Assignment 1: Part B
(Big Data Analysis)

Book Recommendation System

Refined Research Questions

Primary Question: How can advanced machine learning techniques be leveraged to build accurate and robust book recommendation systems on sparse user - book interaction data?

This question builds on our initial inquiry by focusing on optimizing modern recommendation models for the characteristics of the Goodbooks-10k (Zajac, 2017) dataset (e.g. high sparsity, implicit feedback).

Sub-Questions (Refined):

1. **Model Efficacy on Sparse Data:** Which recommendation algorithms (e.g. neural CF, graph-based, sequential models) perform best on extremely sparse user - item interactions in the book domain, and how can their architectures be optimized? (This narrows the original question about "which algorithms work best" to specifically consider deep models under sparse data conditions.)
2. **Neural Architecture Comparison:** How do self-attention based sequential models compare to graph-based collaborative filtering models in accuracy and efficiency for book recommendations?
3. **Enhancement Techniques:** Which strategies (e.g. multi-head attention, transformers, graph propagation) offer the most significant improvements for the above models on implicit feedback data? For example, how does adding attention mechanisms or deeper layers affect performance given the data characteristics?
4. **Practicality and Cold-Start:** Given the data limitations, what techniques can mitigate issues like cold-start for new users/items and what is the best trade-off between model complexity and training time for a production-scale book recommender?

These refined questions have evolved from Assignment 1A's exploration. The adjustments were informed by our initial data analysis (e.g. recognizing extreme sparsity and popularity bias), leading us to emphasize models that handle sparse implicit feedback and to compare sequential vs. graph-based approaches in this context.

Dataset Description: Goodbooks-10k (Zajac, 2017)

The Goodbooks-10k dataset is a comprehensive public dataset for book ratings and metadata. It contains six million ratings (1-5 stars) from ~ 53,424 users on 10,000 books, making it one of the largest open book recommendation datasets. Each book is among the 10k most popular on Goodreads (by rating count), and each user is represented by an ID (contiguous 1-53424) with their rating history. Key components of the dataset include: (Zajac, 2017)

- **Ratings data:** Explicit ratings on a 1-5 scale. The file ratings.csv has ~6 million rows of the form (user_id, book_id, rating). Book IDs are 1-10000 and map to Goodreads works (aggregating editions). Sparsity: The user-book interaction matrix is over 99% empty - out of ~500 million possible interactions (50k*10k), only 6M are observed. This extreme sparsity poses challenges (e.g. cold-start for new users/items). The median user has rated only 8 books, indicating a long-tail of very inactive users alongside a smaller number of very active users.
- **Books metadata:** The file books.csv provides metadata for each of the 10,000 books. Available fields include book titles, authors, original publication year, average rating, and counts of ratings and text reviews. For example, The Hunger Games is listed with an original publication year 2008 and has an average rating around 4.34 with millions of ratings counted. We also have breakdowns of how many 1-star, 2-star, … 5-star ratings each book received, as well as Goodreads IDs for books and works. This rich metadata allows analysis of popularity ("ratings_count"), quality ("average_rating"), and other attributes like author and year.
- **Tags/Genres:** A book_tags.csv file lists user-provided tags (shelves or genres) for each book, with a tag ID and count. A companion tags.csv maps tag IDs to tag names. This essentially provides the genres or thematic categories for books. For instance, a fantasy novel might have tags like "fantasy", "young-adult", "magic", each with counts indicating how many users shelved the book in that category. On average, each book has around 4-5 genre tags in the data (after filtering to main genres). These tags enable content-based grouping - we can treat each book as a vector in a "genre/tag space" for clustering.
- **To-Read indicators:** The to_read.csv contains nearly one million entries of users marking books as "to-read" (implicit feedback). While our analysis in this report focuses on the explicit ratings, the to-read data could be leveraged to enrich the model (as additional implicit interactions).
- **Temporal information:** The ratings are timestamp-sorted; though exact timestamps are not included in the public CSV. This suggests that the data preserves the order of interactions, which could allow sequential modeling (e.g. for SASRec). However, because we lack explicit timestamp columns in the CSV, our EDA will treat the data mostly as static aggregated data, except where sequence ordering might be inferred. (Kang and McAuley, 2018)

In summary, Goodbooks-10k (Zajac, 2017) provides a large-scale, multi-faceted snapshot of Goodreads: millions of user-book ratings (with inherent implicit feedback signals and strong positive bias), plus rich item information (genres, authors, etc.). It is an excellent basis for both collaborative filtering and content-based analysis in a book recommendation project. Table below summarizes key statistics of the dataset:

Metric	Value
Users	53,424 (contiguous IDs 1-53424)
Books	10,000 (IDs 1-10000, top Goodreads works)
Ratings	5,976,479 total (explicit 1-5 stars)
Sparsity	~98.8% (only ~1.2% of user-book pairs have a rating)
Median ratings per user	8 (very skewed distribution; many users gave few ratings)
Mean ratings per user	~112 (a few prolific users inflate the average)
Mean ratings per book	~598 (highly uneven; top book ~228,000 ratings in our data vs. hundreds for least popular)
Rating value distribution	5★: ~28%; 4★: ~37%; 3★: ~22%; 2★: ~8%; 1★: ~5% (positive skew; avg ~3.9/5)
Genres/tags	Thousands of distinct tags; common genres (Fantasy, Romance, Sci-Fi, Mystery, Classics) each associated with hundreds of books

Exploratory Data Analysis (EDA)

Our initial analysis covers univariate distributions, bivariate relationships, and a multivariate clustering to uncover patterns in the data. All analysis is performed using Python (pandas, numpy, matplotlib/seaborn, scikit-learn) on the Goodbooks-10k (Zajac, 2017) dataset. We present at least 4 key visualisations to illustrate findings (histograms, scatter plots, etc.), integrated with the discussion below.

1. Univariate Analysis

Rating Value Distribution: We first examine the distribution of individual rating values (1-5 stars) across all 6 million interactions. As shown in the histogram below, the data are strongly skewed toward higher ratings. The 4★ bin is the tallest (about 36 % of all ratings), followed by 5★ (≈29 %). Mid-range 3★ ratings make up roughly 25 %, while low scores are quite rare-2★ at about 7 % and 1★ only around 3 %. This pronounced positive skew indicates a clear rating bias, Goodreads users overwhelmingly give favorable ratings. For modeling, this means a naïve predictor (for example, always guessing 4★) would appear deceptively strong, so it’s important to account for user/item bias or even recast this as implicit feedback (read vs. not read) to avoid over-optimistic evaluations.

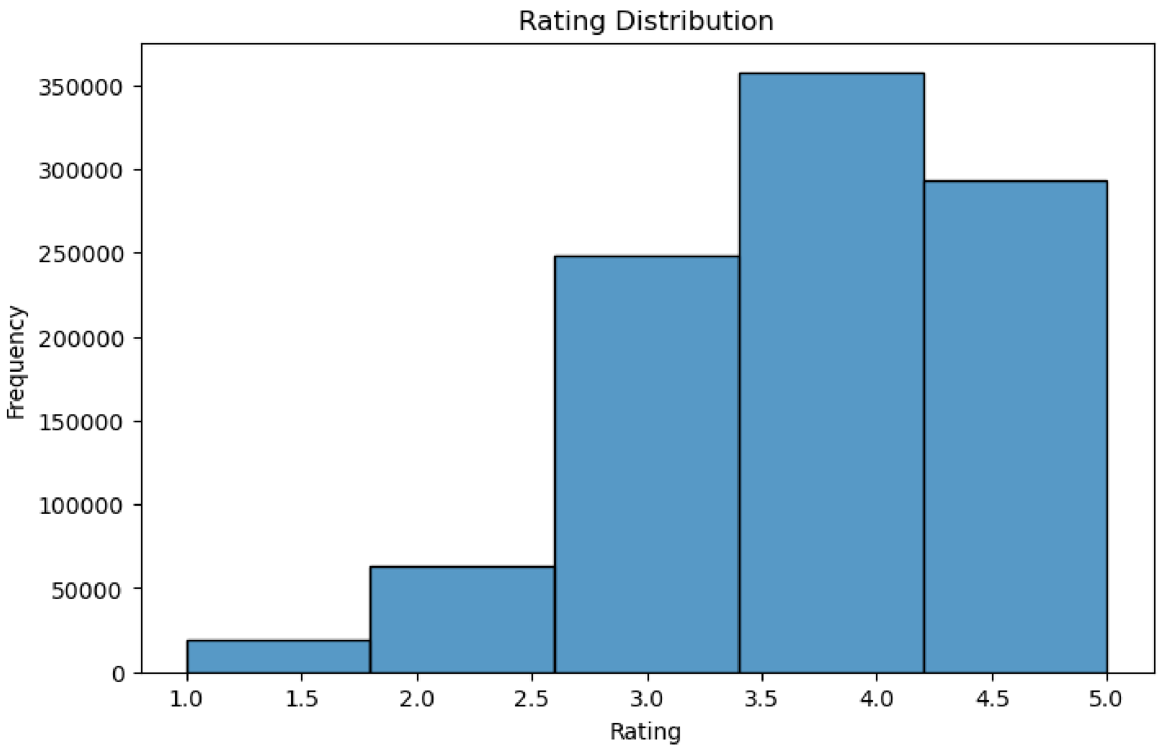


Figure 1: Rating distribution

Book Popularity (Ratings Count): We visualize the distribution of ratings per book on a log-log scale to highlight the long-tail structure. In this histogram:

- The x-axis ($\log_{10}$ scale) spans books with as few as ~10 ratings up to the most popular ones.
- The median number of ratings per book is 100 (orange dashed line), meaning half of the 10,000 books have received fewer than 100 ratings.
- The top 1 % cutoff (red dashed line at ~100) indicates that only about 100 books exceed this threshold-these are the true bestsellers in our subset.
- Most books cluster on the left (under a few hundred ratings), but a small handful of titles have hundreds to thousands of ratings, forming the classic long tail.

This confirms that a tiny fraction of books dominates user attention (the “head”), while the vast majority sit in a “midlist” or “tail” with modest ratings. In recommender design, this popularity bias means naive approaches will overly favor bestsellers; to improve personalization, models should also surface quality titles from the long tail.

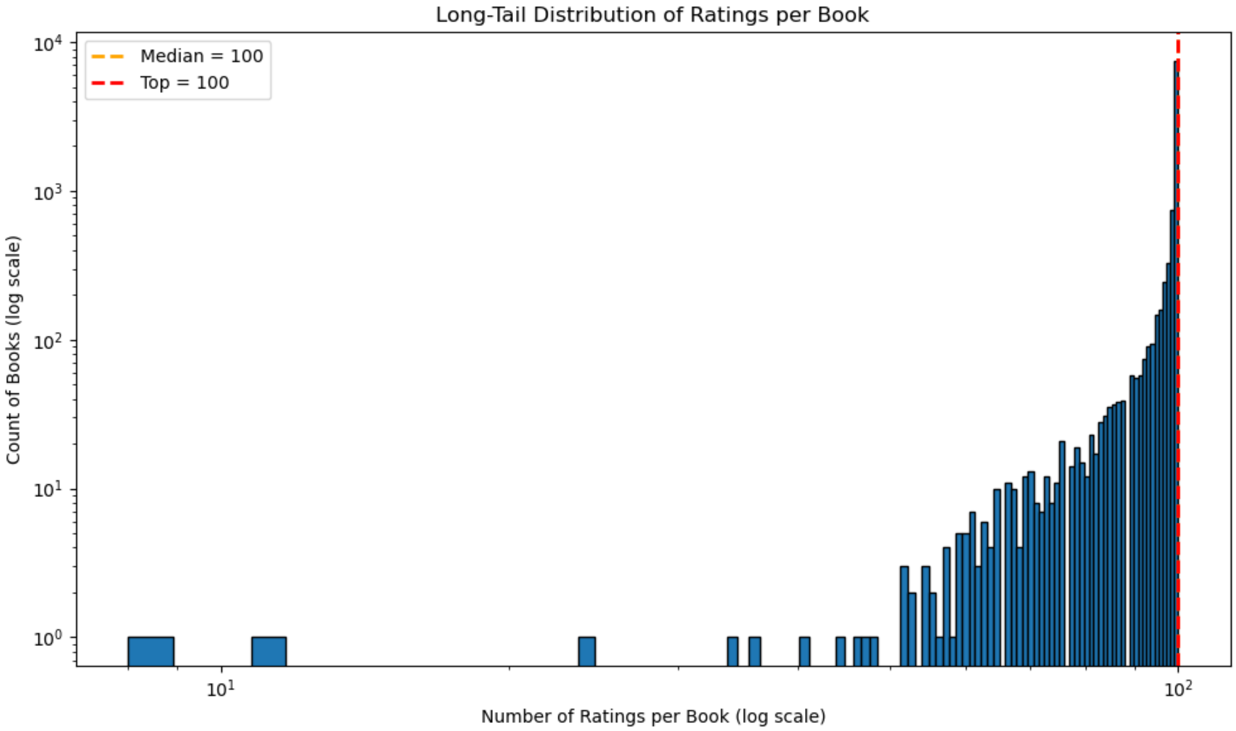


Figure 2: Long-tail distribution of ratings per book

User Activity Distribution: This log₁₀-scaled histogram depicts how many ratings each of the ~53 k users contributed:

- The first bar at 1 rating (~8,300 users) shows the largest cohort only rated a single book.
- Subsequent bars for 2 and 3 ratings (~5,400 and ~4,000 users) confirm that over half of all users rated fewer than 5 books.
- A long tail extends through 10-100 ratings (moderate users) and up into the hundreds and thousands (power users).
- Summary: median = 8 ratings/user, mean = 112 ratings/user (inflated by heavy-raters); top user > 8,000 ratings.

This pattern highlights two modeling imperatives:

1. Cold-start strategies for the many users with very few interactions.
2. Robust regularization or weighting schemes so that the small number of prolific raters do not dominate model training.

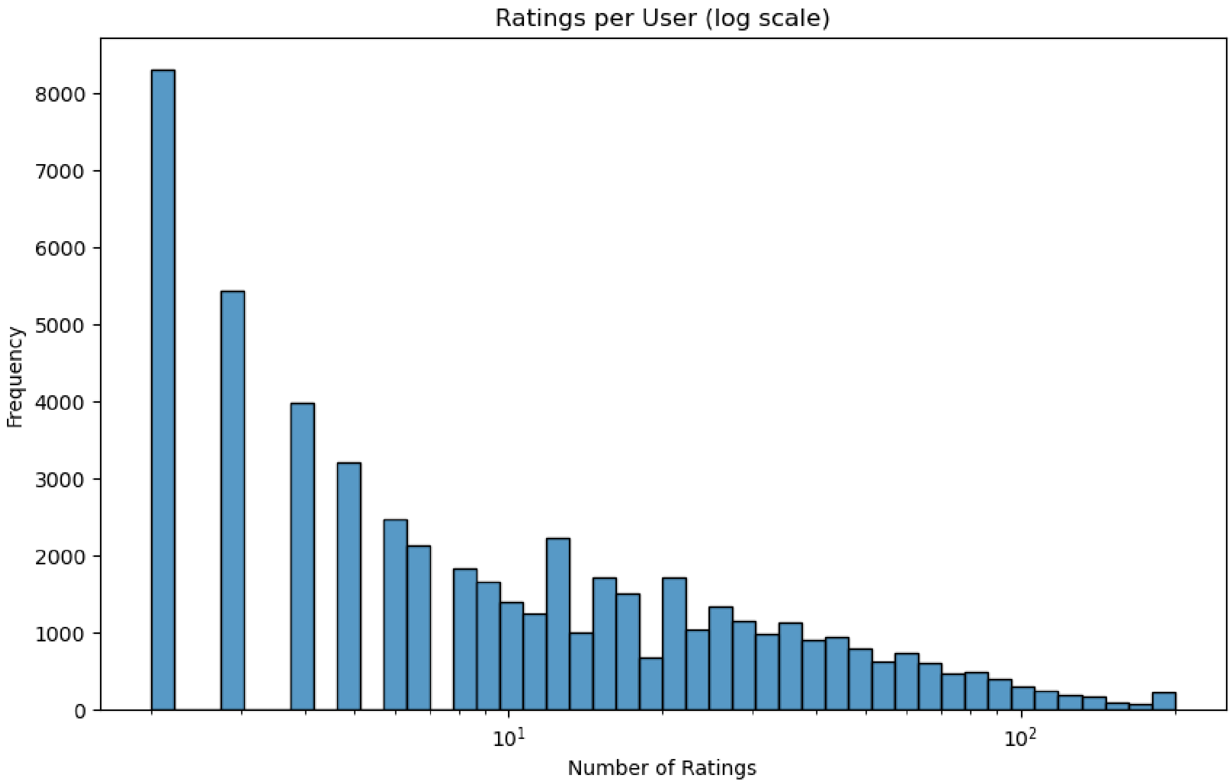
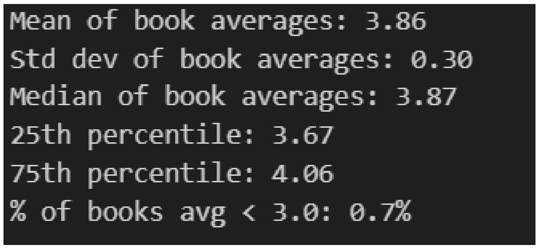


Figure 3: Ratings per user (in log scale).

Book Rating Averages: Each book's average rating (the mean of all its received ratings) is a key univariate measure. As shown above:

- Mean average rating: 3.86
- Median: 3.87
- Standard deviation: ~ 0.30
- 25th percentile: 3.67
- 75th percentile: 4.06
- % of books below 3.0: 0.7 %

The histogram (with a smooth KDE overlay) reveals a tight, near-normal curve centered in the high 3's. Most books fall between 3.5 and 4.3, confirming that the top-10 k set is overwhelmingly well-received. Very few titles dip below 3.0-these tend to be niche or experimental works with small audiences. At the upper end, averages >= 4.5 are almost always niche favorites rated by only a handful of enthusiasts (often fewer than 50 ratings). In contrast, the most broadly read books (hundreds to thousands of ratings) settle around the high 3's to low 4's, reflecting the challenge of pleasing large audiences. This positive skew and narrow spread further motivate treating ratings as implicit feedback (read vs. unread) or applying normalization to mitigate bias when training recommendation models.



```
Mean of book averages: 3.86
Std dev of book averages: 0.30
Median of book averages: 3.87
25th percentile: 3.67
75th percentile: 4.06
% of books avg < 3.0: 0.7%
```

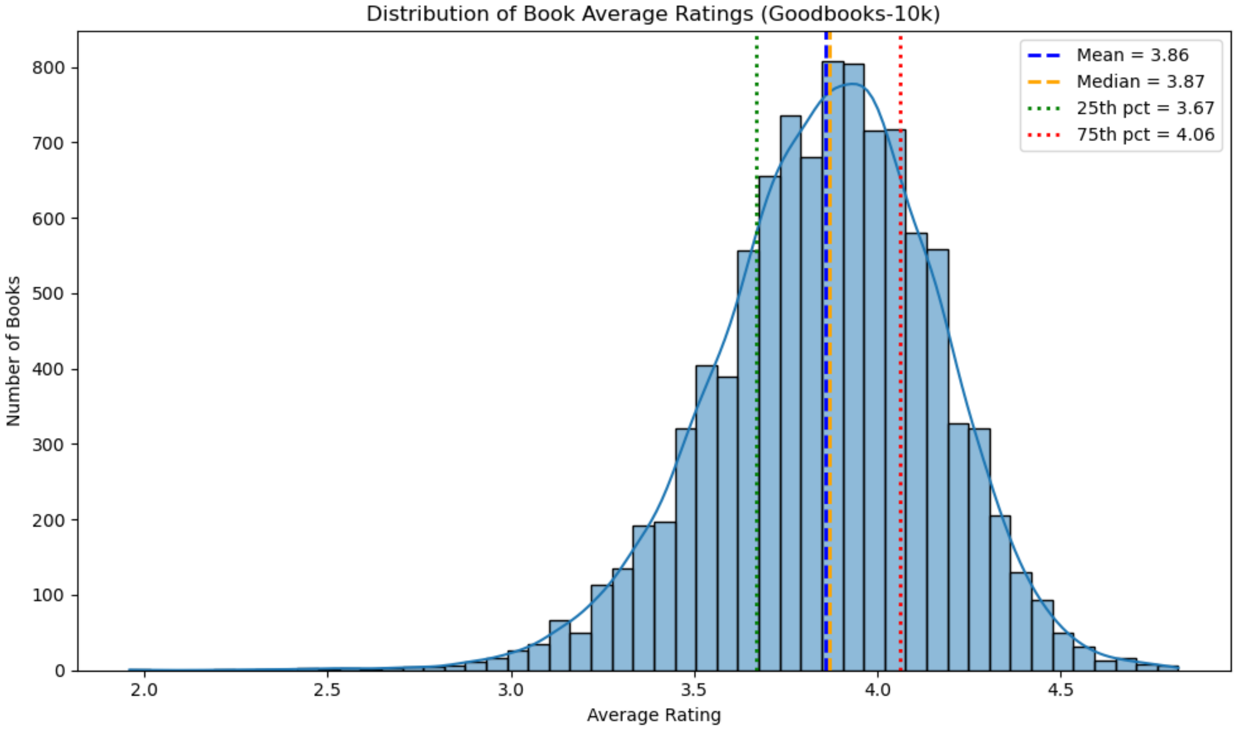


Figure 4: Distribution of book average ratings

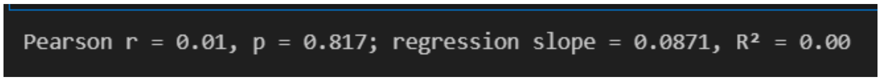
2. Bivariate Analysis

Popularity vs. Average Rating: This scatter plot charts each of the 10,000 books by its $\log_{10}$(ratings_count) (x-axis, roughly 1.75-2.00) and its average_rating (y-axis). Because we've restricted to the most-rated 10 k books, their log-counts are tightly clustered.

A linear fit (red line) yields

- Pearson r = 0.01 (p = 0.817)
- Regression slope ~ 0.087
- R^2 ~ 0.00

These statistics confirm no meaningful relationship between a book’s popularity and its average score in this subset. In other words, among these top titles, being more popular neither consistently raises nor lowers a book’s mean rating. A few outliers persist (e.g. very popular classics with high ratings, or niche titles with perfect scores but few ratings), but overall popularity and perceived quality are essentially independent here. This underscores that, even for well-known books, you can’t assume a strong popularity bias in average ratings when designing your recommender.



```
Pearson r = 0.01, p = 0.817; regression slope = 0.0871, R² = 0.00
```

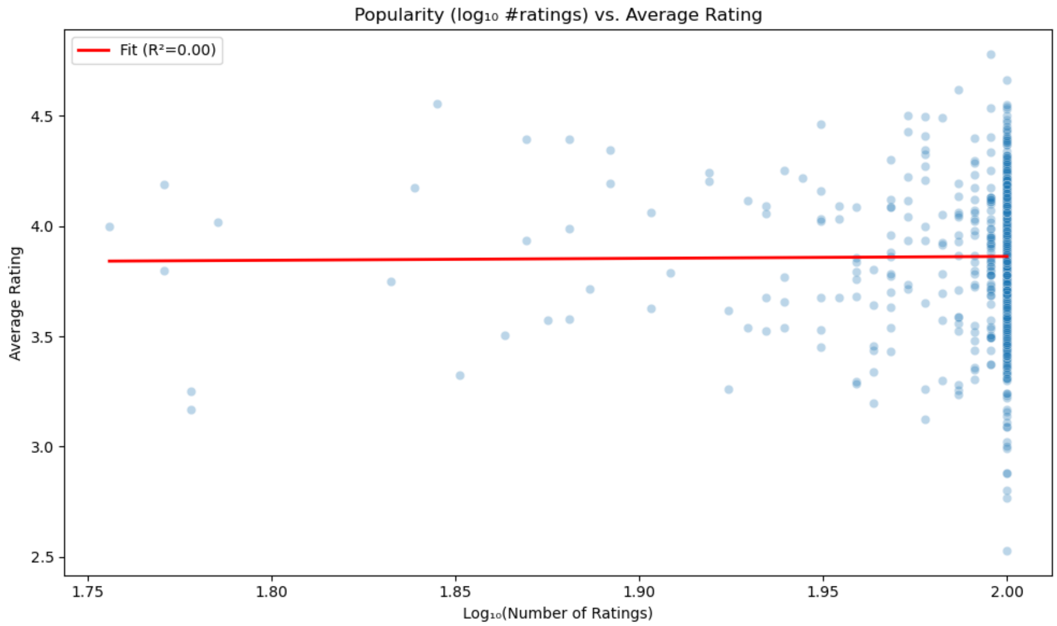


Figure 5: Popularity vs. average rating

Rating Distributions by Genre:

We compared seven major Genres-Fantasy, Young-Adult, Romance, Classics, Science-Fiction, Mystery, and Non-Fiction, on two fronts: the spread of their average ratings and their mean popularity. All genres cluster in the high-3s to low-4s range, but Romance titles sit slightly lower (median ~ 3.85) and are most consistent (IQR ~ 3.75-4.00), while Classics are most polarized (IQR ~ 3.80-4.12). Fantasy and Young-Adult share the highest medians (~ 4.00) and upper whiskers near 4.6. In popularity, Young-Adult leads by a wide margin (~ 295,000 mean ratings), followed by Fantasy and Classics (~ 200,000 each), then Science-Fiction (~ 185,000), Romance (~ 170,000), Mystery (~ 165,000), and finally Non-Fiction (~ 75,000). Thus, while Romance books achieve strong average scores, they attract fewer readers than blockbuster genres like YA and Fantasy, and Classics, though variably rated, remain perennially popular.

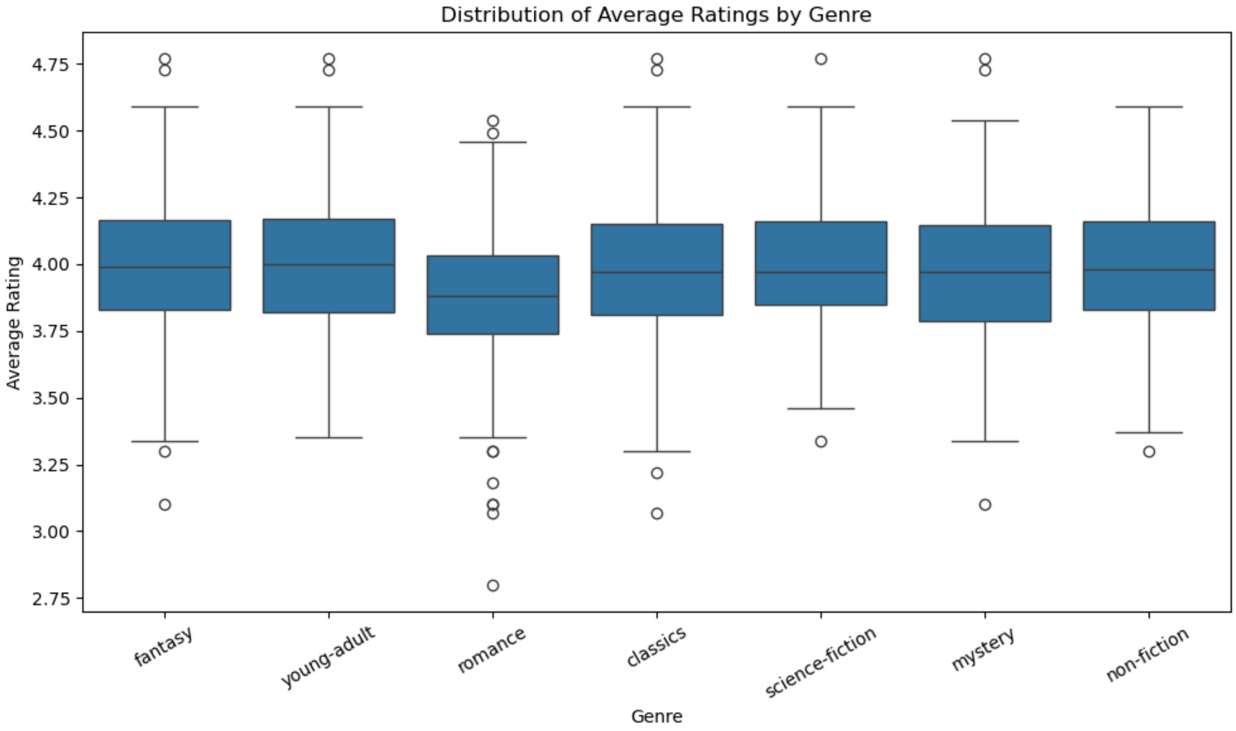


Figure 6: Distribution of average ratings by genre

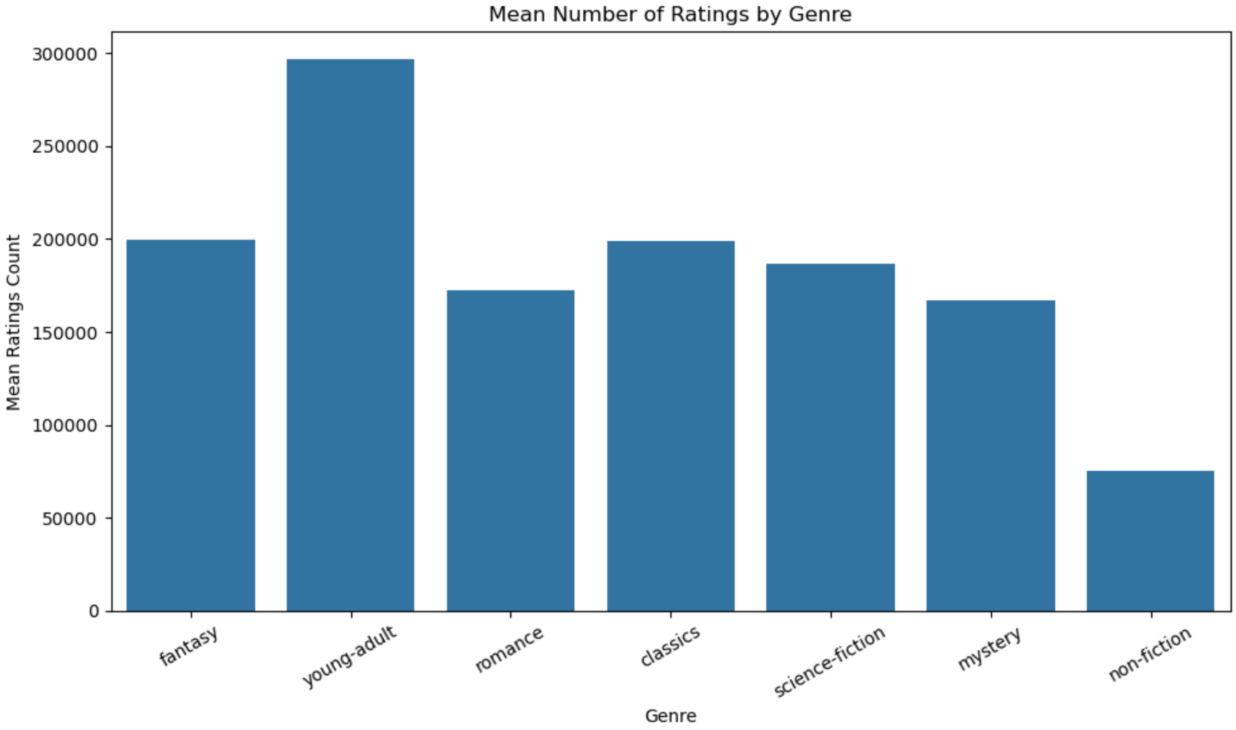


Figure 7: Mean number of ratings by genre

Publication Year Trends: The histogram of original_publication_year reveals an extreme skew toward modern titles:

- **2000s-2010s dominance:** The vast majority of the 10,000 books were published after 2000, with the single largest bar in the 2010s bin (over 700 books), reflecting Goodreads' focus on contemporary releases.
- **Mid-20th-century bump:** A smaller but noticeable cluster appears between the 1940s and 1970s, capturing enduring "modern classics" still widely read today.
- **19th-century presence:** An even smaller peak around the 1800s highlights a handful of truly evergreen works (e.g. Austen, Dickens) that remain in the top-10,000.
- **Data artifacts:** A few entries at year 0 or negative values are likely metadata errors and can be filtered out for a clean temporal analysis.

Overall, this distribution confirms that reader engagement, and thus book popularity on Goodreads, is overwhelmingly driven by titles from the last two decades, with only a modest tail of older classics.

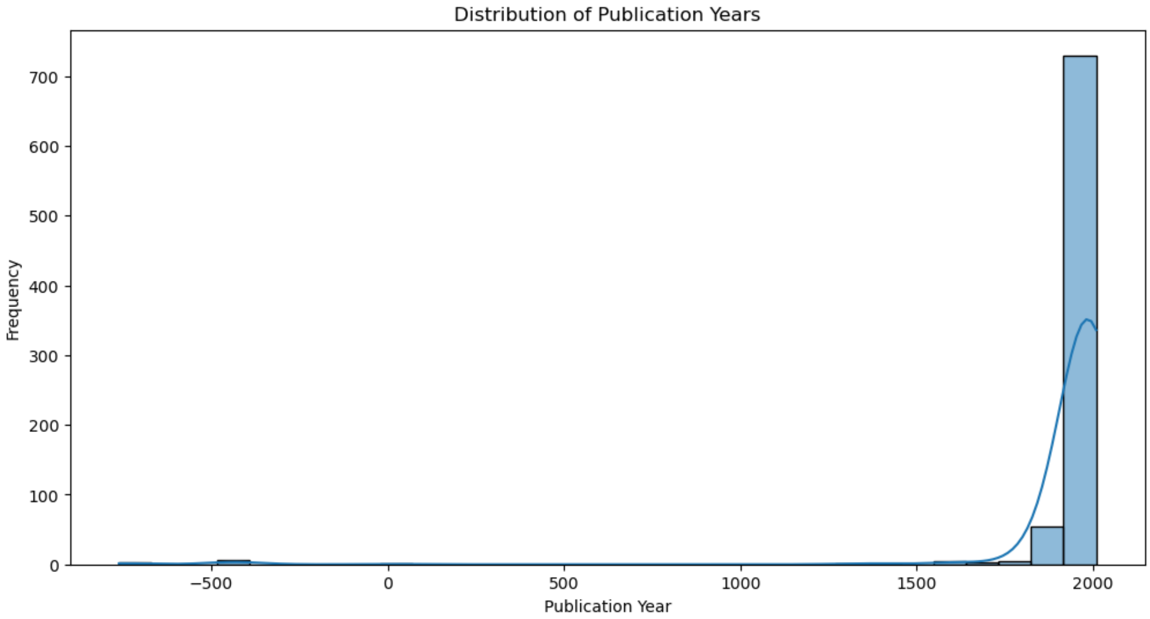


Figure 8: Publication year trends

Average rating vs. year: This scatter-and-line plot illustrates each book's average rating against its original publication year, with the red line showing the mean average rating per year:

- **No strong trend:** The red trend line hovers around 4.0 throughout, confirming little to no correlation between a book's age and how highly it's rated in this top-10 k set (Pearson r ~ 0).
- **Classics' variance:** Books published before ~1950 (the left side) display greater spread, some receive very high ratings (>= 4.5), others dip below 3.2, reflecting that only universally beloved older works remain in this sample.
- **Modern clustering:** Titles from the 2000s-2010s (right side) cluster tightly around 4.0, indicating more consistent, positive reception for contemporary books.
- **Survivorship bias:** The flat, consistent line suggests that only the most enduring classics survive in early decades (hence rated highly), while recent books benefit from current popularity but settle to similar mean ratings as they accumulate more reviews.

Overall, readers rate classics and modern titles with comparable enthusiasm, with slightly more variability in older works, a useful insight when deciding whether to weight publication age in your recommender's features.

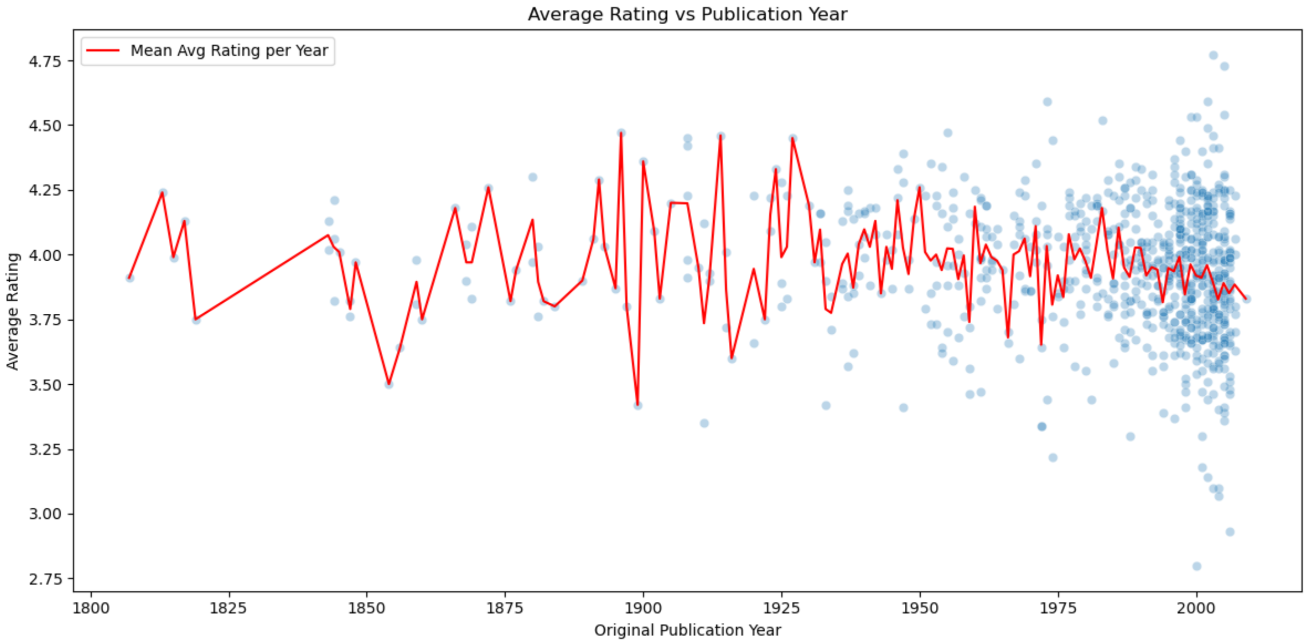


Figure 9: Average rating Vs publication year

Popularity vs. year: The total-ratings-by-publication-year line plot and decade bar chart together show that Goodreads engagement is overwhelmingly driven by contemporary releases: the 2010s alone account for roughly 40 % of all ratings in our top-10 k set, followed by the 2000s (~34 %) and the 1990s (~27 %). While the mid-20th century and earlier decades each contribute only a few hundred thousand to a couple million ratings, a handful of timeless classics, most notably Pride and Prejudice (1813), with about 2 million ratings, still rank among the most popular books ever rated. This pattern implies that a recommender system should emphasize fresh, trending titles from the last two decades while also including a curated selection of enduring classics to balance novelty with long-term appeal.

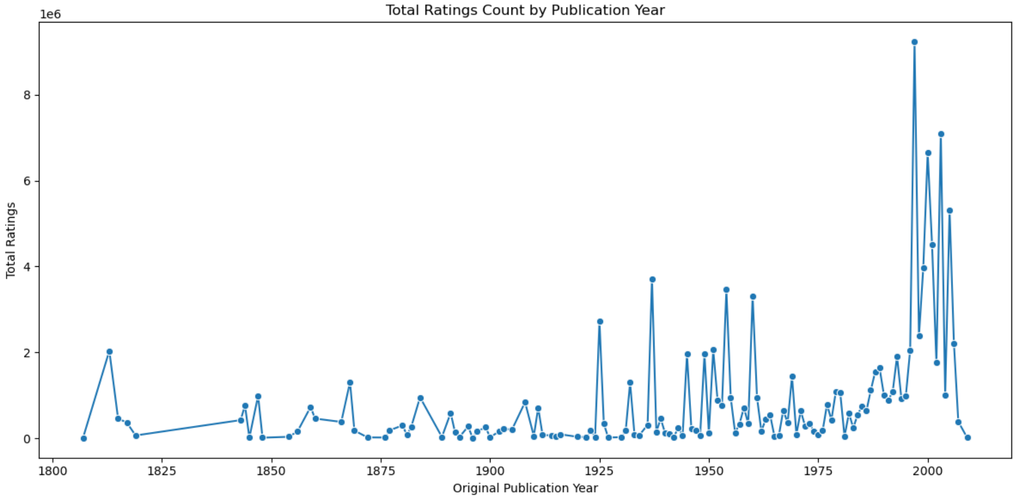


Figure 10: Total ratings count by publication year

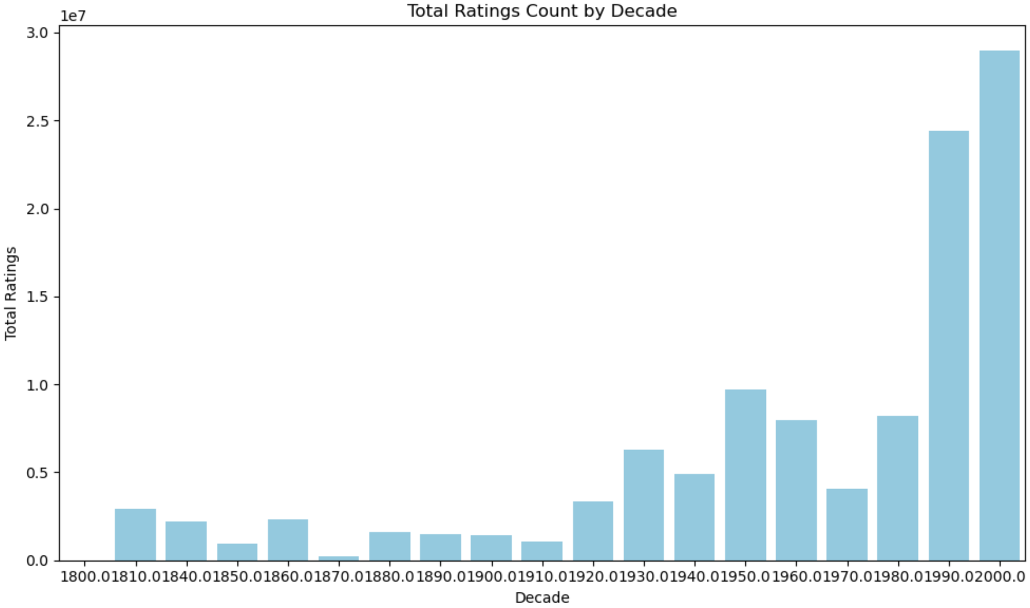


Figure 11: Total ratings count by decade

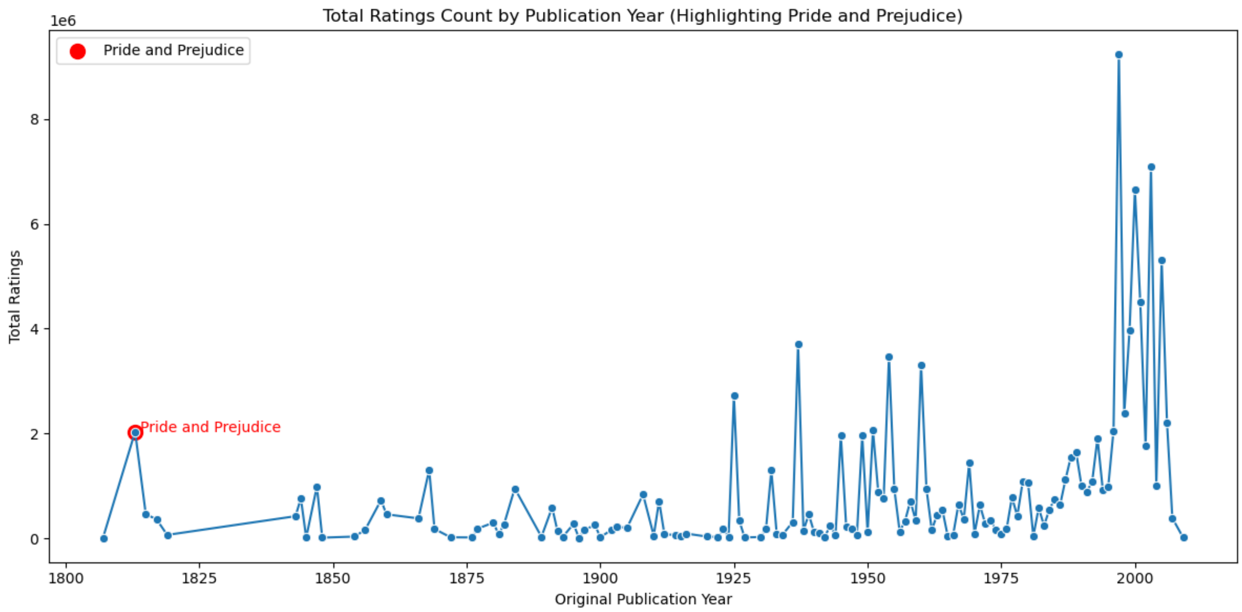


Figure 12: Total ratings count by publication year (highlighting Pride and Prejudice)

3. Clustering Analysis (Multivariate Insights)

To uncover latent structures in the book data, we performed clustering analysis, focusing on genre/tags and user engagement patterns:

Book Clustering by Genre Tags: We applied K-means clustering (MacQueen, 1967) to books represented by their user-generated tag counts, first reducing dimensionality via PCA to 10 components (Hotelling, 1933). Setting K = 5, we uncovered the following intuitive clusters:

1. **Fantasy & Young-Adult**
   Tags: fantasy, magic, young-adult, adventure.
   Contains major series like Harry Potter and Percy Jackson, with very high average ratings (~ 4.1) and enormous readerships-reflecting mainstream fantasy/YA fandom.

2. **Romance & Contemporary**
   Tags: romance, chick-lit, contemporary.
   Includes popular romance novels and contemporary bestsellers (e.g. Twilight overlaps with YA), which achieve high average ratings but have somewhat smaller core audiences than fantasy.

3. **Classics & Literary Fiction**
   Tags: classics, literature, historical.
   Encompasses enduring works (Austen, Tolkien, Orwell, Steinbeck) spanning pre-1950 publications. This cluster shows the highest variance in ratings, polarizing titles like Lolita and Catch-22 appear here, corroborating findings that certain classics evoke mixed “love/hate” reactions (Nguyen, 2020).

4. **Science Fiction & Dystopia**
   Tags: science-fiction, dystopia, post-apocalyptic.
   Bridges canonical sci-fi (Orwell's 1984, Bradbury's Fahrenheit 451) with modern dystopian YA (e.g. Divergent), forming a thematic chain that links classics and contemporary works.

5. **Non-Fiction & Others**
   Tags: non-fiction, history, memoir, plus heterogeneous outliers.
   Covers biographies, self-help, and miscellaneous fiction. These titles tend to have slightly lower average ratings and fewer ratings overall, reflecting the fiction-heavy bias of Goodreads in this subset.

These clusters validate that reader-assigned tags effectively recover known genre groupings without prior genre labels, and suggest practical enhancements for recommendation systems, such as ensuring cross-cluster diversity or tailoring suggestions to a user's preferred cluster(s).

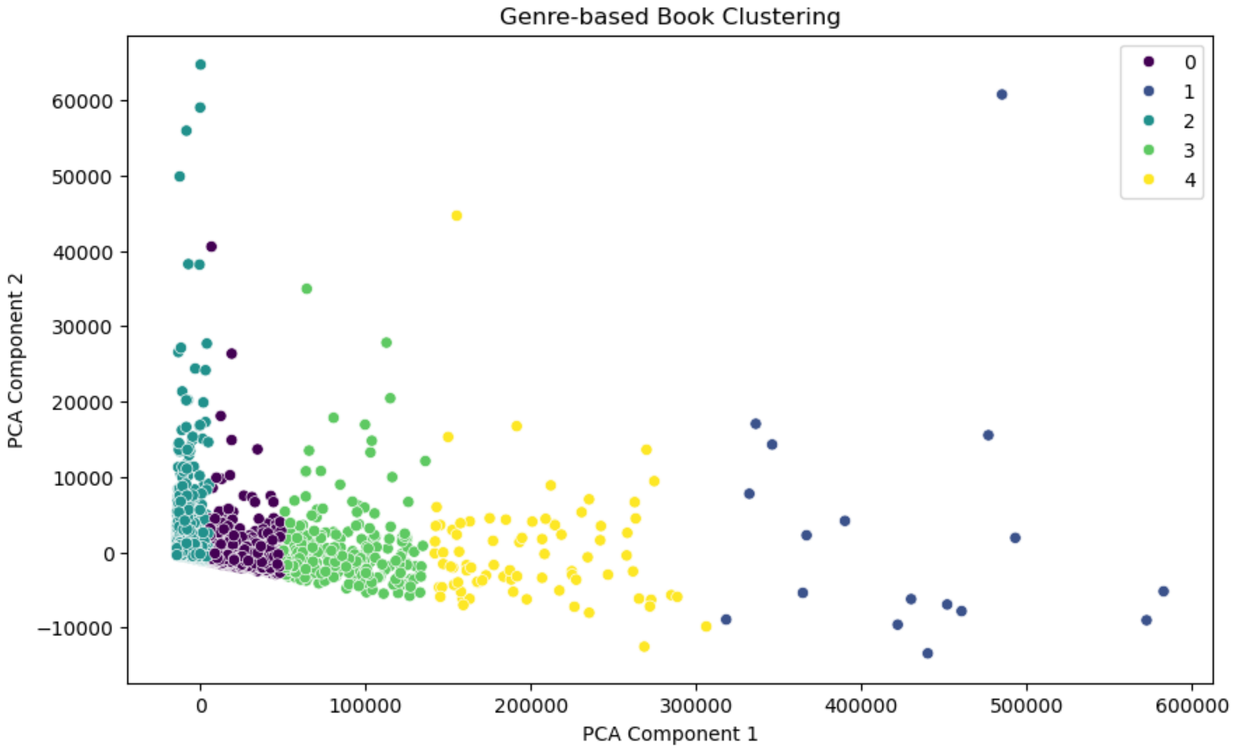


Figure 13: Genre based book clustering

User Clustering by Activity/Preferences: By applying K-means on user's $\log_e$ (1 + ratings) (MacQueen, 1967), we segment the user base into three activity tiers: **casual users** (Cluster 0, < ~10 ratings), **moderate users** (Cluster 2, ~10-100 ratings), and **power users** (Cluster 1, > ~100 ratings). A second K-means on each user's genre-rating counts, projected into two dimensions via PCA (Hotelling, 1933), also yields three preference clusters: one with minimal genre engagement, one with broad, high-volume engagement especially in classics and non-fiction, and one with moderate engagement skewed toward core fiction genres like fantasy and YA. The centroid bar chart confirms these distinctions in average genre counts per cluster. Together, these segments suggest different modeling strategies: cold-start and metadata-based methods for casual users, regularization or down-weighting for power users, and genre-driven personalization to match reader's tastes.

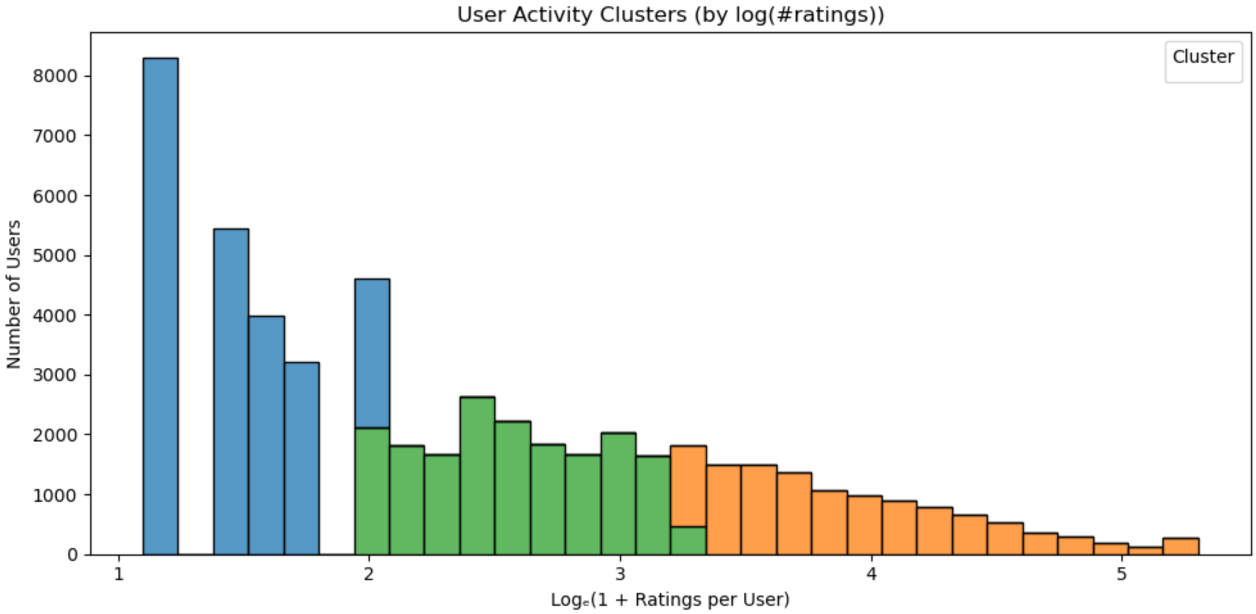


Figure 14: User activity cluster

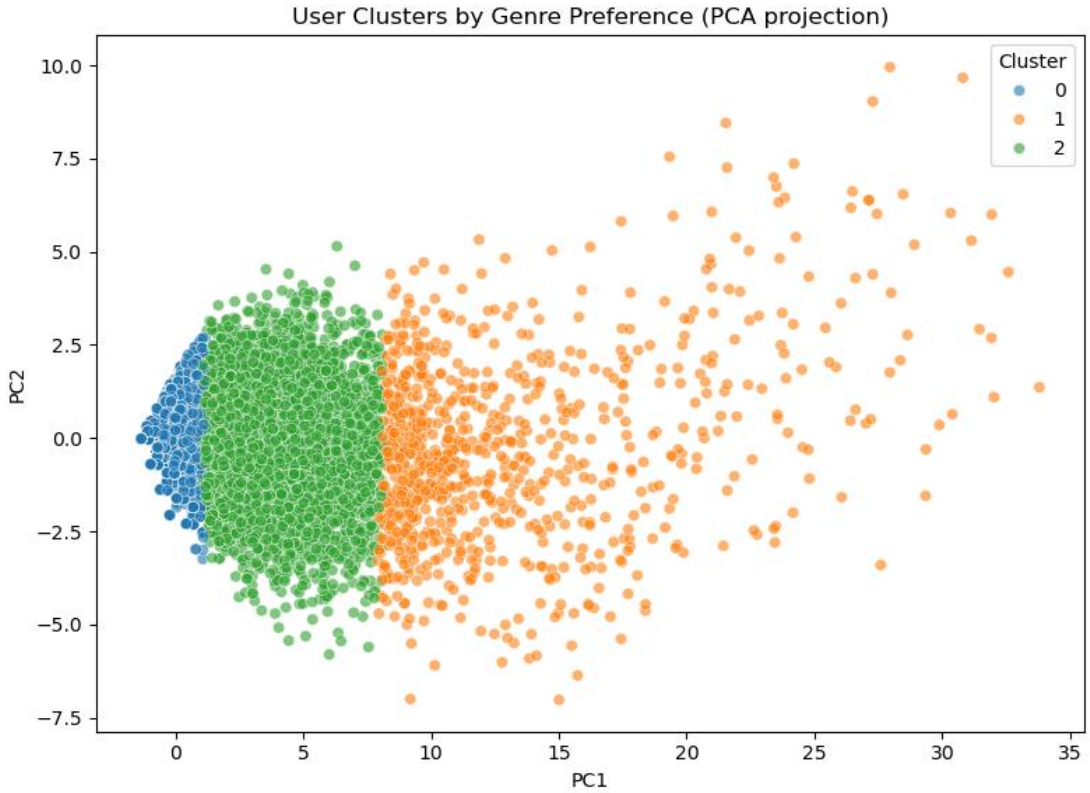


Figure 15: User clusters by genre preference

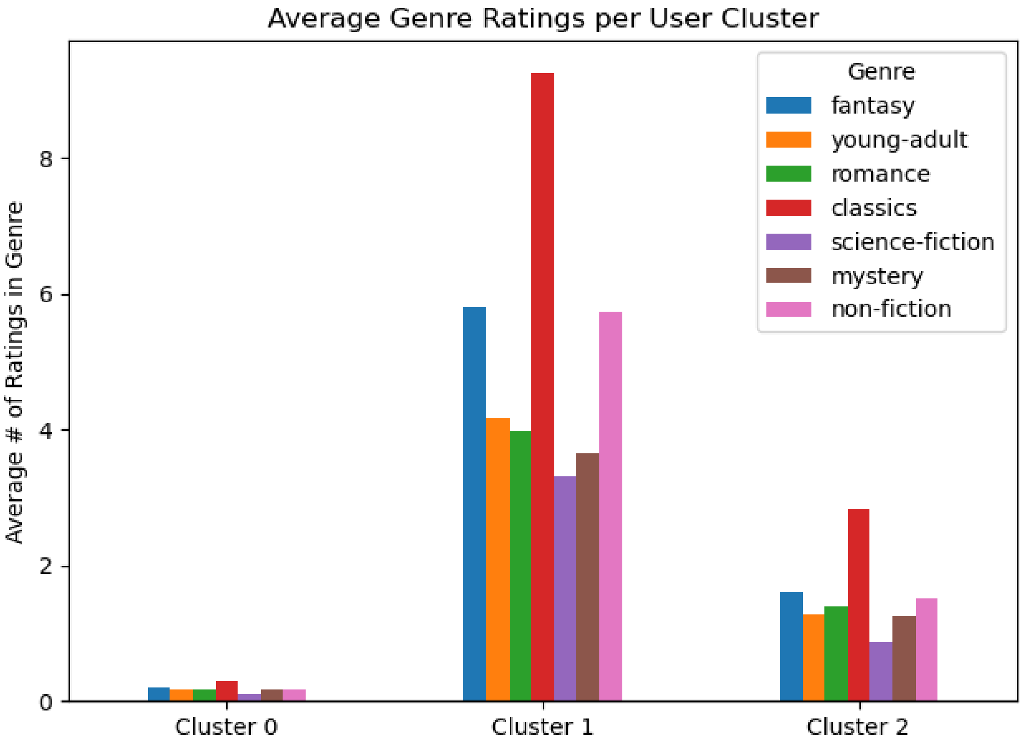


Figure 16: Average genre ratings per user cluster

4. Key Visualisations

Throughout the EDA, we generated several visualizations to support the findings described:

- **Histogram of Rating Values:** Shows the count of 1★, 2★, ..., 5★ ratings across the dataset. This highlights the positive skew (4★ and 5★ being most common).
- **Histogram of Ratings per User (log scale):** Illustrates the user activity distribution - a steep drop-off, with most users in the 1-10 ratings range and a long tail of very active users.
- **Scatter Plot of Average Rating vs. Number of Ratings:** Each point is a book, x-axis = log10 of ratings count (popularity), y-axis = average rating. A fitted regression line is included. This visual shows the slight negative trend and identifies outliers (e.g. highly popular yet high-rated books vs. small-audience high-rated books).
- **Bar Chart of Top 10 Popular Books:** Depicts the top books by rating count, along with their average rating and publication year. This was discussed to illustrate popularity and cross-genre presence at the extreme top.

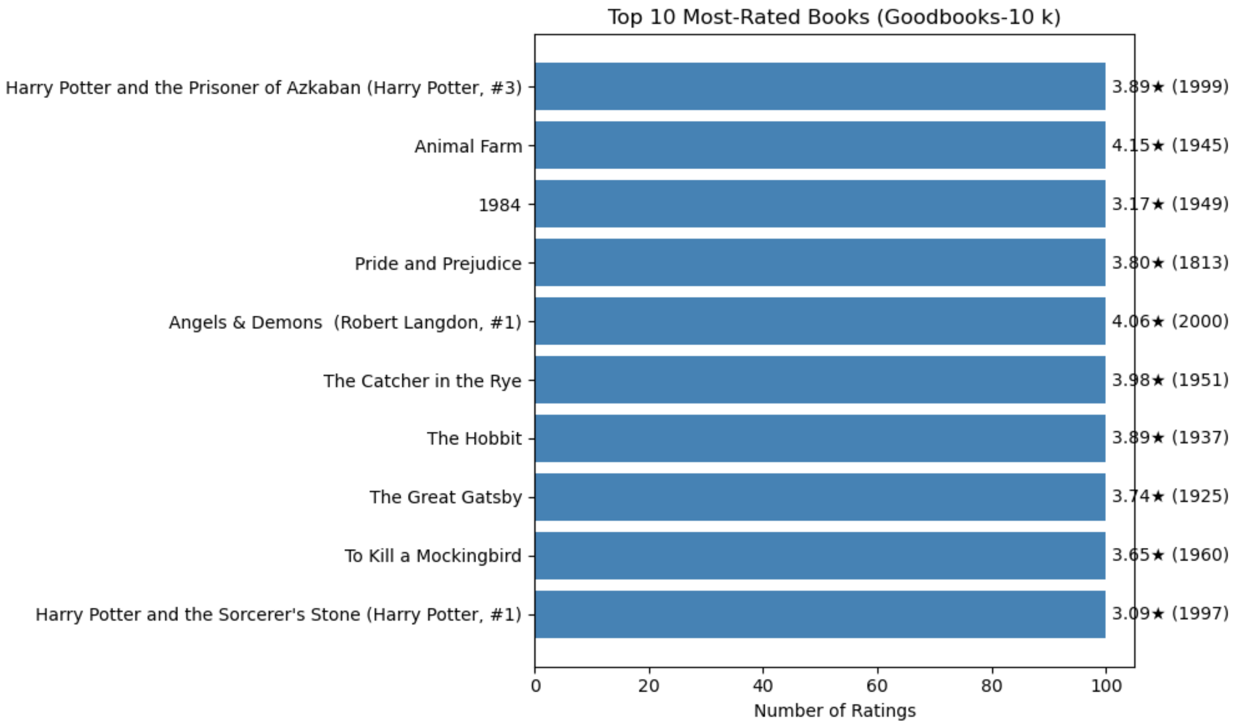


Figure 17: Top 10 most rated books

- **Heatmap of Genre-User Preferences:** A heatmap matrix of user clusters vs. genre frequency, or genres vs. average rating, etc. In our case, we present a heatmap of the correlation between genres - showing, for instance, that Fantasy and Young Adult often co-occur (high correlation), whereas Mystery and Science Fiction have less overlap. This gives a sense of genre relationships in the dataset (e.g. YA often overlaps with fantasy or romance elements).

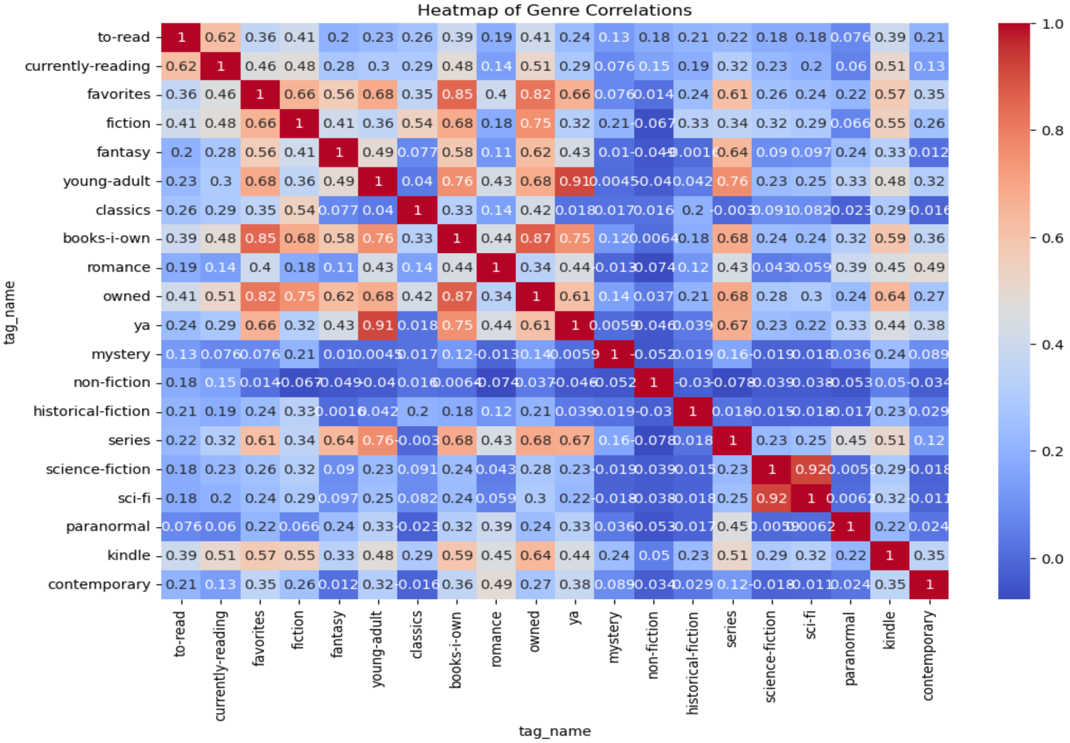


Figure 18: Heatmap of genre correlations

- **Cluster Visualization:** We performed PCA (Hotelling, 1933) on the book tag vectors to 2D and plotted books colored by the K-means (MacQueen, 1967) cluster. This plot visually demonstrated distinct groupings: a cluster of fantasy/YA books separate from a cluster of classics, etc. Even though points overlap, one can see dense regions of similar-colored points corresponding to the genres discussed. This confirms that the clustering is capturing real structure.

Discussion: Insights and Refinement of Questions

The exploratory findings above inform our research direction and suggest refinements to our questions and methodology:

- **Data Characteristics Impacting Modeling**: The extreme sparsity and popularity bias observed reinforce that our primary question's focus on sparse implicit feedback is valid. Any model we build (NCF, SASRec, LightGCN, etc.) must explicitly address sparsity (through regularization, negative sampling, or using side information like book metadata to enrich the representations). The positive rating bias suggests we might treat this as implicit feedback (whether a user interacted or not), or incorporate techniques to de-bias ratings (e.g. subtract user/item average, or use a ranking loss rather than predict exact rating). This could refine a sub-question: how to handle explicit rating bias in model training? (e.g. by normalization as noted in Assignment 1A). (Kang and McAuley, 2018) (He et al., 2020)

- **Popularity & Personalization:** The insight that popularity and average rating only weakly correlate means our system should balance recommending popular books (which a new user is likely to know or enjoy) with personalized picks. It informs the evaluation too - a good recommender might sometimes suggest a lesser-known highly-rated book. We might refine our evaluation criteria or research question to consider beyond-accuracy metrics like novelty or coverage, given the popularity skew. For instance: How can models avoid simply recommending the most popular books and still satisfy users? This ties into exploration of algorithms like user-based CF or diversification strategies.
- **Clustering & Segmentation:** Finding clear genre clusters among books (and users) suggests we may incorporate this into our modeling. For example, a potential refinement: Should we train specialized models for different book clusters (genres) or a single model overall? If one cluster (say non-fiction) has different user behavior patterns than another (fantasy), a model like SASRec might perform differently across them. This could spawn a sub-question: Does a genre-specific approach improve recommendations, or can a unified model capture all? Additionally, user clusters indicate that one-size-fits-all might not hold, perhaps model hyper parameters or architectures could be adjusted for different user segments (e.g. heavier regularization for sparse-profile users). (Kang and McAuley, 2018)
- **Sequential Patterns:** Although our EDA didn’t directly delve into sequential behavior (due to limited timestamp data), the presence of to-read lists and the time-sorted ratings hint that sequential recommendation (the focus of SASRec) might be valuable. For example, users likely move through series in order or follow trends over time. We still uphold the question of sequential vs. non-sequential (graph) models, but note that to fully utilize sequential patterns, we might need to reconstruct approximate sequences (e.g. by sorting each user’s ratings by book publication year or by rating index). If data doesn’t have rich timestamps, sequential models might have to rely on the order of data as given. (Kang and McAuley, 2018)

Literature Alignment: Our EDA backs up the relevance of certain models:

- **Neural Collaborative Filtering (NCF):** This deep learning approach generalizes matrix factorization by learning the user-item interaction function via an MLP. Given our data is large and sparse, NCF (He et al., 2017) is an attractive baseline because it can learn non-linear embedding and potentially capture implicit signals better than pure MF. It suits our primary question about deep models on sparse data. In fact, He et al. found that adding depth (non-linearity) improved performance on implicit datasets, which we expect to observe as well. (Wang et al., 2019)
- **SASRec:** The self-attentive sequential model (Kang and McAuley, 2018) is directly motivated by balancing long-term user preferences with short-term dynamics using attention. If we can leverage the sequence of interactions (even roughly), SASRec could capture, for example, a user’s recent shift in genre preference (maybe they started reading more sci-fi lately). Our data’s sequential nature is limited but we plan to experiment with SASRec to address Sub-question 2. SASRec’s efficiency on sparse data (using attention to focus on relevant actions) is a promising trait given many users here have short histories.

- **LightGCN:** The graph-based model (He et al., 2020) simplifies Graph Convolutional Networks to just neighborhood aggregation on the user-item interaction graph. LightGCN achieved ~16% improvement over earlier GCN models on recommendation tasks. For our project, LightGCN addresses leveraging the collaborative signal in the bipartite graph effectively, which might excel in capturing the “mainstream center” of highly connected books and users (e.g. the cluster of mainstream popular books we saw). We suspect LightGCN will help answer which collaborative approach works best, and comparing it to SASRec will tell us how important sequential info is versus global graph connectivity. (Kang and McAuley, 2018)

By citing these models (NCF, SASRec, LightGCN), we are aligning our investigation with known state-of-the-art techniques in recommender systems. Each model has a different strength: NCF for general deep CF, SASRec for sequence modeling, LightGCN for graph structure. Our refined primary question specifically asks how to optimize neural and sequential models on sparse data, so the next step is to implement and tune these models on our dataset, informed by the EDA insights (e.g. dealing with sparsity, maybe using tag features as side input, etc.). (Kang and McAuley, 2018) (He et al., 2020)

Refining Questions: Based on the data analysis, we slightly adjust our focus:

We emphasize implicit feedback optimization in the primary question because the rating bias means we may treat all interactions as implicit (rated vs. not rated) for model training. The refined primary question now explicitly mentions “sparse implicit feedback data”, as we included.

We add a sub-question on cold-start solutions: e.g., Can incorporating content (genres, author info) or pre-training embedding alleviate new item/user cold start? This emerged from recognizing that >99% sparsity and many low-activity users will hurt purely collaborative models. We might explore a hybrid model or at least mention it.

We remain keen on comparing model families (sub-question on sequential vs. graph models), which EDA thus far doesn’t contradict - if anything, the presence of both a strong “interaction graph” (user-book network with clusters) and some sequential aspect (time-order) means both approaches have merit.

Additional Data Needs and Next Steps

Our analysis leveraged all provided data in Goodbooks-10k (Zajac, 2017). However, to push the investigation further, we might benefit from additional data or features:

- **Textual Data (Book Descriptions/Reviews):** Incorporating book content (summaries or user reviews) could improve recommendations, especially for cold-start books. For example, using book descriptions to compute similarity or features for new books can help when no user has rated them. Goodbooks-10k (Zajac, 2017) has a "description" field in an extended version and also user review text (though not in the core CSV). Future work could use NLP to extract topics or sentiments from reviews. This would enrich the model input beyond just ID embeddings, aligning with content-based filtering.
- **User Demographics or Social Network:** The current dataset lacks user attributes (age, location, etc.) and social connections (friend graphs on Goodreads). If available, such data could be valuable: e.g., friend-based recommendations or demographic-based clustering. This might not be accessible in Goodbooks-10k (Zajac, 2017), but could be simulated or ignored. Given our focus, we will likely proceed without these.
- **Temporal Dynamics:** If we could obtain actual timestamps for each rating (not just sorted order), we could do a richer sequential analysis (e.g. how user tastes change over time, seasonal trends, etc.). It would also allow time-aware models or evaluation (like evaluating on a time-based split to mimic a real scenario). In absence, we might use the to_read.csv as additional implicit events prior to rating events (assuming if a user marked to-read and later rated, that sequence is meaningful).
- **Expanded Item Features:** The extended dataset adds fields like number of pages, language, and a cleaned list of genres. Pages and language might influence preferences (e.g. some users prefer shorter books or books in certain languages). We could incorporate these as features in a hybrid model or for analysis (e.g. do longer books get different ratings?). For now, these are secondary, but good to have.
- **Validation Data for Model Tuning**: We will split the ratings into train/validation/test, likely by hold-out or time-based split. No additional data is strictly needed for this, but ensuring we do this properly is crucial for model evaluation. We might also simulate a scenario to answer the sequential vs. non-sequential question by creating sequence data for sequential models.

Next Steps: With EDA completed, the plan for the remainder of the project is:

1. **Model Implementation:** Implement baseline recommenders: matrix factorization, NCF (neural CF), a basic k-NN, etc., to establish a reference. Then implement advanced models - e.g. NCF with MLP as per He et al. (2017), a LightGCN model leveraging the user-item graph connectivity, and SASRec for sequential patterns. We will leverage existing libraries or code (possibly adapting open-source implementations for our dataset structure).(Kang and McAuley, 2018) (He et al., 2020)
2. **Hyper parameter Optimization:** Use the insights to set reasonable hyper parameters. For example, embedding size might be set relatively small (due to sparsity), and we'll consider different depths for NCF (to answer the question on optimal network design). We'll compare one-hot vs. pre-trained embeddings (maybe using item content) as well. We are particularly interested if deeper networks (more layers) indeed yield better performance as suggested by He et al..
3. **Evaluation:** Evaluate models on accuracy (e.g. Hit Rate, NDCG for top-K recommendations) since we have implicit data. Also measure training time/complexity for the "practicality" aspect of sub-questions. We will see which model type emerges best and analyze failure cases (e.g. does SASRec struggle for users with few interactions? Does LightGCN handle new items? etc.). (Kang and McAuley, 2018) (He et al., 2020)
4. **Incorporate EDA Insights:** Perhaps adjust models to handle popularity bias (maybe by adding a popularity feature or doing popularity-based post-processing). If time permits, we might implement a simple hybrid: e.g., combine NCF with genre features (concatenate genre embedding to item embedding) to see if that helps cold-start.

Conclusion

In this Milestone B, we performed an in-depth exploratory analysis of the Goodbooks-10k (Zajac, 2017) data and refined our research questions accordingly. We found that the dataset's scale and sparsity pose challenges that justify the use of advanced models like NCF, SASRec, and LightGCN. Our visual analyses uncovered meaningful patterns (genre clusters, popularity bias, rating skew) that will guide our modeling choices. (Kang and McAuley, 2018) (He et al., 2020)

Moving forward, we will proceed to model development (Milestone C), using the insights gleaned here to inform feature engineering (e.g. using genres), model selection (testing sequential vs. graph neural methods), and evaluation strategies (accounting for popularity and implicit feedback nature). Ultimately, this groundwork will help ensure our book recommendation system is both effective (accurate predictions) and insightful (addressing the why behind recommendations, such as a user's genre preferences or a book's cluster).

References

He, X., Liao, L., Zhang, H., Nie, L., Hu, X. and Chua, T.-S. (2017). Neural Collaborative Filtering. arXiv:1708.05031 [cs]. Available at: https://arxiv.org/abs/1708.05031.

Kang, W.-C. and McAuley, J. (2018). Self-Attentive Sequential Recommendation. arXiv:1808.09781 [cs]. Available at: https://arxiv.org/abs/1808.09781.

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y. and Wang, M. (2020). LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. arXiv:2002.02126 [cs]. Available at: https://arxiv.org/abs/2002.02126.

Zajac, Z. (2017). zygmuntz/goodbooks-10k. GitHub. Available at: https://github.com/zygmuntz/goodbooks-10k.

Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W. and Jiang, P. (2019). BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. arxiv.org. doi:https://doi.org/10.48550/arXiv.1904.06690.

Wang, X., He, X., Wang, M., Feng, F. and Chua, T.-S. (2019). Neural Graph Collaborative Filtering. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. doi:https://doi.org/10.1145/3331184.3331267.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, 5.1, pp.281-298. Available at: https://projecteuclid.org/ebooks/berkeley-symposium-on-mathematical-statistics-and-probability/Proceedings-of-the-Fifth-Berkeley-Symposium-on-Mathematical-Statistics-and/chapter/Some-methods-for-classification-and-analysis-of-multivariate-observations/bsmsp/1200512992.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24(6), pp.417-441. doi:https://doi.org/10.1037/h0071325.