Recommendation System Approaches

Gaurab Pokharel

February 15, 2025

1 Introduction

This report explores multiple recommendation system approaches for a movie rating dataset. We begin with a random baseline and proceed to more sophisticated methods, including content-based filtering, collaborative filtering, a library-based LightFM model, and two hybrid neural models (HybridRecSys and NeuMFHybrid). Our objective is to compare these methods using metrics such as Precision@K, Recall@K, and NDCG@K.

2 Method

- **Random Baseline:** Provides a naive lower bound by recommending a fixed number of items at random from the set of unrated items.
- **Content-Based Filtering:** Constructs a user profile from content features (e.g., genre vectors) of movies the user has liked. Items are then ranked by their similarity (e.g. cosine) to this user profile.
- **Collaborative Filtering:** Relies solely on user–item interactions. We compute user–user similarity via cosine similarity on a user–item pivot table. Predictions or recommendations are made by aggregating ratings from similar users.
- **LightFM Model:** Uses the LightFM library, which supports hybrid (collaborative + content) approaches. We employ a ranking-oriented loss such as WARP and treat this model as a strong baseline to compare against more specialized neural methods.
- **HybridRecSys:** A neural hybrid model that combines learned embeddings for users and movies with a projection of content features (genre vectors). It can be trained either via MSE (for rating prediction) or BPR loss (for ranking).
- **NeuMFHybrid:** Another neural hybrid approach, merging a GMF (Generalized Matrix Factorization) branch with an MLP branch, and also integrating genre vectors. It supports both standard regression objectives (MSE) and pairwise ranking objectives (BPR).

3 Results

In the following section, we will present the performance metrics of each model. These metrics typically include:

- **Precision@K:** Fraction of the top-K recommendations that are relevant.
- **Recall@K:** Fraction of the user's relevant items that appear in the top-K recommendations.

- **NDCG@K:** A position-aware measure that assigns higher weight to relevant items near the top of the recommendation list.

We compare all the methods using the same dataset split and evaluation procedure to ensure a fair comparison.

Table 1 summarizes the performance of each method in terms of Precision@10, Recall@10, and NDCG@10. The HybridRecSys and NeuMFHybrid models were trained using a BPR objective for a limited number of epochs (omitted here); further training and hyperparameter tuning could improve performance.

Method	Precision@10	Recall@10	NDCG@10
Random	0.0012	0.0013	0.0019
Content-Based	0.0088	0.0080	0.0106
Collaborative	0.1645	0.1580	0.2210
LightFM	0.0816	0.0687	0.0677
HybridRecSys	0.1325	0.8380	0.0116
NeuMFHybrid	0.1198	0.6661	0.0108

Table 1: Comparison of recommendation models using Precision@10, Recall@10, and NDCG@10.

4 Discussion

The results reveal distinct differences among the recommendation methods. The Random Baseline, as expected, performs very poorly across all metrics. The content-based filtering approach, which uses movie genres alone, achieves only marginal improvements over the random baseline.

In contrast, the collaborative filtering recommender yields a significant increase in both Precision and NDCG, demonstrating that user–item interactions capture valuable signals for recommendation. The LightFM model, which fuses collaborative and content-based information, achieves moderate performance; however, its metrics are lower than those of the pure collaborative approach.

The two hybrid neural models, HybridRecSys and NeuMFHybrid, were trained using Bayesian Personalized Ranking (BPR) loss. Although both models achieve relatively high recall, their precision and NDCG values remain lower. This suggests that while they retrieve many relevant items overall, these items are not well-ranked in the top positions. Moreover, given that both hybrid models were trained for only 10 epochs without extensive hyperparameter tuning (e.g., grid search), we expect that further training and systematic optimization could yield substantial improvements in ranking quality, particularly in terms of NDCG.