This is an Individual Project. Team member: Xingjian Liu (UM-ID 76386327)

Topic

Enhancing E-commerce Personalization: A Hybrid Recommender System for Amazon Fashion Products

Objective

In the competitive landscape of E-commerce, personalized shopping recommendations are vital for customer experience. Specifically, product recommendations tailored to a customer's individual preferences could not only lead to higher purchase conversions that contribute to the platform GMV, but also boost customer satisfaction and loyalty, fostering a healthy growth for an E-commerce platform in terms of monetization and brand reputation.

Recommender systems are used by E-commerce to suggest tailored purchase recommendations. Take Amazon, the largest worldwide E-commerce platform, as an example. Amazon went into the apparel business in 2002, serving as an online fashion retailer. At first, the apparel business faced the challenge that people preferred to try on items offline than to purchase online. But as of 2019, Amazon became the nation's top fashion retailer, beating out Walmart and Target. The primary advantage Amazon has over the other retailing competitors is the vast amount of data it possesses and its strong recommender system, which leverages advanced data mining algorithms to derive insights into the relationship between customers and products.

Despite the power of recommendation algorithm, developing a recommender system for a large-scale E-commerce platform faces multiple challenges:

1. Cold Start. Making accurate recommendations for new users is difficult due to lack of knowledge about them.
2. Data Sparsity. The vast number of users and items compared to the recorded user-item interactions hinders the learning ability of data mining algorithms
3. Scalability. As the number of users and item grows, the system should maintain performance without excessive computational resources.

The primary objective of this project is to develop a recommender system for E-commerce fashion businesses that aims to mitigate the aforementioned challenges.

Data Source

The data to use for constructing our recommender system comes from Amazon's customer reviews and product metadata restricted to the Fashion category from 2010 to 2018, which is constructed by the McAuley's Lab at UCSD[1]. The data comprises two separate datasets ---- the review data and product metadata:

- Review Data: 825795 records, contains 12 fields including `reviewerID` as user id, `asin`

[1] He, R., & McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. *Proceedings of the 25th International Conference on World Wide Web (WWW)*, 507–517. https://doi.org/10.1145/2872427.2883037

as product id, `overall` as rating (from 0 to 5 in integer type), ` unixReviewTime` as the timestamp for each review, etc. This dataset provides user-item interaction data essential for collaborative filtering.

- Product Metadata: 186637 records, contains 16 fields including `asin` as product id, `title` as product title, `description` as a sentence or two describing the product, etc.

Methodology Overview

We propose a Hybrid approach that integrates both collaborative filtering and content-based filtering to leverage their strengths and offset their weaknesses in terms of cold start, data sparsity, and scalability. Specifically, it would be a two-step sequential recommender system with the following structure:

- Step 1: Recall (Candidate Generation). This process aims to efficiently retrieve the top 100 candidate items for each user. Potentially, Matrix Factorization method would be applied to learn the latent user and item factors from the sparsity user-item interaction data, and Approximate Nearest Neighbors algorithms used to increase the efficiency of retrieval based on similarity in the latent space.
- Step 2: Ranking (Final Recommendation). This process aims to rank the top 100 candidates precisely based on item similarity in product metadata, and select the top 5 items with highest similarity scores. Potentially, content-based filtering with NLP-based feature engineering would be used to extract feature vectors of items, and similarity scores would be measured using specific similarity metrics.

For model training, the dataset would be split into training, validation, and test sets based on timestamp in order to ensure the model generalization and prevent information leakage.

For model evaluation, due to lack of online implicit interaction data (e.g., CTR, likes, saves), the evaluation focuses on offline metrics including rating prediction accuracy such as RMSE and recommendation accuracy such as Hit Rate.

Expected Deliverables and Timeline

Deliverables:

- A comprehensive technical report that details the algorithms including the mathematics behind and analyses of model performance as well as discussions of limitations.
- An accessible API endpoint allowing users to input a user id and get the top 5 product purchase recommendations.

Timeline:

- 11/01/2024 – 11/20/2024: Offline development and experimentation of the model
- 11/21/2024 – 12/04/2024: Online deployment of the data product
- 12/05/2024 – 12/08/2024: Preparation of report and video presentation