Recommender Systems: Movies

Timon Weidemann

esade

Table of Contents

1 Project Goal & Context

Design and implement a prototype of a recommender system

Dataset

We will base our recommendations on the MovieLens 100K Dataset with IMDb rated movies

- ratings.csv
- tags.csv
- links.csv
- movies.csv

Recommenders

We will develop and implement all recommender systems seen in class to choose the best-suited

1. Non-personalised
2. Collab. Filtering
3. Content Filtering
4. Matrix Factorisation

Evaluation

We will evaluate all recommender systems developed to take a decision on which is best-suited

1. Precision
2. Recall
3. MRR
4. RMSE

2 Technical Approach

links.csv

movies.csv

ratings.csv

tags.csv

Clean and merge different data sets

P1: Data

1. Create Non-Personalized Functions
2. Create Collaborative Filtering Functions
3. Create Content-based Filtering Functions
4. Create Matrix-Factorisation Functions
5. Create Hybrid Functions

Reevaluate Algorithms

P2: Model

Create Streamlit Dashboard

Reiterate functionalities

Launch Streamlit App in Cloud

Create Business Presentation

P3: Prototype

Repository: https://github.com/timonweide/Recommender_System_Movielens

3 Algorithm Exploration – Overview

Algorithm	Type	Description
Top-N	Count Likes	Returns recommendations based on non-personalized frequency of appearance or frequency of > 4-star ratings
Average Rating	Standard Weighted Normalized	Returns recommendations based on non-personalized ratings, weighted by number of user votes or normalized by individual user's scale
Collaborative Filtering	User-Pearson User-Cosine Item-Pearson Item-Cosine	Finds nearest neighbours (users or items) via Pearson or cosine similarity and predicts missing ratings from a similarity-weighted average of their scores
Content-Based	Cosine	Matches user preferences to genre TF-IDF profiles using cosine
Matrix-Factorisation	SVD NMF ALS	Factorises the user-item ratings matrix into low-rank user x item latent vectors and predicts unseen scores as the dot-product of those learned vectors (e.g., SVD, NMF, ALS).
Hybrid	Recommended	Blends personalized algorithms with popularity-based algorithms

Non-person.

Personalized

3 Algorithm Exploration – Deep-Dive

A custom hybrid recommender was designed for ultimate user control

User input:	Cold Model	Warm Model	Min. Interactions	Alpha
	Choice of non-personalized models (e.g. average rating)	Choice of personalized models (e.g. content-based)	Choice of the interaction threshold required to consider warm	Choice of what percentage the warm model score is used in blend

If

Min. Interactions is not exceeded → Scored by: Cold Model

Min. Interactions is exceeded → Scored by: Alpha * Warm Model + (1 - Alpha) * Cold Model

4 Model Evaluation – Overview

Algorithm	Type	Precision	Recall	MRR	RMSE
Top-N	Count	0.0357	0.1081	0.1811	160.6179
	Likes	0.0429	0.1111	0.1994	119.8845
Average Rating	Standard	0.0286	0.0985	0.0400	**1.6000**
	Weighted	0.0643	0.1233	0.2024	**0.3744**
	Normalized	0.0286	0.0985	0.0400	**1.6000**
Collaborative Filtering	User-Pearson	0.0571	0.1109	0.2381	1.6648
	User-Cosine	**0.1214**	**0.1860**	**0.4054**	2.4447
	Item-Pearson	0.0214	0.0896	0.0312	1.7690
	Item-Cosine	0.0643	0.1262	0.1821	2.6268
Content-Based	Cosine	0.0071	0.0179	0.0357	2.1500
Matrix-Factorisation	SVD	0.0357	0.0353	0.1786	2.0380
	NMF	0.0429	0.0387	0.1865	3.1826
	ALS	0.0286	0.0238	0.0476	3.7432
Hybrid	Recommended	**0.1000**	**0.1765**	**0.4355**	3.3867

Legend: **Top Value** ; **Runner-up**

4 Model Evaluation – Deep-Dive

Algorithm	Type	Precision	Recall	MRR	RMSE
Top-N	Count	0.0357 (base)	0.1081 (base)	0.1811 (base)	160.617 (base)
Average Rating	Weighted	0.0643 (+1.8x)	0.1233 (+1.1x)	0.2024 (1.1x)	0.3744 (-160.2)
Collaborative Filtering	User-Cosine	0.1214 (+3.4x)	0.1860 (+1.7x)	**0.4054 (+2.2x)**	**2.4447 (-158.2)**
Hybrid	Recommended	**0.1000 (+2.8x)**	**0.1765 (+1.6x)**	0.4355 (+2.4x)	3.3867 (-157.2)

Weighted Average Rating

Best RMSE: Ultra-accurate rating predictions minimize churn risk

Moderate ranking (MRR): Only modest lift in retention

Use for accurate rating prediction

User-Cosine CF

Top ranking lift (Precision & MRR): Engagement & conversion gains

Lower RMSE than Hybrid: Good balance of accuracy & relevance

Use for precise personalized picks

Hybrid (UB-CF & AWR*)

Highest MRR: Prioritizes most relevant items, maxing revenue

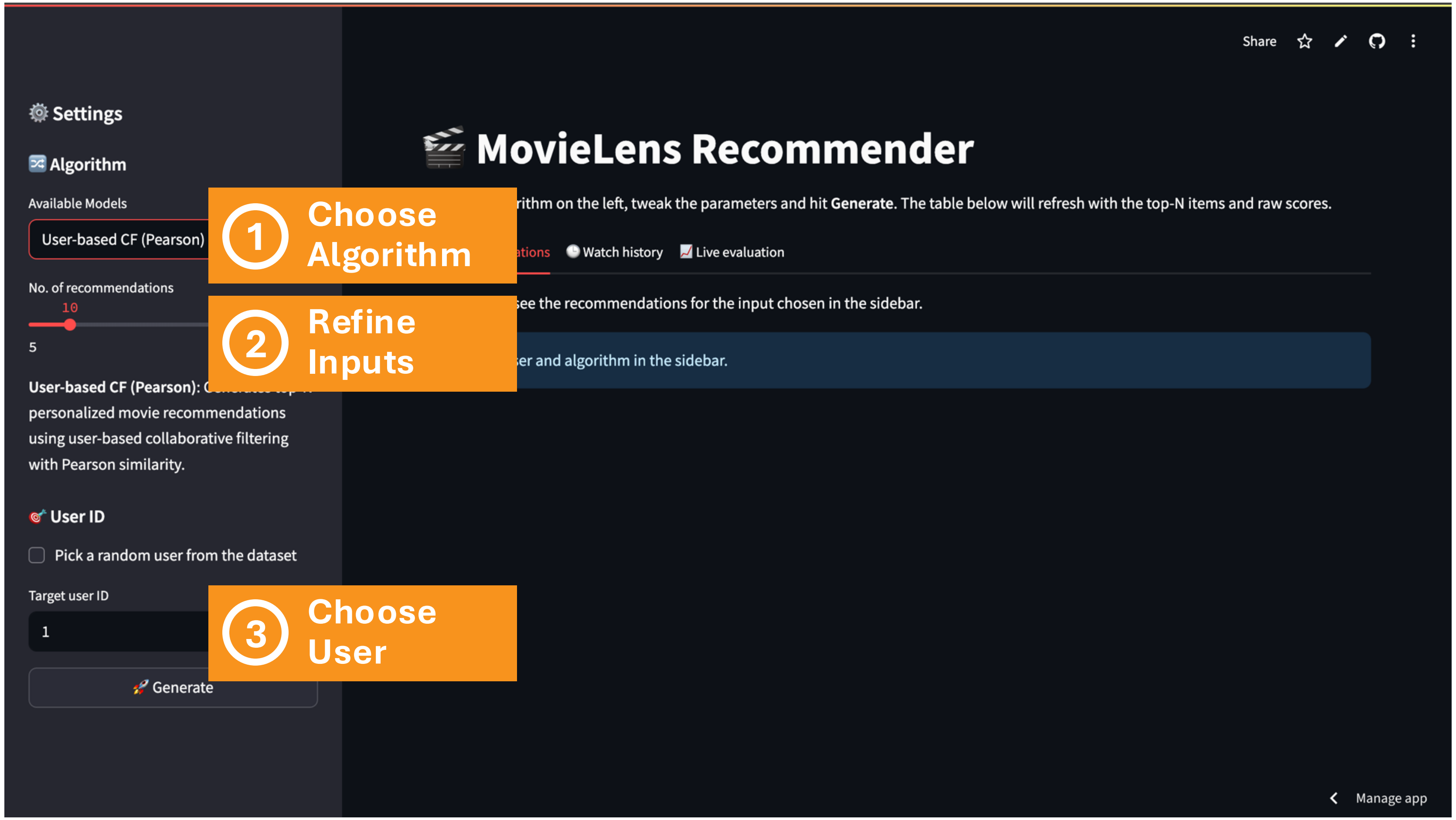
Slightly higher RMSE vs. CF: Still far better than baseline

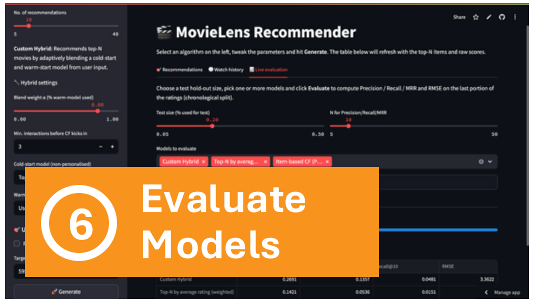
Use to max relevance & revenue

We would use the hybrid engine as it shows the best business performance

* Recommended hybrid is a blend of user-based collaborative filtering (UB-CF) and average weighted rating (AWR); Legend: Top Value ; **Runner-up**

5 Prototype Walkthrough





Live link: https://movielensrecommender.streamlit.app/

6 Business Implications

Area	Business Effect	Metric(s)
Conversion	When people see many relevant movies during their trial phase, they will convert to subscribers	Optimize for Precision
Engagement	Customers are more likely to watch a movie when the suggested movie appeals to their interest	Optimize for Recall
Marketing	Happy customers will let their friends know where they discovered their new favourite movie	Optimize for MRR
Retention	Once customers are subscribed, they are less willing to cancel when they feel like they keep discovering relevant new movies and get their money's worth	Optimize for RMSE
Revenue	As customers spend more time watching movies they like, we will earn more money through ads or more premium subscriptions as well as less churn	Optimize for MRR

Through a good recommender system, we can drastically increase CLTV

Thank you for your attention!

AI2: Supervised Classification

Timon Weidemann

esade