Movie Recommendation System Project Report

Comparative Analysis on MovieLens 20M Dataset

Yuri Croci

LinkedIn — GitHub

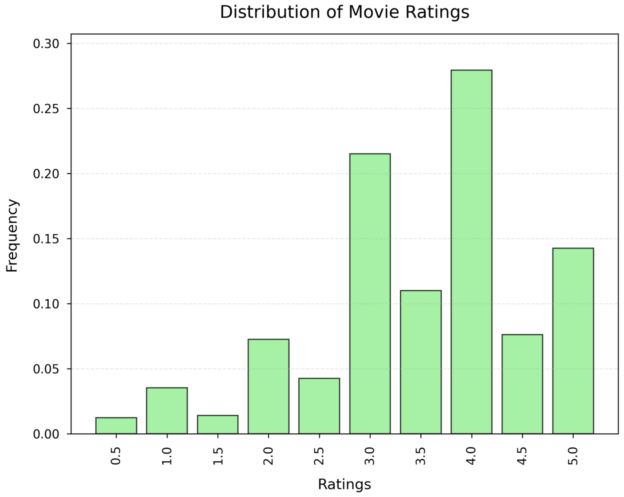
1 Introduction

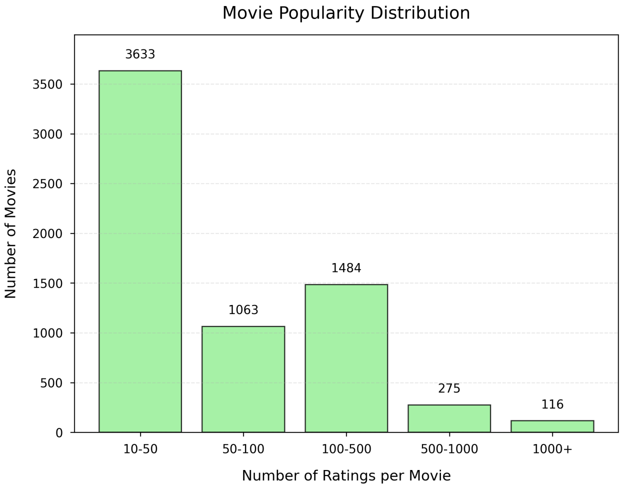
As data tracking increases, recommendation systems have become a key part of modern digital platforms. Netflix, Spotify, and Amazon use these systems daily to suggest content tailored to individual preferences. While users interact with these recommendations seamlessly, the underlying algorithms reveal distinct behaviors and trade-offs depending on available data and system design. This project provides a systematic comparison of four core recommendation approaches, along with a novel hybrid neural network architecture designed to address their individual limitations.

The goal of this project is to understand the trade-offs between different models and how they leverage the data available in the MovieLens 20M dataset, which contains user ratings and movie metadata (genre tags, relevance scores, etc.). This comparison employs a systematic evaluation process with comparative hyperparameter tuning and separate analysis of warm-start and cold-start scenarios, where the model has seen many or few training examples, respectively.

2 Dataset Overview

This project uses a random subset of the MovieLens 20M dataset, a well-established benchmark for recommendation system research. To make the dataset computationally manageable, only 6,000 randomly selected users were retained, along with movies that received at least 10 ratings. The resulting subset contains over 850,000 ratings across approximately 6,500 movies, along with rich metadata including movie genres and user-generated tag relevance scores. Following exploratory analysis and data cleaning, relevant features were created to support both collaborative and content-based recommendation methods. The dataset exhibits typical challenges such as high sparsity and a long-tail popularity distribution, where most movies receive few ratings, creating cold-start scenarios for less popular items.





3 Methodology

3.1 Approaches Evaluated

As a comparative study, four types of recommendation algorithms were systematically analyzed to identify their respective strengths and weaknesses in leveraging the available data. For each approach, a theoretical foundation was established, followed by implementation, hyperparameter tuning, and rigorous evaluation to enable fair comparison across methods.

- **Memory-Based Collaborative Filtering:** Generates predictions by computing similarity between users (User-User) or items (Item-Item) based on historical rating patterns, then aggregating ratings from nearest neighbors to predict unknown preferences.
- **Content-Based Filtering:** Recommends items based on similarity of movie features, relying solely on item metadata without considering user interaction patterns. A movie feature vector was engineered including one-hot encoded genres, dimensionality-reduced tag scores, Bayesian-adjusted ratings, and temporal metadata.
- **Model-Based Collaborative Filtering:** Employs Funk SVD matrix factorization to decompose the user-item rating matrix into latent embeddings, learning compressed representations that capture underlying preference patterns through iterative optimization.
- **Hybrid Neural Network:** Combines the model-based embeddings with content-based features through a neural network architecture, integrating collaborative and content signals to leverage the strengths of both approaches.

3.2 Evaluation Strategy

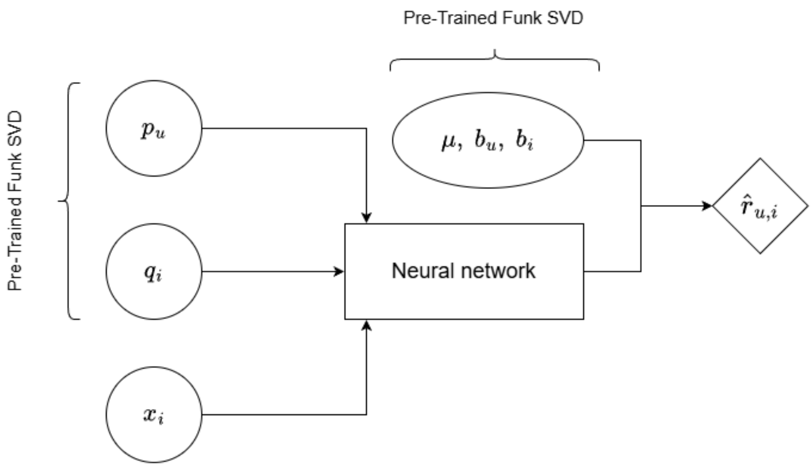
The evaluation follows a realistic implementation scenario where each user's most recent ratings are predicted. During the exploration of the algorithms, each model was optimized through hyperparameter tuning on a validation set and evaluated on a held-out test set using a 70/10/20 train-validation-test historical split. Test performance was measured both overall and separately for warm-start movies (frequently rated in training) and cold-start movies (fewer than 10 ratings in the training set), revealing how each algorithm handles varying levels of data availability. Additionally, computational efficiency was assessed by measuring training and prediction times across all approaches using the same workstation.

4 Hybrid Model Architecture

As the main contribution of this comparative study, I designed a hybrid neural network implemented in PyTorch that refines collaborative filtering predictions by integrating content features. Rather than training embeddings from scratch or using a simple weighted combination, the model leverages high-quality representations learned by pre-trained Funk SVD. The user and item embeddings are then combined with the movie feature vector used in the content-based approach and passed through a trainable feedforward network that learns non-linear refinements to the standard dot-product prediction. The network output is then combined with the frozen global mean and user and item bias terms to form the final rating prediction:

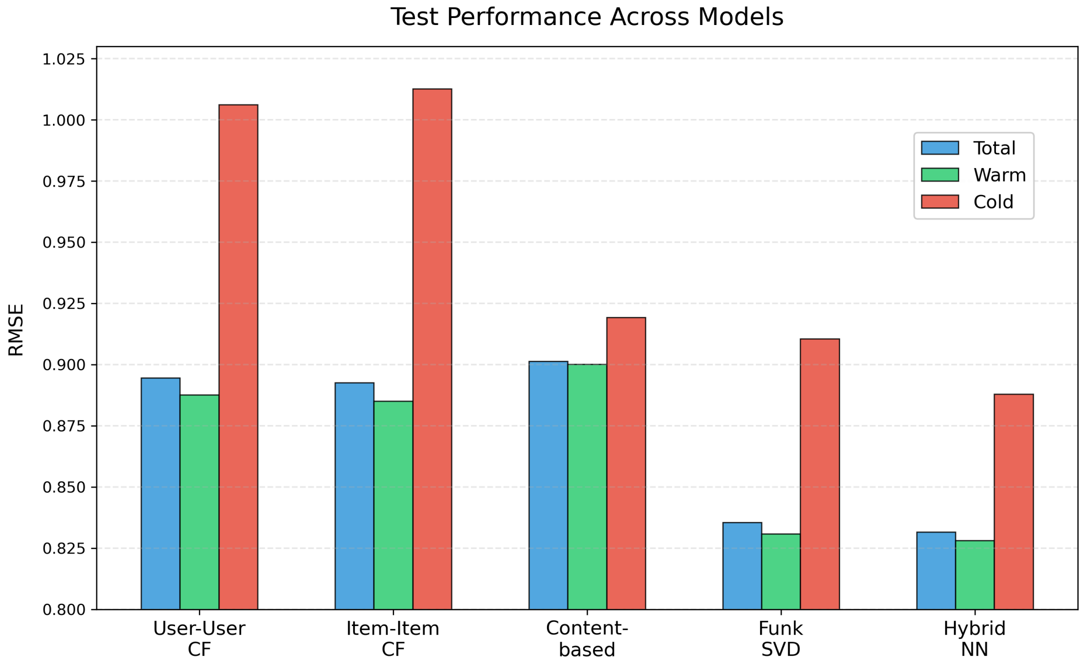
$$\hat{r}_{u,i} = \mu + b_u + b_i + f(p_u, q_i, x_i)$$

where $\hat{r}_{u,i}$ denotes the predicted rating given by user u to movie i. This design aims to preserve the strong performance of collaborative filtering on popular items while incorporating content signals to improve predictions in cold-start scenarios.



5 Results

The systematic comparison reveals distinct strengths and trade-offs across recommendation approaches. Memory-based collaborative filtering achieved moderate accuracy but faced significant computational scalability challenges due to the large similarity matrices required. Both User-User and Item-Item methods showed substantial performance degradation on cold-start items, with RMSE exceeding 1.0 in these scenarios. In contrast, content-based filtering proved quick and efficient, handling cold items reasonably well by relying on movie features rather than interaction history, demonstrating stable performance across different data availability conditions.



Funk SVD emerged as the strongest baseline, offering better scalability and improved performance across all metrics through efficient matrix factorization. With a test RMSE of 0.8355 overall and 0.8309 on warm items, it demonstrated the power of learning latent representations from interaction patterns. However, cold-start performance remained a challenge with an RMSE of 0.9105 on rarely-rated movies.

Building upon this foundation, the hybrid neural network achieved the best overall test RMSE of 0.8317, marginally improving on warm items (0.8282) while significantly addressing the cold-start problem. Most notably, the hybrid approach reduced cold-start error by approximately 2.5% compared to Funk SVD alone (0.9105 → 0.8880), demonstrating how content signals can refine collaborative predictions when interaction data is sparse. This improvement validates the design choice of integrating content features with pre-trained collaborative embeddings rather than relying on either approach in isolation.

6 Conclusion

This comparative study systematically evaluated recommendation algorithms across content-based, memory-based, and model-based collaborative filtering paradigms, identifying distinct trade-offs in how each leverages available data. The hybrid neural network successfully addresses these limitations by combining pre-trained collaborative filtering embeddings with content features, achieving the best overall performance and reducing cold-start error by approximately 2.5% compared to matrix factorization alone. The results demonstrate that integrating complementary information sources through flexible neural architectures enables more robust predictions across varying data availability scenarios. Future work could extend this approach by incorporating user demographic or behavioral features to further exploit available information and improve recommendation quality on datasets with richer user profiles.