Personalised Recommendation System for Live Application

Introduction

The Journal application aims to deliver a personalized news feed to readers by recommending articles based on their preferences. To enhance user engagement and improve content discovery, we propose implementing Neural Collaborative Filtering (NCF) as the primary recommendation technique. This deep learning-based approach models complex user-article interactions through neural networks and leverages both behavioral data and semantic content information, such as category embeddings generated using language models.

Objective

The primary objective of this proposal is to implement a Neural Collaborative Filtering model using PyTorch, which recommends articles based on user reading history and interaction patterns. Additionally, we propose enriching this recommendation system with category-level semantic information using SentenceTransformer-based embeddings, thereby improving personalization, relevance, and overall recommendation quality.

Methodology

The recommendation system will be developed using the following approach:

Data Collection:

- User Interaction Data: We collect user behavior such as article clicks, reading time, likes, and shares.

- Article Metadata: We utilize article-level information including title, abstract, and category.

- Category Mapping: Articles are grouped into categories using a predefined mapping of category → list of news_ids.

Text Embedding and Semantic Enrichment:

To infuse semantic understanding into the model, we perform the following steps:

- Use the SentenceTransformer model (all-MiniLM-L6-v2) to generate category-level embeddings by:
    - Aggregating the titles and abstracts of up to five representative articles per category.
    - Concatenating their text content and encoding it into fixed-size vectors using the transformer model.
- These category embeddings are stored as tensor representations and are optionally used as auxiliary input in the recommendation model.

Model Architecture:

We build a Neural Collaborative Filtering model in PyTorch with the following structure:

- User Embedding Layer: Learns latent features of users.
- Item Embedding Layer: Learns latent features of articles (news items).
- Multi-Layer Perceptron (MLP): Takes the concatenated user-item embeddings and optionally the category embeddings to model the interaction.
- Output Layer: Predicts a relevance score between 0 and 1, representing the probability of user interest.

Training Process:

- Loss Function: Binary Cross-Entropy or Mean Squared Error, depending on interaction data type.
- Optimizer: Adam with scheduled learning rate decay.
- Regularization: Dropout and weight decay to prevent overfitting.
- Batching and Sampling: Efficient mini-batching with negative sampling to balance implicit feedback data.

Recommendation Generation:

- For each user, the trained model predicts scores for all unseen articles.
- Articles are ranked by predicted scores and filtered based on recency, popularity, and diversity.
- Recommendations are generated dynamically and updated periodically as user behavior evolves.

Cold Start Strategy Using RAG-inspired Category Embeddings

To address the cold start problem for new users with no historical data, we implement a Retrieval-Augmented Generation (RAG)-inspired strategy using SentenceTransformer embeddings. Specifically, we create semantic representations of each category by aggregating the titles and abstracts of up to five representative news articles. These combined texts are embedded using the all-MiniLM-L6-v2 model to form fixed-size category embeddings. When a new user provides a prompt (e.g., keywords or interests), it is embedded in the same semantic space and compared to category vectors using cosine similarity. The system then recommends the top-matching categories, enabling accurate and personalized onboarding for first-time users with no behavioral history.

Expected Benefits

- **Personalized Experiences:** Captures both user behavior and semantic content for more accurate recommendations.

- **Enhanced Discovery:** Surfaces relevant content from diverse topics and categories.

- **Scalability:** Neural models adapt well to dynamic datasets and perform efficiently at scale.

- **Content-Aware Recommendations:** By using sentence-transformer embeddings, recommendations benefit from semantic understanding of articles and categories.

  For RAG Model - Cold start problem

- **Solves Cold Start Problem**: Enables personalized recommendations for new users without requiring historical interaction data.

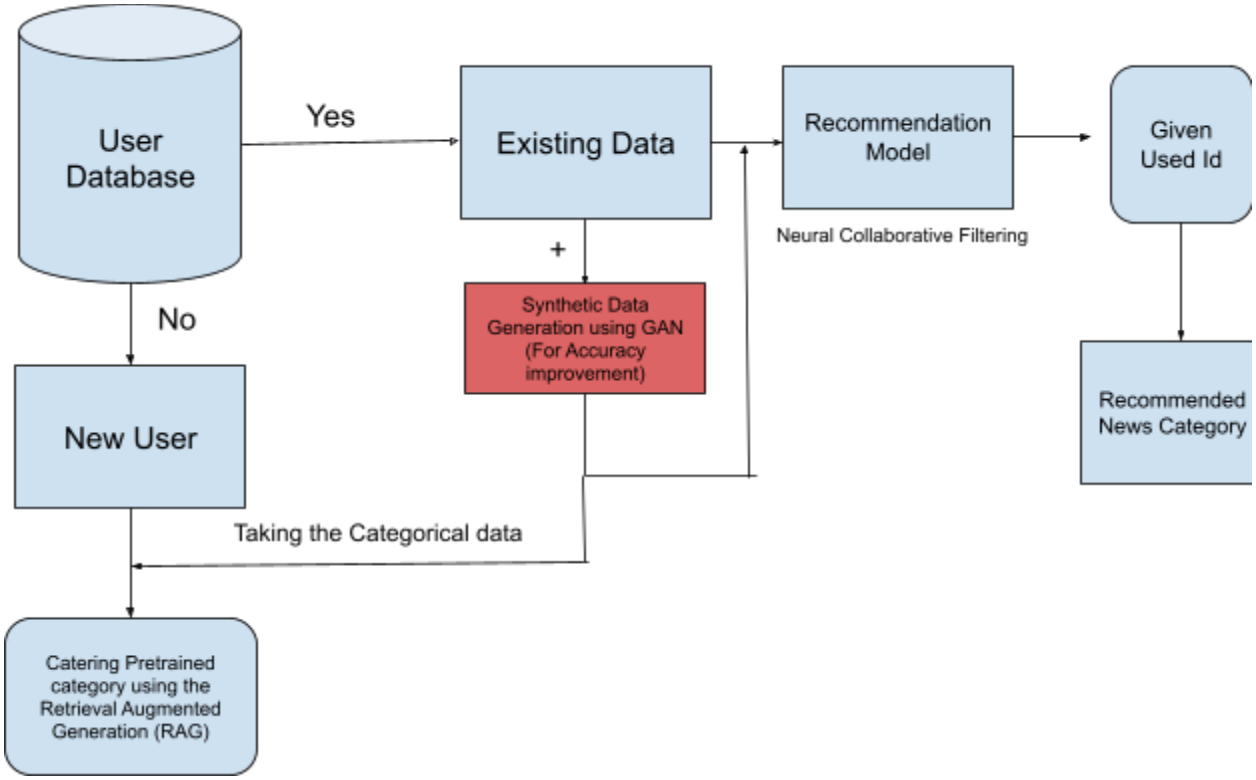
- **Fast & Scalable**: Embedding and similarity search is computationally efficient and can scale to thousands of categories or articles.

- **Context-Aware Recommendations**: Leverages pre-trained transformer models to understand user intent from short prompts or keywords.

- **Improved User Onboarding**: Helps first-time users quickly discover relevant content, boosting early engagement and retention.

- **Easy Integration**: Can be integrated alongside collaborative filtering models for a hybrid recommendation engine.

Schematic Pattern:



Conclusion

This proposal outlines the implementation of a Neural Collaborative Filtering model in PyTorch, enhanced with SentenceTransformer-based category embeddings, to elevate the recommendation capabilities of the Journal application. By integrating deep learning with semantic enrichment, we aim to deliver more relevant and personalized news feeds that enhance user satisfaction, retention, and engagement. Upon approval, we will begin model development and deploy a scalable pipeline for real-time recommendation generation.

Prepared by: Anand Ramaswamy Jayshree
Graduate Student, Master of Science in Information Science
Date: 04-21-2025