MOVIELENS RECOMMENDATION SYSTEM

Building a personalized movie recommendation system using collaborative and content-based filtering.

Business Understanding & Problem Statement

Enhanced User Experience

Personalized recommender systems guide users through vast digital catalogs, boosting satisfaction and discovery.

Unlock Long Tail Value

Intelligent filtering and prediction helps surface niche content that users may have overlooked.

Improved Business Outcomes

Personalized experiences drive user engagement, loyalty, and revenue growth for digital businesses.

Problem Statement

Recommend Relevant Movies

Provide personalized movie recommendations from vast catalogs to improve user satisfaction.

Discover New Content

Help users find new movies they'll love based on their individual preferences and viewing history.

Increase Engagement

Enable businesses to retain users and improve content engagement through tailored recommendations.

Project Overview

Objectives

Predict the top 5 movies personalized for each user based on their past ratings and behaviors.

Leverage collaborative filtering and hybrid approaches to handle cold start scenarios.

Key Features

- Dataset: MovieLens small with 100,004 ratings
- Core method: Collaborative filtering
- Evaluation: Precision@K > 0.6
- Stretch goal: Hybrid model for cold start

File	Records	Features
Movies	58,098	Title, Genres, Year
Ratings	100,004	UserID, MovieID, Rating (0.5-5.0)
Tags	3,684	User-generated metadata

Data Understanding & Cleaning

Data Merging

Combined movies, ratings, and tags into one dataset with 233,213 entries.

Cleaning

Checked for nulls and duplicates, removed outliers using IQR method.

Feature Engineering

Extracted release year from titles, split and one-hot encoded genres.

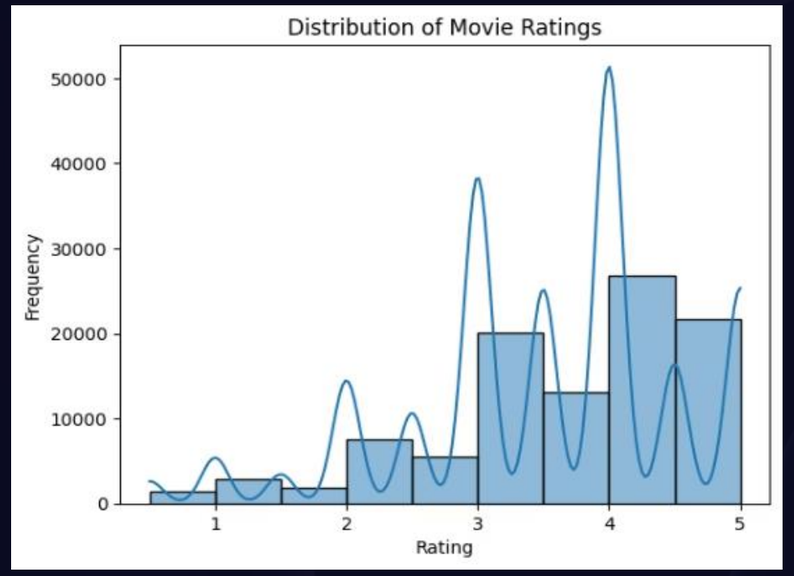
Normalization

Normalized ratings and timestamps for better model performance.

Exploratory Data Analysis

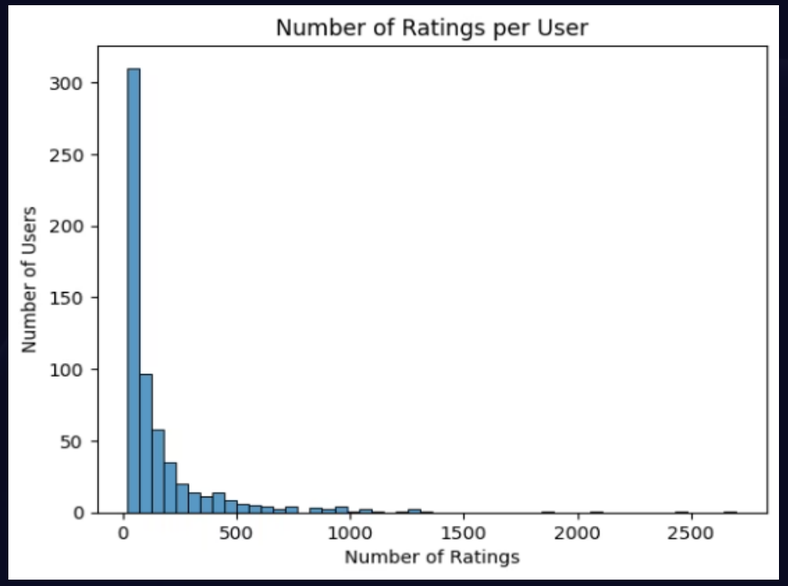
Rating Distribution

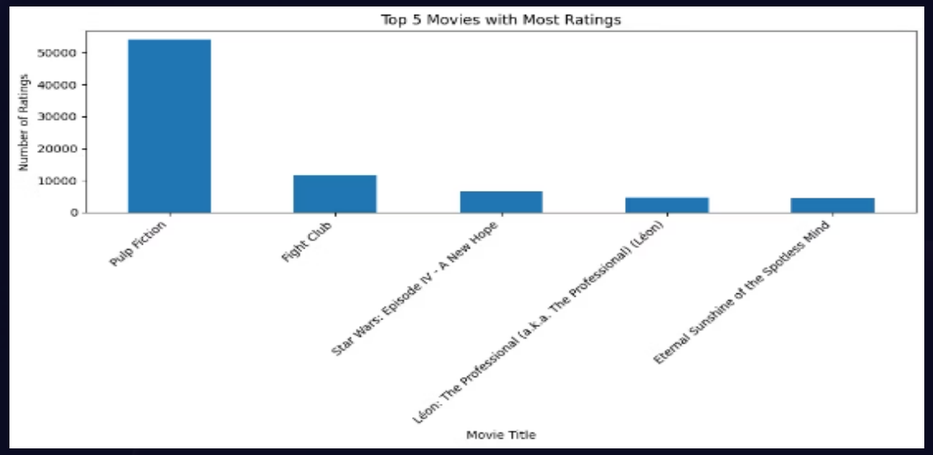
Ratings mostly between 3-5, with 4.0 and 5.0 most common.



Top Movies & Users

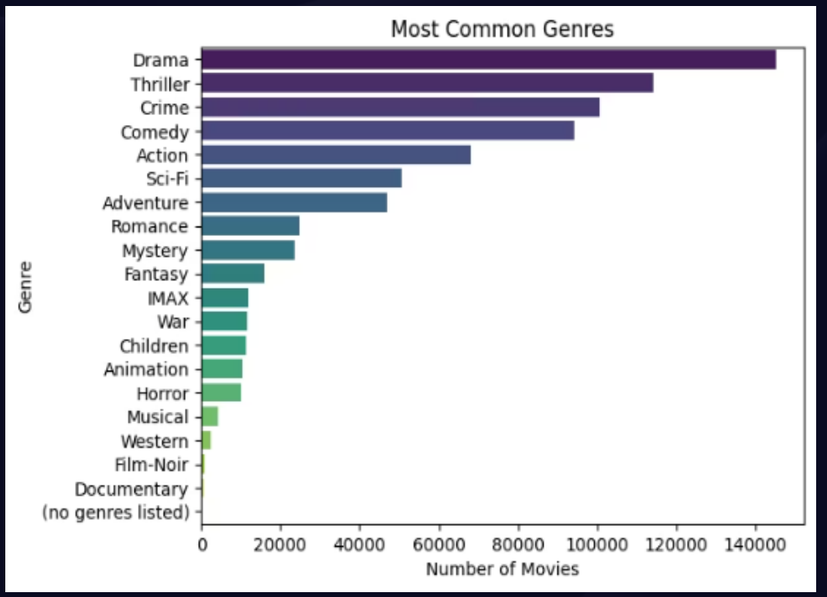
Few users rated majority of movies; Pulp Fiction highest rated.





Genres & Trends

Common genres: Drama, Comedy, Action. Frequent tags: Pixar, war, classic.



Modeling: Collaborative Filtering

User-Based CF

Low precision due to cold start and sparse ratings amounting to 98.3% in utility matrix

Item-Based CF

Improved precision (0.6) by focusing on item similarities.

Matrix Factorization (SVD)

Poor performance likely due to data reduction and mean imputation.

Content-Based & Hybrid Filtering

Content-Based Filtering

Uses tags and metadata but suffers from sparse TF-IDF matrix.

Hybrid Model

Combines content and item-based filtering, achieving precision of 0.6.

Best balance for cold start and recommendation accuracy

	Precision@K	Recall@K
User Based Collaborative Filtering	0.0	0.000
Item Based Collaborative Filtering	0.6	0.005
Matrix Vectorization SVD	0.0	0.000
Content Based Filtering	0.0	0.000
Hybrid: Content Based and Collaborative Filtering	0.6	0.005

Classification Models for Movie Preferences

Random Forest

Best performance with 87% precision and 83% recall.

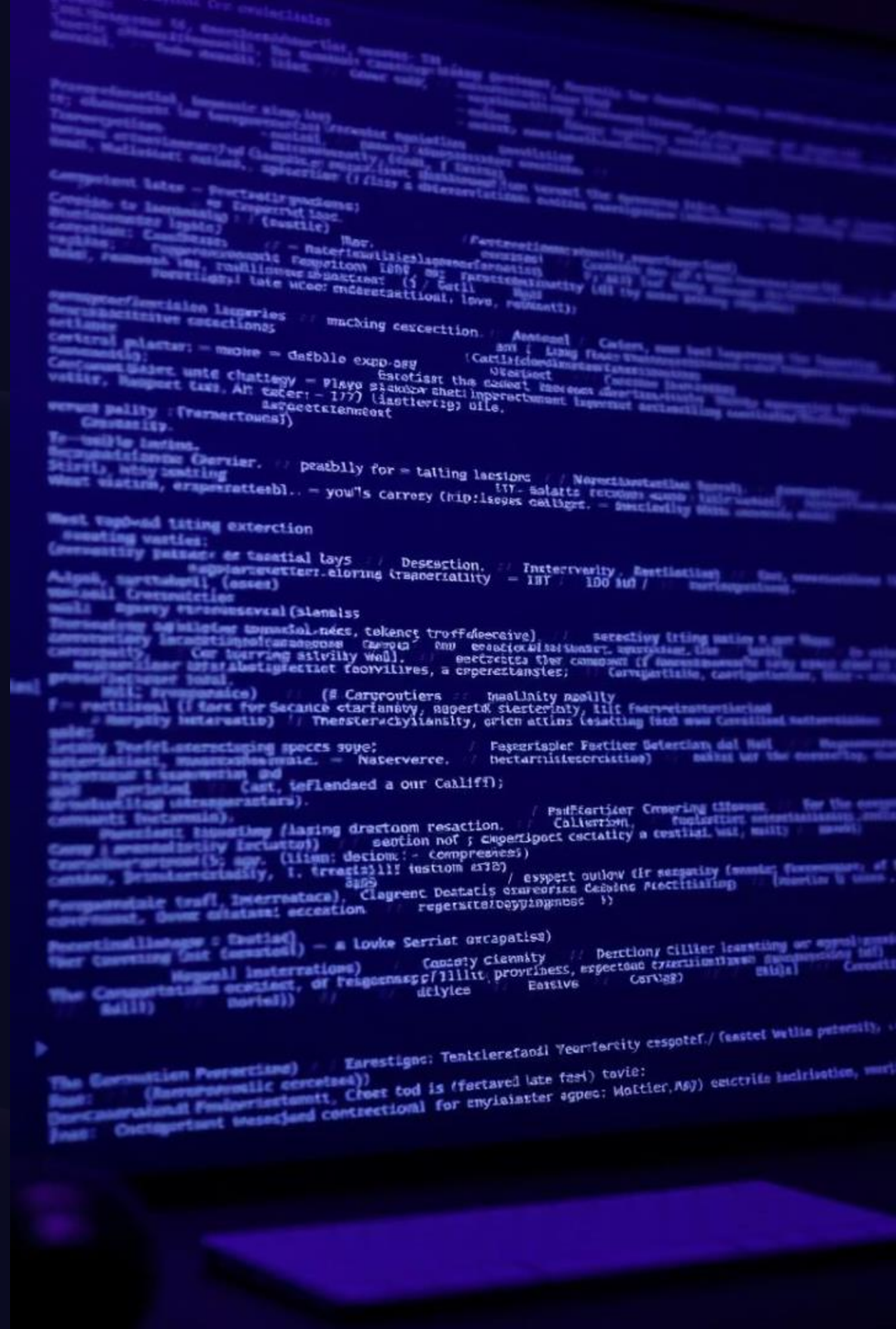
Decision Tree

Good precision (84%) and recall (85%), slightly below Random Forest.

Gradient Boosting & Neural Net

Poor recall and precision; need further tuning and balancing.

	precision	recall
Random Forest Classifier	0.875	0.8220
Gradient Boosting Classifier	0.747	0.0400
Neural Network Classifier	0.000	0.0000
Decision Tree Classifier	0.838	0.8408



Summary & Future Recommendations

Achievements

- **Collaborative Filtering**: Used user/item-based models. Reduced cold start issues.
- **Hybrid Model**: Combined collaborative & content-based filtering. Achieved **Precision@5 = 0.6**.
- **Classification Models**: Helped distinguish liked vs. disliked movies. Random Forest was most effective.
- **Top-5 for User 68** (*Pulp Fiction*):
- *Forrest Gump, Ferris Bueller's Day Off, Se7en, Silence of the Lambs, The Usual Suspects*

Success Criteria

- **High Precision**: 3/5 accurate recommendations
- **User Alignment**: Recommendations match preferences
- **Scalability**: Ready to incorporate new users/movies

Challenges

- Data sparsity & limited ratings
- Long training time; runtime crashes
- Future: Tune models, balance classes, add real-time UI

Next Steps

- Hyperparameter Tuning Fine-tuning model settings (like number of neighbors, depth, learning rate) could greatly improve accuracy and performance.
- Model Optimization Improving model efficiency through feature selection, dimensionality reduction, and regularization helps prevent overfitting.
- Computational Constraints Training large models requires significant memory and processing time—leading to crashes and delays without optimized hardware or cloud support.

Thank You!

Thanks for exploring our Movielens recommendation system project.

Prepared by:

- Maureen Ngahu
- Immaculate Kimani
- James Mwaura
- Maureen Maina
- Jael Kirwa
- Japhet Cheboiywo

Gith...