosine similarity. The NLTK package was used to preprocess the text, including tokenization, removal of special characters and stopwords, and lemmatization. The sklearn package performed the TF-IDF vectorization, SVD, and LSA pipeline. The number of components for the model was selected so that 90% of the variance was explained, striking a balance between maximizing information retention and preventing overtraining (Figure 1) [4]. The final trained model was saved as a pickle file for later access.

Given the unsupervised nature of the SVD model, conventional assessment methodologies like train-test splitting, accuracy, and F1 scores were not applicable. Instead, the model's efficacy was evaluated via two alternative methods. First, scree plots of cumulative explained variance and explained variance ratio were employed to gauge the extent to which the model captured and retained essential information from the dataset. Second, user assessment of task-specific model output provided qualitative insights into the model's performance, ensuring alignment with the intended application. This process consisted of inputting eight book reviews of different genres (i.e., mystery, romance, science fiction, fantasy, thriller, historical fiction, young adult, and nonfiction) and manually assessing how many of the recommended books were similar to the review prompt in terms of genre/topic. Book reviews for model assessment were generated by ChatGPT (Table 1) [5].

KNN There are two steps in this part of the experiment. First, we used a pre-trained BertForSequenceClassification model to predict the category that the user potentially likes based on the input

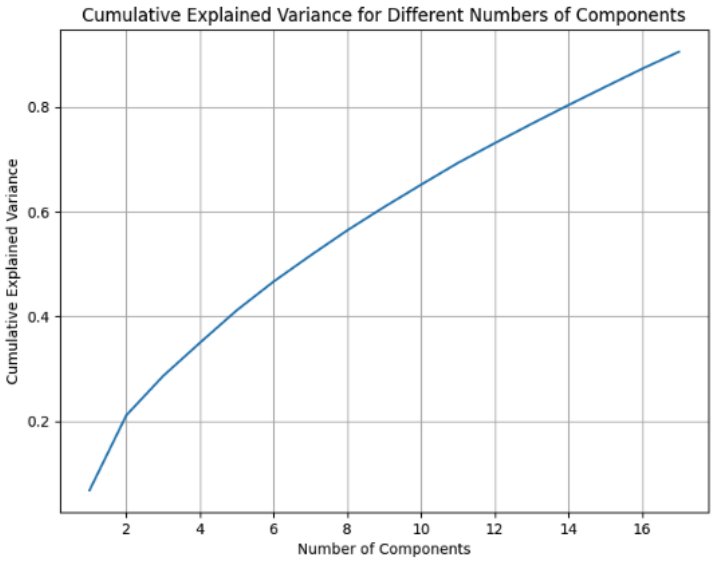

Figure 1: Scree plot of the cumulative explained variance per number of components. Seventeen components explained 90% of the cumulative variance for the SVD model.

review. Second, we filtered the data frame to include only those rows where the 'categories' column matches the predicted category in the first step. Then, KNN will help to find the nearest books using the same input review. The details will be explained as follows.

One of the critical architectures in this part is BERT (Bidirectional Encoder Representations from Transformers), a groundbreaking model in the field of natural language processing (NLP) developed by researchers at Google AI. Introduced in their 2018 paper, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," BERT has significantly advanced the performance of various NLP tasks. BERT was designed to address the limitations of previous models that processed words in a sentence sequentially, either from left to right or right to left. BERT, by contrast, reads the entire sequence of words at once, which allows it to learn the context of a word based on all of its surroundings (left and right of the word).

For tokenization, a pre-trained BertTokenizer([6]) was used to tokenize the 'review_text', replacing classical tokenization methods such as NLTK and Spacy tokenization. The BertTokenizer is a crucial component of the BERT architecture, which has been instrumental in advancing stateoftheart NLP. The BertTokenizer is designed to effectively preprocess text data before it is fed into a BERT model. We can directly get the vector representation of each text review with BertTokenizer, which is content-based, instead of TFIDF, which is frequency-based.

The dataset was divided into three subsets: 80% for training, 10% for validation, and 10% for testing. We have six book categories: Religion(18,894), Business & Economics(10,813), Young Adult Fiction(10,582), Social Science(6,624), Philosophy(6,131), and Science(5,155).

A pretrained BertForSequenceClassification([6]) model was applied to solve this book category classification problem. We encoded the book categories with LabelEncoder and set the format of tokenized reviews and labels for Pytorch. We only need to fine-tune (change hyperparameters) our model since it is pretrained. A validation set will test if the model is overfitted while training.

After getting the predicted category, we can filter the dataset and apply KNN to recommend books. There are many duplicate books in our dataset. Many books have more than one review. Thus, we added some post-processing to our predicted books to remove duplicates. We dropped rows with the same review summary, utilized the 're' package to delete brackets and inside content in the title, dropped duplicate books with the same title, and finally ranked the books based on review score and output the

top 6 as our recommendation.

The assessment is similar to what we did in the SVD model. The training dataset only contains 6 categories. Thus, we used ChatGPT to generate reviews for books from the 6 categories (Table 2). Then, we manually assessed the similarities of recommended books to the review prompt. [5].

5 Results

The results for the development and testing of each model are described below.

DeepLake. The experimental evaluation of DeepLake was primarily qualitative, focusing on its ability to process and respond to natural language queries about books in real-time. Unlike traditional models where performance might be measured through metrics like precision or recall, DeepLake's effectiveness was demonstrated through live demonstrations. The testing involved using a series of book-related queries, such as "I want to read a book about quantum mechanics" or "a book similar to 'Pride and Prejudice'", to assess how well the system could understand and match the semantic content of the queries with appropriate book recommendations.

This practical approach allowed us to observe the system's response times and the relevance of its book suggestions, which are crucial for user satisfaction in real-world applications. The fast response times and accurate matching demonstrated during these live queries highlighted DeepLake's capability to leverage SentenceTransformer-generated embeddings and efficient vector management through VectorStore. This setup ensures that the system not only provides semantically relevant recommendations but also delivers them swiftly, enhancing the overall user experience in interactive environments such as digital libraries and online bookstores.

SVD For the SVD model, 17 components were found to explain 90% of the variance (see Figure 1). We selected 90% cumulative variance as our threshold, a priori, to ensure that a significant portion of the original variance in the dataset is captured and preserved in the reduced-dimensional space. This can help maintain the richness and complexity of the data while still achieving dimensionality reduction benefits.

We also examined a scree plot of explained variance ratio (see Figure 1). This plot indicates that the explained variance ratio peaks at component 2. However, the magnitude of the explained variance ratio is small across components. While the explained variance ratio plot may show diminishing returns after two components, it's essential to consider the model's overall performance in achieving its intended task. Retaining more components might lead to better performance in tasks such as similarity calculation and recommendation generation, where capturing subtle nuances in the data is crucial. To test this hypothesis, in our task-specific assessment, we selected to test an SVD model with 17 components (90% of cumulative variance explained) and an SVD model with 2 components (peak of explained variance ratio), as well as the saturated SVD model with 20 components.

Finally, based on our human user assessment of task-specific output, we found that the SVD model with 17 components was significantly better at correctly recommending books in the same genre or topic area as the review prompt (see Table 3). Compared to the SVD model with 17 components, the SVD model with 20 components was slightly better for prediction in some genres/topics and slightly worse in others. Thus, for our final demonstration, the SVD model with 17 components was selected for implementation.

KNN For the BertClassification model, the metric for the test dataset is F1-weighted at 0.846, F1-macro at 0.834, F1-mico at 0.843, and Cohen-kappa-score at 0.804.

The artificial assessment results of the KNN model are in table 4. Results are subjective and represent personal views only.

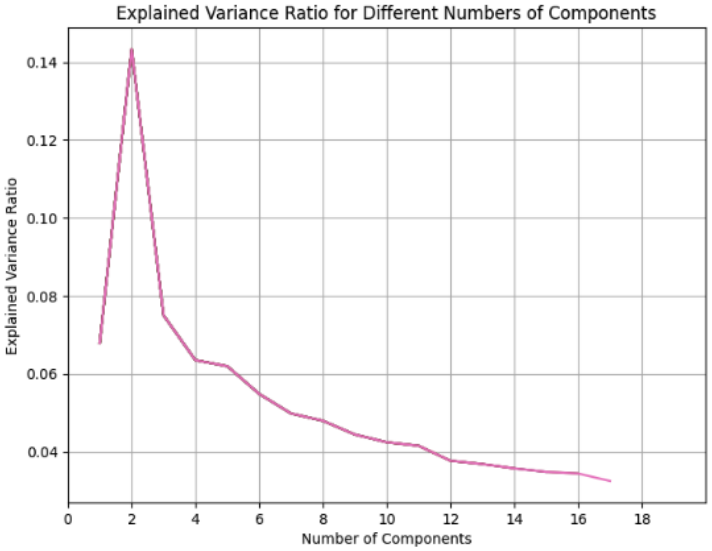

Figure 2: Scree plot of the explained variance ratio per component. Explained variance ratio peaks at component 2 for the SVD model.

6 Conclusions

After conducting our quantitative and qualitative assessments, we concluded that the DeepLake model was best suited for the book recommendation task because of its relative speed (compared to KNN model) and relevant book review output (using manual inspection).

All models had strengths and limitations. For example, the SVD model was relatively straightforward to develop and implement (being a classical NLP model) and did not require GPU resources. This could benefit a small company or an individual looking to implement a model without large resources. However, it is also a limitation of the model in that it ran on CPU and therefore took some time to initially reduce dimensionality and train with the 3GB Amazon Books Reviews dataset. Other limitations of the model include that it required rule-based text preprocessing tasks and manual qualitative assessment of output, given that it was a classical model using unsupervised learning techniques. For KNN model, instead of using this algorithm directly, identifying the book category first can narrow the book's range, which I personally think can increase the KNN performance. One critical limitation of the KNN model is it runs BertTokenizer every time, which requires high CPU computational power and is time-consuming. Additionally, our work lacks quantitative metrics for testing the accuracy of unsupervised machine learning models (e.g., SVD, LSA, KNN) . Our qualitative assessments were based on book genre/topic, but this may underestimate model performance as there may be more nuanced themes within the data that the models can identify.

For future implementations of this task, we recommend utilizing techniques such as recommendation systems and negative sampling to allow the models to make recommendations from negative reviews as inputs and positive reviews.

Overall, comparing SVD, KNN, and DeepLake models was an interesting thought experiment. Comparing the application of more resource-intensive models, such as transformers, to classical models, such as SVD, is an important experiment to ensure that a specific application does, in fact, benefit from utilizing more resource-intensive models. Sometimes, simple may be effective enough, particularly for small, focused tasks and/or small companies. We encourage others to build on our work in future applications of book (or other product) recommendations. In particular, we recommend others work on techniques to allow negative book reviews as model inputs and/or novel methods to assess the functionality of these models, given that they were built using unsupervised learning architecture.

References

[1] M. Bekheet, "Amazon books reviews. kaggle." 2022. [Online]. Available: https://www.kaggle.com/datasets/mohamedbakhet/amazon-books-reviews

[2] Deep Lake Docs, "DeepLake version 3.9.0," 2024. [Online]. Available: https://docs.activeloop.ai/?utm_source=github&utm_medium=github&utm_campaign=github_readme&utm_id=readme

[3] G. W. Stewart, "On the early history of the singular value decomposition," *SIAM Review*, vol. 35, no. 4, pp. 551–566, 1993.

[4] P. Bushel, "Principal variance component analysis," National Institute of Environmental Health Sciences, 2021, retrieved from. [Online]. Available: https://www.niehs.nih.gov/research/resources/software/biostatistics/pvca

[5] OpenAI, "Chatgpt, version 3.5," OpenAI, 2023, retrieved from. [Online]. Available: https://chat.openai.com

[6] H. Face, "Transformers: State-of-the-art natural language processing for pytorch and tensorflow," 2024, retrieved from. [Online]. Available: https://github.com/huggingface/transformers

Genre	Review
Mystery	With clever twists and turns, this mystery novel keeps readers on the edge of their seats as they follow the detective's relentless pursuit of the truth. Every clue unravels another layer of intrigue, leading to a shocking revelation that will leave you guessing until the end. A gripping pageturner that masterfully blends suspense and deduction, making it a mustread for fans of the genre.
Romance	In this heartwarming tale of love and second chances sparks fly when two unlikely souls cross paths. With its tender moments and passionate encounters, this romance novel sweeps readers off their feet into a whirlwind of emotions. The characters' chemistry is palpable, drawing you into their journey of self-discovery and redemption. It is a captivating story of love's transformative power that will leave you longing for more.
Science Fiction	Set in a distant future where technology reigns supreme, this science fiction epic explores humanity's quest for survival in the face of existential threats. With its futuristic landscapes and visionary ideas, the novel immerses readers in a world of awe-inspiring possibilities. From space exploration to artificial intelligence, it delves into the ethical dilemmas and moral quandaries of a technologically advanced society. It is a thought-provoking adventure that pushes the boundaries of imagination and intellect.
Fantasy	Journey to a realm of magic and wonder in this enchanting fantasy novel filled with mythical creatures and epic quests. With its richly imagined world and vibrant characters, the story transports readers to a place where anything is possible. From ancient prophecies to epic battles between good and evil, the narrative weaves a tapestry of adventure and intrigue. A spellbinding tale that captivates the imagination and leaves a lasting impression.
Thriller	Prepare to be on the edge of your seat with this pulse-pounding thriller that delivers non-stop action and suspense. From the opening scene to the heart-stopping climax, the tension builds with each twist and turn of the plot. With its complex characters and high stakes, the novel keeps you guessing until the very end. A gripping rollercoaster ride of thrills and chills that will leave you breathless.
Historical Fiction	Step back in time to an era of intrigue and upheaval in this meticulously researched historical fiction novel. Through vivid storytelling and evocative prose, the author brings the past to life, immersing readers in the sights, sounds, and struggles of bygone eras. From sweeping sagas of war and conquest to intimate portraits of ordinary lives, the novel paints a vivid tapestry of history. A captivating glimpse into the past that resonates with timeless themes of love, loss, and resilience.
Young Adult	Navigating the tumultuous waters of adolescence has never been more captivating than in this poignant young adult novel that explores the trials and triumphs of growing up. From first love to friendship struggles, the story delves into the complex emotions and experiences of teenage life. With its relatable characters and authentic voice, it resonates with readers of all ages, capturing the essence of youth with honesty and empathy. A comingofage tale that speaks to the heart and soul of every teenager.
Nonfiction	From riveting biographies to compelling exposés, this collection of nonfiction essays offers a fascinating glimpse into the diverse tapestry of human experience. With its thought-provoking insights and meticulously researched facts, each essay sheds light on a different world aspect. From science and history to politics and culture, the authors explore a wide range of topics with depth and clarity. A compelling anthology that challenges assumptions, sparks conversation, and broadens horizons.

Table 1: Book reviews generated by ChatGPT for eight different genres.

Category	Review
Religion	The Power of Myth" is a fascinating exploration of the universal themes woven into religious and cultural narratives. Joseph Campbell's insightful discussions with Bill Moyers offer a thought-provoking journey into the depths of human consciousness and the collective imagination. This book provides a profound understanding of the underlying structures of myths and their relevance to our modern lives, making it an enlightening read for anyone seeking wisdom and insight into the human experience.
Business & Economics	The author explores the two systems of thinking that govern our decision-making processes, shedding light on the biases and heuristics that often lead us astray in economic and business contexts. Readers praise the book for its engaging style and eye-opening insights, which challenge conventional wisdom and provide valuable lessons for navigating the complexities of the modern world.
Young Adult Fiction	This heart-wrenching tale follows the lives of two teenagers as they navigate love, loss, and the complexities of living with cancer. The book's authentic portrayal of the character's emotions, coupled with Green's poignant writing style, creates a deeply moving narrative that resonates with readers long after they've turned the final page.
Social Science	This insightful work delves into the complexities of human behavior, drawing upon psychology, sociology, and neuroscience to explore what drives our actions and relationships. Through engaging narratives and compelling research, he illuminates the subtle influences that shape our lives, offering a deeper understanding of the social forces at play in our everyday interactions.
Philosophy	The narrative seamlessly weaves together philosophical concepts with an engaging storyline, making complex ideas accessible and thought-provoking. It's a must-read for both beginners and seasoned philosophers alike, offering a delightful journey through the wonders of human thought.
Science	Sapiens is an eye-opening exploration of humanity's journey from ancient hunter-gatherer societies to the technologically advanced civilization we inhabit today. Harari's ability to distill complex ideas into accessible narratives makes this book informative and engaging. It's a must-read for anyone curious about the origins of our species and the forces that have shaped our societies

Table 2: Book reviews generated by ChatGPT for 6 categories.

	SVD(2 components)	SVD(17)	SVD(20)
Mystery	0/10	8/10	8/10
Romance	1/10	9/10	7/10
Science Fiction	0/10	7/10	4/10
Fantasy	0/10	7/10	9/10
Thriller	1/10	8/10	6/10
Historical Fiction	1/10	7/10	8/10
Young Adult	2/10	6/10	6/10
Nonfiction	10/10	10/10	10/10

Table 3: Proportion of book recommendations similar to the review prompt regarding genre/topic.

Religion	3/6
Business & Economics	5/6
Young Adult Fiction	5/6
Social Science	4/6
Philosophy	6/6
Science	1/6

Table 4: Proportion of book recommendations similar to the review prompt regarding categories.