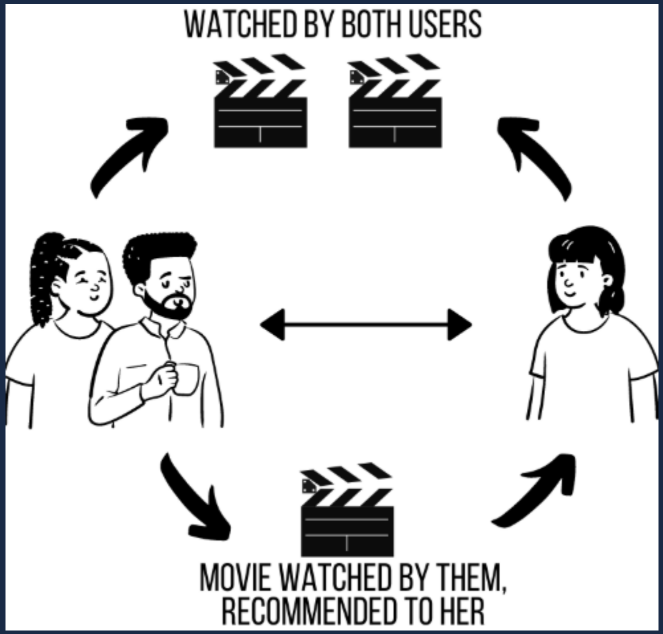
Movie Recommendation System

Machine Learning: Collaborative Filtering



Value:

Reduce choice overload • Increase engagement • Support retention

Author: Rahaman Yusuf

Business Problem

Stakeholder

Movie streaming platform product & personalization team

Problem

Users face choice overload. If they can't quickly find movies they'll enjoy, engagement drops and churn risk increases.

Goal

Predict how a user would rate unseen movies and return Top-5 personalized recommendations.

How it would be used

- “Recommended for You” row on homepage
- Email/push recommendation campaigns
- Content discovery and retention initiatives

Data Overview

Dataset

MovieLens "latest small" (GroupLens)
Explicit ratings on a 0.5–5.0 scale

Key facts

- 100,836 ratings
- 610 users
- 9,742 movies
- 3,683 tag applications

Why this matters

The user–movie matrix is ~98.3% sparse. Sparsity makes similarity-based methods harder and motivates matrix factorization.

Sparsity

98.3%

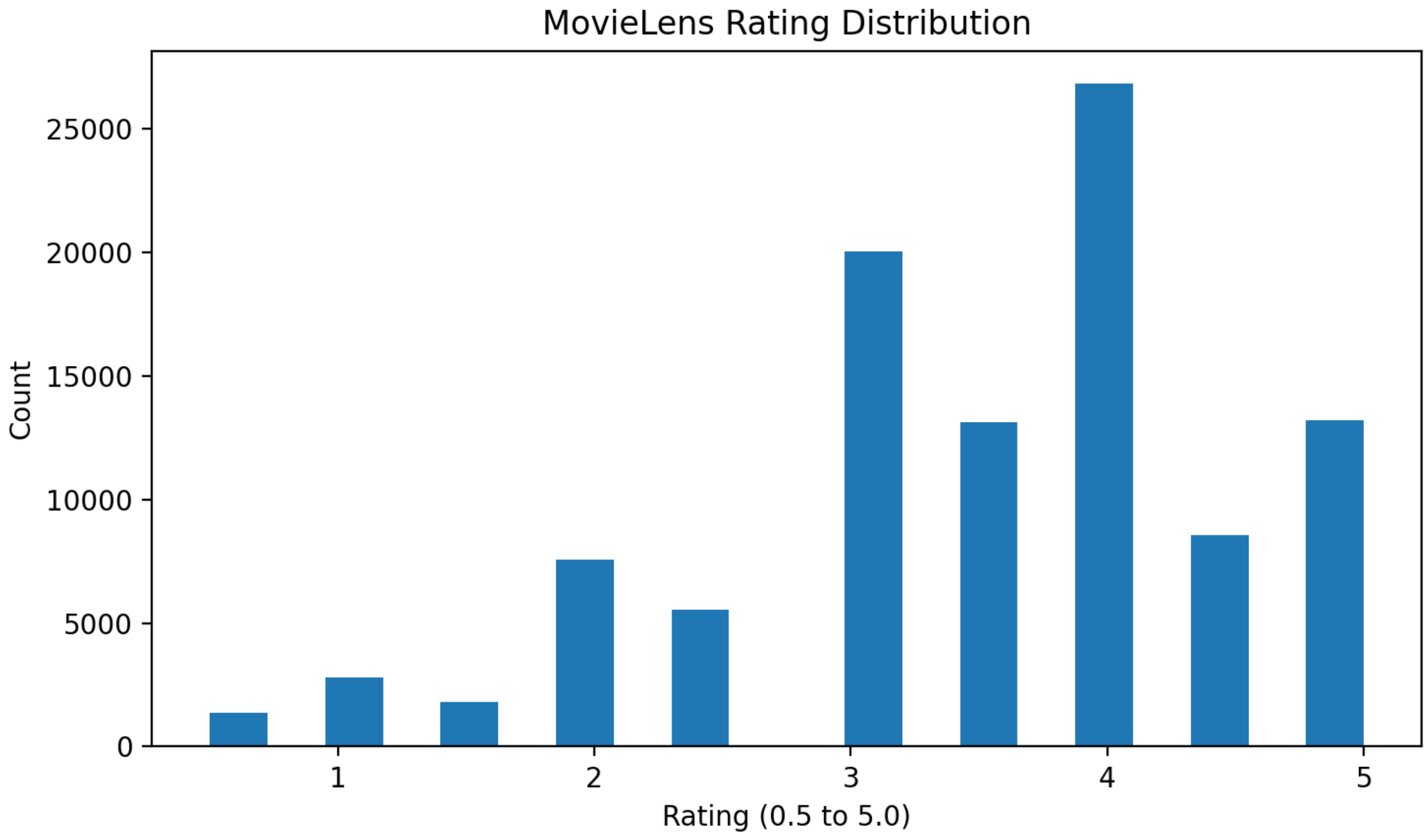
Only ~1.7% of user× movie pairs have ratings

Limitations

- Cold start for new users/movies
- No demographics
- Offline error ≠ business KPIs

Exploratory Data Analysis

Rating distribution (explicit feedback)



Observation: ratings cluster around 3–4 stars (selection bias: users tend to rate movies they chose to watch).

Modeling Approach

Iterative modeling: start simple → add complexity → tune → evaluate on holdout test

BaselineOnly
Global + user + movie biases

→

kNNBaseline
Item-item similarity

→

SVD (MF)
Latent factors

→

Tuned SVD
GridSearchCV (CV)

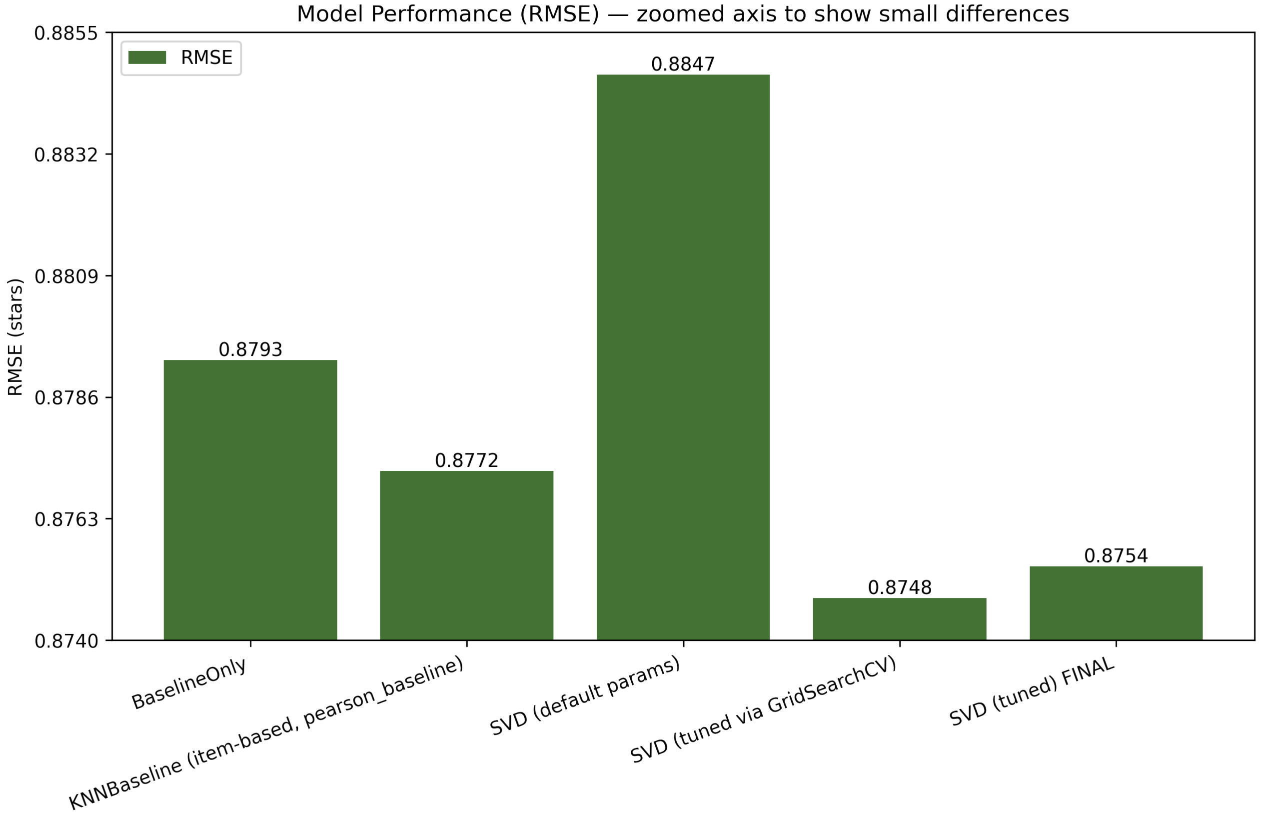
Why factorization helps: learns global structure from sparse ratings

Evaluation plan

- 80/20 split (holdout test)
- 3-fold CV on train for model selection
- Metrics: RMSE + MAE (error in "stars")

Results

Model performance (lower is better)



Key takeaways

- Baseline is strong (bias effects explain much of rating behavior)
- Tuning makes SVD competitive and stable on holdout (CV ≈ Test)
- RMSE ≈ 0.88 ⇒ typical error ~0.9 stars on a 0.5–5 scale

Note: RMSE (zoomed) y-axes so small differences are easier to see. Lower is better.

Demo: Existing User Recommendations

User 599: what they loved

User 599: Top-Rated Movies (from historical ratings)

title	rating
Kill Bill: Vol. 1 (2003)	5.0
Magnolia (1999)	5.0
His Girl Friday (1940)	5.0
Ghost in the Shell (Kôkaku kidôtai) (1995)	5.0
Lost in Translation (2003)	5.0

Top-5 recommendations (example)

Example Top-5 Recommendations (Existing User ID 509)

movieId	title	predicted_rating
1104	Streetcar Named Desire, A (1951)	4.045198
318	Shawshank Redemption, The (1994)	4.038804
1217	Ran (1985)	4.037783
3468	Hustler, The (1961)	4.031782
898	Philadelphia Story, The (1940)	4.014178

Note: recommendations table shows an example user demo output from the notebook run.

Cold Start: New User Strategy

Problem: collaborative filtering cannot personalize for a user with 0 ratings.
Solution: ask for 5–10 initial ratings (popular movies) → refit/update user profile → recommend Top-5.
Fallback: if 0 ratings are provided, recommend popular + highly-rated movies until feedback arrives.

New user Top-5 (after 5 seed ratings)

Example Top-5 Recommendations (New User ID 611 after 5 seed ratings)

movieId	title	predicted_rating
1217	Ran (1985)	4.623696
1104	Streetcar Named Desire, A (1951)	4.576792
3468	Hustler, The (1961)	4.573437
1223	Grand Day Out with Wallace and Gromit, A	4.569716
905	It Happened One Night (1934)	4.567882

True cold start fallback (0 ratings)

Fallback Top-5 (True Cold Start: 0 ratings)

movieId	title	avg_rating
318	Shawshank Redemption, The (1994)	4.429022
260	Star Wars: Episode IV - A New Hope (1977	4.231076
527	Schindler's List (1993)	4.225
296	Pulp Fiction (1994)	4.197068
2571	Matrix, The (1999)	4.192446

Limitations & Next Steps

Limitations

- Sparse data limits personalization for low-activity users
- No demographics or rich item metadata
- RMSE/MAE are offline metrics; production success is measured by CTR, watch-time, retention

Next steps (production-minded)

- Hybrid recommender: combine CF with genres/tags for better cold start
- Add ranking metrics (Precision@K / Recall@K / NDCG@K)
- Online A/B test to validate business lift
- Incorporate implicit feedback (clicks, watch time) for scale