Movie Recommendation System Using the MovieLens Dataset

About The Project

This project develops and evaluates a movie recommendation system using the MovieLens dataset (ml-latest-small) from GroupLens, containing 100,836 ratings from 610 users across 9,742 movies. The goal is to identify the most effective algorithm for predicting user ratings and generating high-quality recommendations, evaluated primarily through Recall@20 and NDCG@20, with secondary metrics including accuracy (±0.5 or ±0.25 tolerance), precision@20, and F1-score@20. Five algorithms were implemented: Matrix Factorization (MF), Tuned Matrix Factorization, Funk Singular Value Decomposition (SVD) with Robust Principal Component Analysis (RPCA), Neural Collaborative Filtering (NCF), and NCF with Contrastive Learning. Key challenges addressed include:

1. **Data Sparsity**: The rating matrix is ~1.7% filled, mitigated by filtering users (≥20 ratings) and movies (≥5 ratings).
2. **Cold-Start Problem**: Handled by focusing on users and movies with sufficient ratings.
3. **Overfitting**: Addressed with regularization (L2 in NCF, weight decay in MF) and early stopping.
4. **Model Complexity**: Balanced simple (MF, Funk SVD) and complex (NCF) models to optimize performance and interpretability.
5. **Generalization**: Ensured through user-based data splitting (70% train, 15% CV, 15% test) and hyperparameter tuning.

Models were trained and evaluated on a filtered dataset (~90,274 ratings) with the following strategies:

- **Challenges 1 & 2 (Sparsity & Cold-Start)**: Applied filtering to reduce sparsity and improve data quality.
- **Challenge 3 (Overfitting)**: Used regularization, early stopping, and contrastive learning in NCF to enhance generalization.
- **Challenge 4 (Model Complexity)**: Compared linear (MF, SVD) and non-linear (NCF) approaches for optimal performance.

To further optimize the Neural Collaborative Filtering (NCF) and NCF with Contrastive Learning models, several experiments were conducted to evaluate performance improvements:

- **NCF Experiments**:
    1. **Patience Parameter**: Doubling the patience parameter increased training epochs from 8 to 20, improving Recall@20 from 0.2906 to 0.3168 and NDCG@20 from 0.1941 to 0.2155, indicating better model convergence.
    2. **Dropout Parameter**: Increasing the dropout rate from 0.2 to 0.9 further improved Recall@20 from 0.3168 to 0.3631 and NDCG@20 from 0.2155 to 0.2297, enhancing generalization by reducing overfitting.
    3. **Activation Function in NeuMF Layer**: Changing the activation function from none to sigmoid resulted in Recall@20 of 0.2944 and NDCG@20 of 0.1962, which was better than the initial baseline but lower than the dropout experiment.
    4. **Number of MLP Layers**:
        - Increasing from 3 to 5 layers slightly improved performance, with Recall@20 increasing from 0.3631 to 0.3640 and NDCG@20 from 0.2297 to 0.2298.
        - Decreasing to 2 layers yielded the best NCF performance, with Recall@20 of 0.3651 and NDCG@20 of 0.2366, suggesting a simpler architecture improved generalization.
    5. **Defining Numerical Variables**: Defining numerical variables (e.g., hyperparameters) at the beginning of the code improved reproducibility and ease of tuning.
- **NCF with Contrastive Learning Experiments**: Applying the above parameter changes (patience, dropout, activation, MLP layers) improved performance from a baseline of Recall@20=0.3494 and NDCG@20=0.2330 to Recall@20=0.3748 and NDCG@20=0.2447, demonstrating the effectiveness of these optimizations in enhancing ranking quality and recommendation relevance.

Next Steps

To further improve the recommendation system:

1. Incorporate content-based features (e.g., movie genres) to enhance hybrid recommendations.
2. Explore advanced contrastive learning techniques (e.g., SimCLR) for better embedding alignment.
3. Optimize hyperparameters using grid search or Bayesian optimization to boost Recall@20 and NDCG@20.
4. Implement attention mechanisms in NCF to capture complex user-item interactions.
5. Test ensemble methods combining MF, SVD, and NCF for improved recommendation quality.
6. Validate on larger datasets (e.g., MovieLens 1M) for better generalizability.

Models Performance Comparison

Model	Split	Accuracy	Precision@20	Recall@20	F1-Score@20	NDCG@20
Matrix Factorization	Train	0.7987 (±0.5)	0.4011	0.7822	0.4711	0.7678
	CV	0.4008 (±0.5)	0.0666	0.3688	0.2320	0.2238
	Test	0.4088 (±0.5)	0.0716	0.3712	0.2400	0.2311
Tuned-Matrix Factorization	Train	0.8000 (±0.5)	0.3960	0.7689	0.4671	0.7575
	CV	0.4004 (±0.5)	0.0677	0.3586	0.2343	0.2217
	Test	0.4103 (±0.5)	0.0719	0.3739	0.2377	0.2372

FunkSVD+RPCA	Train	0.8825 (±0.25)	0.4348	0.7872	0.5432	0.7834
	CV	0.2277 (±0.25)	0.0582	0.2744	0.2633	0.1695
	Test	0.2251 (±0.25)	0.0621	0.2854	0.2667	0.1804
NCF	Train	0.2725 (±0.25)	0.2659	0.5193	0.3567	0.5681
	CV	0.2533 (±0.25)	0.0725	0.3460	0.2613	0.2272
	Test	0.2476 (±0.25)	0.0757	0.3658	0.2658	0.2390
NCF+Contrastive Learning	Train	0.1668 (±0.25)	0.2778	0.5623	0.3445	0.6258
	CV	0.1101 (±0.25)	0.0706	0.3977	0.2255	0.2474
	Test	0.1109 (±0.25)	0.0730	0.3947	0.2342	0.2478

Comparison with Papers

- **Xiangnan et al.(2017), Neural Collaborative Filtering**: On MovieLens 1M (1M ratings), their NeuMF model achieved HR@10≈0.71 and NDCG@10≈0.426–0.450. My NCF model reached Recall@20=0.3658 and NDCG@20=0.2390, and NCF with Contrastive Learning hit (Recall@20=0.3947 and NDCG@20=0.2478) on MovieLens ml-latest-small (90K ratings). The gap is due to the smaller dataset and simpler architectures, but contrastive learning improved ranking quality.

- **Koren et al. (2009), Matrix Factorization Techniques for Recommender Systems, IEEE Computer 2009**: Their SVD++ model on Netflix (~100M ratings) achieved RMSE≈0.86, with strong ranking performance. My Funk SVD + RPCA yielded Recall@20=0.2854 and NDCG@20=0.1804, likely with higher RMSE, due to overfitting on the smaller MovieLens ml-latest-small dataset. Matrix Factorization (Recall@20=0.3712, NDCG@20=0.2311) and Tuned MF (Recall@20=0.3739, NDCG@20=0.2372) performed better, showing effective tuning despite dataset limitations.

Conclusion

The key evaluation metrics for this project are Recall@20, which measures the ability to recommend relevant items, and NDCG@20, which assesses ranking quality. NCF with Contrastive Learning outperformed all models in both Recall@20 (0.3947) and NDCG@20 (0.2478) on the test set, demonstrating superior ability to recommend relevant movies and rank them effectively. Tuned Matrix Factorization followed closely (Recall@20=0.3739, NDCG@20=0.2372), slightly improving over standard Matrix Factorization (Recall@20=0.3712, NDCG@20=0.2311). Funk SVD + RPCA, despite strong training performance (Recall@20=0.7872, NDCG@20=0.7834), showed the weakest generalization (Recall@20=0.2854, NDCG@20=0.1804). Compared to literature, my models lag behind advanced methods like SVD++ or NCF on larger datasets due to the limited size of ml-latest-small, but NCF with Contrastive Learning is competitive in ranking quality. Filtering, regularization, and contrastive learning significantly enhanced performance. Future work includes integrating content-based features, advanced ensembles, and testing on larger datasets to further improve Recall@20 and NDCG@20.