STA521 Final Project

Bohao Yang
December 8, 2024

1 Introduction

The objective is to optimize Amazon's book recommender system to drive higher revenue by delivering personalized and accurate recommendations. Leveraging Amazon's known data on user purchases, examination history, and basic demographic information, this proposal addresses the limitations of traditional recommender systems like collaborative filtering and content-based filtering, which fail to capture the complex, sequential, and nuanced preferences of users[3]. To overcome these challenges and improve Amazon's books recommender system, the proposed hybrid solution integrates the following techniques:

- Sapling Similarity Collaborative Filtering (SSCF): An advanced developed recently that captures static user preferences through interaction similarity scoring.[2]
- LSTM-RNN for Sentiment Analysis: Analyzes user reviews to uncover nuanced patterns in text.[4]
- Demographic Analysis: Improves personalization using users' attributes such as age, location, and language preferences.

This hybrid system leverages the strengths of collaborative filtering, sentiment analysis, and demographic features to enhance recommendation accuracy, improve user engagement, and ultimately boost revenue.

2 Assumptions and Data Exploration

2.1 Assumptions

Amazon can have access to:
1. User Interaction Data: Purchase and examination history.
2. Behavior data: timestamps, frequency of interactions.
3. Book Metadata: Titles, genres, authors, pricing, and availability.
4. Review Data: Reviews from users, used for sentiment extraction.
5. Demographic Data: Age group, location (e.g., city or country), and language preferences.

2.2 Data Exploratory Analysis:

The data will be analyzed to uncover key patterns:
1. User Preferences: Identify purchase and rating trends across age groups and regions.
2. Sentiment Trends: Analyze positive, neutral, and negative sentiment patterns in reviews.
3. Regional Preferences: Examine how location influences preferences for book genres or authors.

3 Model Analysis and Comparisons

3.1 Collaborative Filtering (CF)

Collaborative filtering is a widely used model for traditional book recommender systems. Assuming users' previous purchases and examination history are known, the two common memory-based CF models are:[3]

- User-User Collaborative Filtering: captures similarity between users based on shared interactions.

- Item-Item Collaborative Filtering: captures similarity between items based on user ratings.

One of the most effective recent advancements is Sapling Similarity Collaborative Filtering (SSCF), a tree-based model developed in 2022, which effectively captures complex relationships between users and items.[2] Comparison experiments on the Amazon-Book Dataset show SSCF outperforms most of recent CF models, achieving higher scores in nDCG@20[1] (0.0647) and recall@20[2] (0.0773). The whole visualization can be seen in Figure 1. [6]

3.2 Implementation of SSCF [2]

1. Data preparation: Input a table of user-item interactions where rows represent users, columns represent items and values are binary(`1` if a user interacted with an item, `0` otherwise)
2. Construct the Bipartite Graph: Treat rows as users and columns as items and represent interactions as a binary matrix (`users` × `items`).
3. Calculate Basic Statistics: For each user, count the number of items they interacted with, and for each item, count the number of users who interacted with it.
4. Measure User Similarity:
① For two users, compute how often they interacted with the same items.
② Adjust similarity based on items both users interacted with (positive correlation) and items only one user interacted with (negative correlation).

3.3 Sentiment Analysis

Sentiment analysis started with pre-processing users' reviews, including converting them into lowercase, removing stop words, special character and punctuation, tokenization, lemmatization, and part of speech tagging. Then the sentiment analysis can be done through different models, the most popular ones are: [3]

- Traditional Machine Learning: Logistic Regression or Support Vector Machine with n-grams, bag-of-words or TF-IDF, etc.

- Convolutional Neural Network (CNN): Uses convolutional layers or modified CNN[5] to extract sentiment features.

- Recurrent Neural Network (RNN): Sequential modeling of reviews to capture complex sentiment patterns, such as LSTMs.

Studies indicate that LSTM-RNN performs well for sentiment analysis due to its ability to retain essential information and discard irrelevant keywords.[4]

3.4 Hybrid Architecture of Sentiment Analysis and Collaborative Filtering

Even though traditional recommender systems use users' past behavior and preferences through collaborative filtering is useful, it cannot capture the emotional tone of reviews. Thus many researchers incorporate

[1]Normalized Discounted Cumulative Gain (nDCG) measures ranking quality by comparing a given ranking to an ideal one where all relevant items are at the top. nDCG at K is the ratio of Discounted Cumulative Gain (DCG) to the ideal DCG, with K indicating the cutoff point for relevant items.

[2]Recall at K measures the proportion of correctly identified relevant items in the top K recommendations out of the total number of relevant items in the dataset.

collaborative filtering and sentiment analysis as an enhanced recommendation engine to overcome this drawback and give users more personalized recommendations[1]. Recent studies have given us insights into effective hybrid models.

One study uses the published Amazon book review dataset, which can be accessed through http://jmcauley.ucsd.edu/data/amazon/ and compares different hybridizations of collaborative filtering and machine learning-based sentiment analysis[4]. The results are shown in Figure 2 and Figure 3, indicating that a hybridization of collaborative filtering with a combination of Lexicon-based and deep learning-based sentiment analysis outperforms in accuracy(80.95%), precision(7.03%), and F1-score(12.16%) while comparing with other hybridized techniques.

The Lexicon-based sentiment analysis employs the SentiWordNet dictionary, which contains synsets and scores for positivity, negativity, and subjectivity of words. For Deep Learning-based sentiment analysis, the Long Short-Term Memory (LSTM) technique of Recurrent Neural Network (RNN) is employed. LSTM effectively learns sentiments by retaining essential keywords and discarding less relevant ones as new words are encountered. RNNs are highly effective in handling sequential data, as they are specifically designed to understand the inherent order and context within such data. By incorporating a memory mechanism, RNNs capture information from preceding steps in a sequence, influencing the generation of subsequent outputs. This ability makes them particularly well-suited for sentiment analysis tasks.

3.5 Incorporating Demographic Analysis

Demographic information like age, location, job, and gender can be used to enhance the recommendations by tailoring suggestions to different user groups based on their likely interests, assuming that people within similar demographic categories might share similar reading preferences.

These demographic information can be represented as dense vectors embeddings, and can adjust the similarities in collaborative filtering (here we assume the most effective one is SSCF). One potential similarity adjustment can be done by weighting similarity scores in SSCF and demographic similarities:

$$\text{Hybrid Similarity} = \alpha \cdot \text{SSCF Similarity} + (1 - \alpha) \cdot \text{Demographic Similarity}$$

4 Proposed Hybrid Model

Based on the analysis and comparison of previous and recent recommender models, a potential enhancement for Amazon's book recommender system can be outlined as follows. Assume Amazon has the previous purchase and examination history of all of its users, and basic demographic information about each user are known as well.

4.1 Model Components

- Sapling Similarity Collaborative Filtering (SSCF): Generates static similarity scores enhanced with demographic similarity. Scikit-learn and XGBoost can be used.[2]
- LSTM-RNN: Provides sentiment scores based on user reviews and demographic features. TensorFlow or PyTorch can be used. [4]
- Demographic Integration: Refines recommendations using user-specific attributes. Use pandas for preprocessing and embedding layers for dense feature representations.

4.2 Steps of Complete Implementation

1. Preprocess user and item interaction, users' reviews, and demographic information data.
2. Train SSCF to compute similarity scores with demographic adjustments.
3. Train LSTM-RNN with concatenated review and demographic embeddings.
4. Fuse the outputs and evaluate the hybrid system on demographic-specific metrics.
5. Output: Top-N recommendations personalized by interactions, sentiment, and demographics.

5 Further Recommendations

5.1 System Improvements

1. New Users Recommendations: Use demographic and metadata features for new users to give personalized cold-start recommendations.
2. Real-Time Personalization: Utilize real-time interactions and reviews in the recommender system to dynamically update recommendations for users.
3. Incorporate Large Language Models (LLMs): The emergence of LLMs, such as ChatGPT and GPT-4, has transformed Natural Language Processing and Artificial Intelligence with their exceptional language understanding, generation, generalization, and reasoning capabilities. Leveraging these strengths, LLMs may have their ground in the improvement of recommender systems in the future. [7]

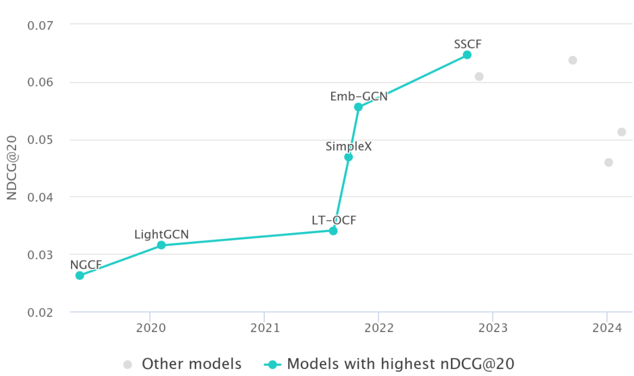
5.2 Marketing Strategies

Promote books that align with regional and cultural trends and design campaigns targeting specific demographics, integrating user preferences and behaviors to improve personalization.

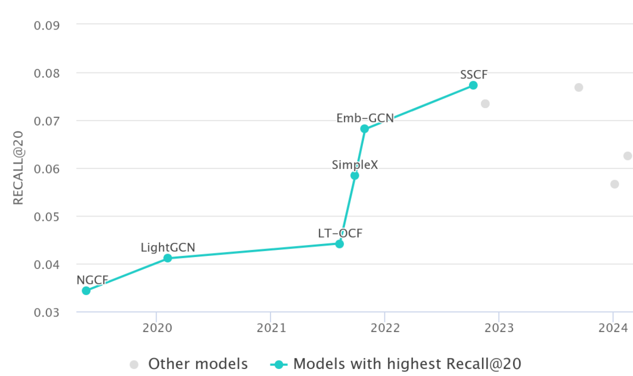
5.3 Revenue Strategies

Amazon can focus on recommending high-margin genres and books and optimize books with positive sentiment trends in user reviews.

6 Visulizations and tables



(a) nDCG@20



(b) recall@20

Figure 1: Performance metrics of recommendation systems on Amazon-Book dataset[6]

Hybridized Techniques*	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
CF + LR-NG-SA	68.74	5.18	55.96	9.49
CF + LR-WC-SA	68.34	5.17	56.53	9.47
CF + LR-TF-SA	56.15	4.16	63.49	7.82
CF + SVM-NG-SA	32.58	3.15	74	6.04
CF + SVM-WC-SA	27.13	3.04	77.41	5.86
CF + SVM -TF-SA	32.54	3.15	74.28	6.06
CF + LDA-NG-SA	70.5	6.1	63.06	11.13
CF + LDA-WC-SA	49.55	4.24	75.28	8.04
CF + LDA-TF-SA	67.17	5.66	65.19	10.42
Proposed HSBRS (Lexicon + DL + CF)	80.95	7.03	45.02	12.16

* CF+LR-NG-SA (Collaborative Filtering + Logistic Regression using n-gram feature vectorization-based sentiment analysis)
CF+LR-WC-SA (Collaborative Filtering + Logistic Regression using word count feature vectorization-based sentiment analysis)
CF+LR-TF-SA (Collaborative Filtering + Logistic Regression using TFIDF feature vectorization-based sentiment analysis)
CF+SVM-NG-SA (Collaborative Filtering + Support Vector Machine using n-gram feature vectorization-based sentiment analysis)
CF+ SVM-WC-SA (Collaborative Filtering + Support Vector Machine using word count feature vectorization-based sentiment analysis)
CF+ SVM-TF-SA (Collaborative Filtering + Support Vector Machine using TFIDF feature vectorization-based sentiment analysis)
CF+LDA-NG-SA (Collaborative Filtering + linear discriminant analysis using n-gram feature vectorization-based sentiment analysis)
CF+ LDA -WC-SA (Collaborative Filtering linear discriminant analysis using word count feature vectorization-based sentiment analysis)
CF+ LDA -TF-SA (Collaborative Filtering + linear discriminant analysis using TFIDF feature vectorization-based sentiment analysis)

Figure 2: Comparison of Hybridization of CF and ML based sentiment analyis with HSBRS Technique[4]

Three Hybrid Techniques	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Lexicon+LogisticRegression+CF	66.55%	4.45%	50.99%	8.20%
Lexicon+LinearDiscriminantAnalysis+CF	77.34%	6.40%	49.43%	11.33%
Lexicon+SupportVectorMachine+CF	79.84%	6.18%	41.47%	10.76%
Proposed HSBRS (Lexicon + DL + CF)	80.95%	7.03%	45.02%	12.16%

Figure 3: Comparison of Deep Learning and Machine Learning Techniques in Hybridization[4]

References

[1] Fatemeh Abbasi and Ameneh Khadivar. Collaborative filtering recommendation system through sentiment analysis. *Turkish Journal of Computer and Mathematics Education*, 12(14):1843–1853, 2021.

[2] Giambattista Albora, Lavinia Rossi Mori, and Andrea Zaccaria. Sapling similarity: A performing and interpretable memory-based tool for recommendation. *Knowledge-Based Systems*, 275:110659, 2023.

[3] P Devika and A Milton. Book recommendation system: reviewing different techniques and approaches. *International Journal on Digital Libraries*, pages 1–22, 2024.

[4] A Kumar and S Chawla. Hsbrs: Hybrid sentiment-based collaborative architecture for book recommendation system. *Indian Journal of Science and Technology*, 17(11):1003–1015, 2024.

[5] Yarasu Madhavi Latha and B Srinivasa Rao. Amazon product recommendation system based on a modified convolutional neural network. *ETRI Journal*, 2024.

[6] Papers with Code. Recommendation systems on amazon book dataset. https://paperswithcode.com/sota/recommendation-systems-on-amazon-book, 2024. Accessed: 2024-12-05.

[7] Zihuai Zhao, Wenqi Fan, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, et al. Recommender systems in the era of large language models (llms). *arXiv preprint arXiv:2307.02046*, 2023.