Book Recommendation System Using Goodreads Dataset

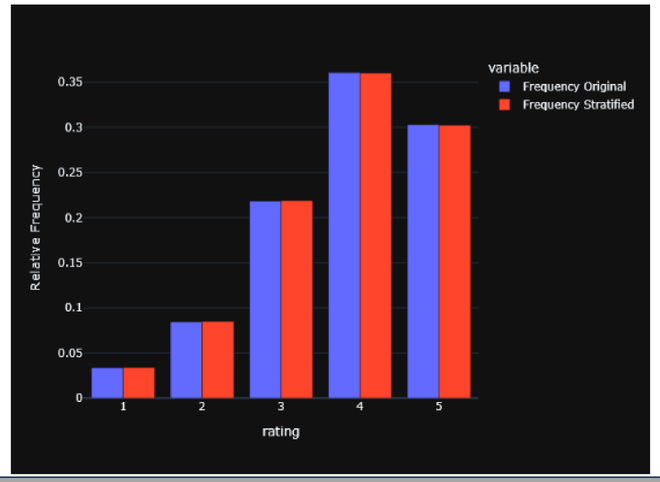
Neil Ghugare
ghugare.1@osu.edu
Department of Physics, The Ohio State University

Motivation

Book enthusiasts all have one common problem: it is extremely difficult to find new, enjoyable books. Not only does it waste time trying to read book descriptions and titles trying to conclude if they would enjoy that book, but it could also cost a lot of money. Being a fellow book enthusiast, it would be extremely useful if some sort of program could take book reviews or ratings, and then spit out books that others have found enjoyable based on that singular book (or collection of books). Therefore, this project seeks to solve this problem using multiple machine learning and natural language processing methods. We repurpose a Goodreads dataset for spoiler detection to fit our needs and undergo BERT sentiment analysis and KNN/SVD collaborative filtering to give recommendations.

Dataset

We made two copies of the data. The first set only contains the book id and the review text from a user for sentiment analysis. The other set only contains the book id, user id, and rating values, which is used for collaborative filtering. The dataset has over 1.3 million data rows. For sentiment analysis, this would take an extremely long time to train and be resource intensive, so we stratified sampled the dataset to 10% of its original size, maintaining the relative frequency of the rating score, as seen in figure below. We ensure that reviews on the same book all occur either in the test set or in the training/validation set, not both, using GroupShuffleSplit. We then tokenized the data using the default BERT-Tiny tokenizer after preprocessing the review text to prepare for training. The other dataset for collaborative filtering was pivoted to make a matrix of the book id vs. user id, with the values being the ratings, filling the missing values with 0.



Methods

BERT:
Fine-tuning on a BERT model was used for sentiment analysis. Because BERT is a huge model, we opted to use the smallest of them, BERT-Tiny. We trained this BERT-Tiny model on the review text dataset to predict the user's rating score.

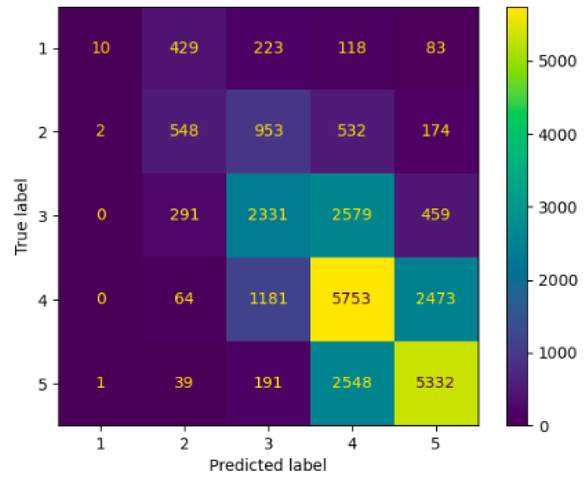
SVD:
SVD stands for Singular Value Decomposition and is a linear algebra method to decompose a matrix into three other matrices. Applying this to the sparse matrix dataset we made earlier allows us to retrieve a list of "latent factors", which is a list of dimensionality-reduced vectors of each book and user.

KNN:
KNN stands for K-Nearest Neighbors. Using a predefined number N=10, we use the Euclidean distance to find the N-nearest datapoints to some datapoint in vector space, where our vector space is the book and user id (and the rating score).

Predictions:
We calculate an average rating score from the 10 nearest neighbors and weight it at 0.5x. We then add the predictions from SVD using a dot product between the user latent factor and the book latent factor vectors, weighted at 1.5x. We then tabulate this scores for all books and return the highest scoring books.

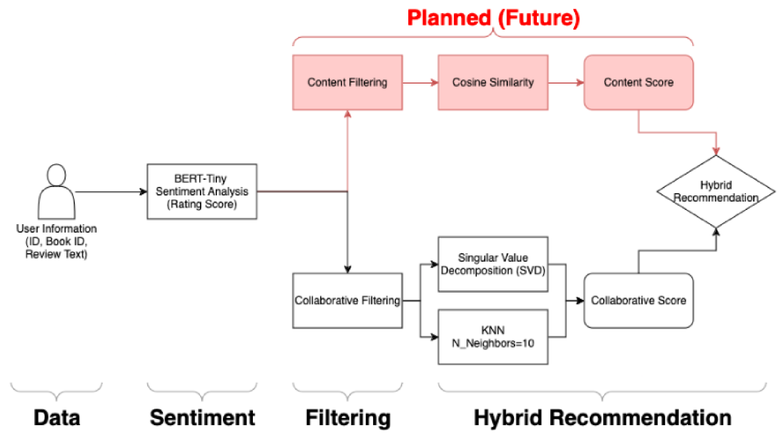
Results

We trained the BERT-Tiny model monitoring accuracy and minimizing the sparse categorical crossentropy loss. Training and validation accuracy ended at 55%, with a training accuracy of 53%. This means our model makes good generalizations. Although accuracy could end up being higher, 53% is sufficient for our needs, as it is better than random guessing, and we can see in the confusion matrix that the deviations from the guesses don't commonly stray further than 1.



Discussion

Through this project, we were able to create a pipeline where a user could feed in their information, book id, and review text. The data would automatically process the sentiment using the BERT-Tiny model yielding a predicted rating score. This rating score would be used in the KNN collaborative filtering value. The SVD collaborative filtering value would be collected as well and then presented to the user.



The figure above graphically shows the pipeline we have created. There is also some parts of the pipeline in red which are planned but not yet finished in this project.

Conclusions and Future Work

In conclusion, this project created a pipeline to make book recommendations to a user, given their user id, book id, and the review of the book. We utilized BERT-Tiny sentiment analysis, KNN, and SVD collaborative filtering to do so, creating a custom weighting system for recommending books.

The advantage of this project is its scalability. There are three main future work ideas that could be implemented:

- Content Filtering: Add to hybrid recommendation system by using measures like cosine similarity on book descriptions to find books with similar descriptions.
- UI: Create a user interface like a website or app that the user could interact with. As of right now, the user would need to pass in the information into python directly.
- More BERT: Try sentiment analysis on a larger BERT model like DistilBERT or try training on a larger training/testing set to improve accuracy.

References: The paper by Lee Choo Hui was the main paper used in the creation of this project. Other references are provided in the report.
Hui, Lee Choo, et al. "Sentiment Analysis and Innovative Recommender System: Enhancing Goodreads Book Discovery Using Hybrid Collaborative and Content Based Filtering." Lecture Notes on Data Engineering and Communications Technologies, 1 Jan. 2024, pp. 97–111, https://doi.org/10.1007/978-3-031-59707-7 9. Accessed 16 Nov. 2024.