ReelSense: Explainable Movie Recommender System

Project Report

1. Project Overview

ReelSense is a movie recommendation system built using the MovieLens Latest Small dataset. The project develops an explainable recommender focusing on personalized recommendations, diversity, coverage, and natural language explanations.

2. Data Analysis and Preprocessing

2.1. Datasets

- **ratings.csv:** User ratings (0.5 to 5.0)
- **movies.csv:** Movie metadata (title, genres)
- **tags.csv:** User-assigned free-form tags
- **links.csv:** Movie ID mappings to IMDB, TMDb

2.2. Preprocessing Pipeline

- **Time-based Train-Test Split:** Leave-last-1 strategy per user
- **Feature Engineering:** One-hot encoding for genres (21 features) and tags (1,476 features)
- **Similarity Matrix:** Cosine similarity computed on combined features (9,742×9,742)

Dataset Statistics

Dataset	Shape
Training Ratings	100,226 × 4
Test Ratings	610 × 4
User-Item Matrix	610 users × 9,701 movies
Combined Movie Features	9,742 movies × 1,496 features

2.3. Key Findings from EDA

- **Rating Distribution:** Peak at 4.0-5.0; users rate movies they enjoy
- **Genre Popularity:** Drama, Comedy, Action most frequently rated
- **Average Ratings:** Film-Noir, Documentary, War genres have highest averages
- **User Activity:** Long-tail distribution; few highly active users, most provide few ratings
- **Movie Popularity:** Long-tail; blockbusters receive many ratings, niche movies rated infrequently

3. Popularity-Based Recommender

Baseline model identifying movies with highest average ratings (minimum 50 ratings threshold). Provides non-personalized benchmark for comparison.

Top 10 Popular Movies

Rank	Movie Title	Avg Rating	Count
1	Shawshank Redemption, The (1994)	4.43	315
2	Godfather, The (1972)	4.28	189
3	Fight Club (1999)	4.27	218
4	Cool Hand Luke (1967)	4.27	57
5	Dr. Strangelove (1964)	4.26	96
6	Godfather: Part II, The (1974)	4.25	128
7	Rear Window (1954)	4.25	83
8	Goodfellas (1990)	4.25	125
9	Departed, The (2006)	4.25	106
10	Princess Bride, The (1987)	4.24	141

4. Evaluation Metrics and Results

Model evaluated with K=10 recommendations per user in test set.

Metric	Value	Interpretation
Precision@10	0.0018	Very low prediction accuracy
Recall@10	0.0180	Captures few relevant items
NDCG@10	0.0096	Poor ranking quality
Catalog Coverage@10	0.0010	Uses only 0.1% of catalog
Intra-List Diversity@10	0.8079	High within-list diversity
Popularity-Normalized Hits	0.2069	Low novelty (expected)

4.1. Key Insights

- **Non-Personalized Limitations:** Low precision/recall/NDCG confirm inability to predict individual preferences
- **Severe Catalog Coverage:** Recommends only top 10 movies, missing 99.9% of catalog
- **Positive Aspect:** High intra-list diversity shows top movies differ in genre/tag features

5. Explainability Feature

Natural language explanations link recommendations to user's past preferences through shared genres and tags.

Example Explanations:

- **User 1, '20 Dates (1998)':** "Because you liked She's the One (1996), Wedding Singer, The (1998) and are both 'Comedy, Romance' films."
- **User 2, 'Town, The (2010)':** "Because you liked Departed, The (2006), Kill Bill: Vol. 1 (2003) and are both 'Thriller, Drama' films."
- **User 3, 'You've Got Mail (1998)':** "Because you liked The Lair of the White Worm (1988) and are both 'Comedy' films."

Impact: Explanations improve transparency, user trust, and system understanding by revealing recommendation logic.

6. Conclusions and Next Steps

6.1. Conclusions

The popularity-based baseline effectively demonstrates trade-offs between popularity, personalization, diversity, and novelty. While simple to implement, lack of personalization yields poor effectiveness metrics. The explainability feature provides valuable transparency.

6.2. Recommended Next Steps

- **Personalized Models:** Implement Collaborative Filtering, Matrix Factorization (SVD), Content-Based Filtering, and Hybrid approaches
- **Comparative Evaluation:** Benchmark personalized models against baseline using established metrics
- **Enhanced Explainability:** Integrate feature importance and latent factor interpretation
- **UI Integration:** Develop user interface for real-time recommendations and feedback

7. References

Dataset:

Harper, F. M., & Konstan, J. A. (2015). The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS), 5(4), Article 19. https://doi.org/10.1145/2827872

Libraries:

- Pandas: https://pandas.pydata.org/
- NumPy: https://numpy.org/
- Matplotlib: https://matplotlib.org/
- Seaborn: https://seaborn.pydata.org/
- Scikit-learn: https://scikit-learn.org/