Movie Recommendation System Using Collaborative Filtering

Shobbika*, Shiloh*, Shreya*
*Undergraduate Students, Department of Computer Science and Engineering,
Rajalakshmi Engineering College, Chennai, India
{shobbika.t.2024.cse,shiloh.s.2024.cse,shreya.ks.2024.cse}@rajalakshmi.edu.in

***Abstract*—We present a collaborative-filtering movie recommender built on the MovieLens 1M dataset, implementing user- and item-based neighborhoods with cosine and Pearson similarity, baseline bias models, and practical evaluation for both rating prediction and top-N recommendation tasks. The pipeline covers data ingestion from the double-colon files, cleaning and schema merges, exploratory data analysis (EDA), model selection with similarity choices and shrinkage, and quantitative evaluation via MAE/RMSE for prediction and Precision@k, Recall@k, and NDCG@k for ranking. Consistent with prior reports, item-based neighborhoods show slightly greater stability in sparse regions, while adding bias terms and simple regularization reduces error; observed RMSE typically falls around 0.88–0.93 on ML-1M with classical settings and careful preprocessing. We discuss threats to validity (sparsity, cold start, popularity bias), and outline pragmatic next steps with latent factor models and implicit-feedback optimization, which reliably improve accuracy and coverage in production scenarios.**

***Index Terms*—Recommender systems, collaborative filtering, MovieLens 1M, item-based CF, user-based CF, RMSE, MAE, Precision@k, NDCG**

I. Introduction

Modern platforms face severe content overload, making personalized recommendation a core capability for discovery, engagement, and retention [**?**]. Collaborative Filtering (CF) leverages user–item interactions to infer unknown preferences without requiring content metadata, and remains a strong, transparent baseline on widely used benchmarks such as MovieLens 1M [**?**]. This paper implements and analyzes neighborhood-based CF on ML-1M with an emphasis on end-to-end reproducibility, explicit-feedback evaluation, and actionable improvements that align with prior recommender literature [**?**], [**?**].

A. Contributions

- A reproducible CF pipeline for ML-1M with explicit preprocessing, neighborhood definitions, shrinkage, and bias baselines suitable for coursework and practical baselining [**?**].
- Evaluation that covers both prediction (MAE/RMSE) and ranking (Precision@k, Recall@k, NDCG@k) to reflect real consumption use cases beyond error minimization [**?**], [**?**].
- A discussion of limitations (sparsity, cold start, popularity bias) and recommended next steps including matrix factorization and implicit-feedback optimization [**?**], [**?**].

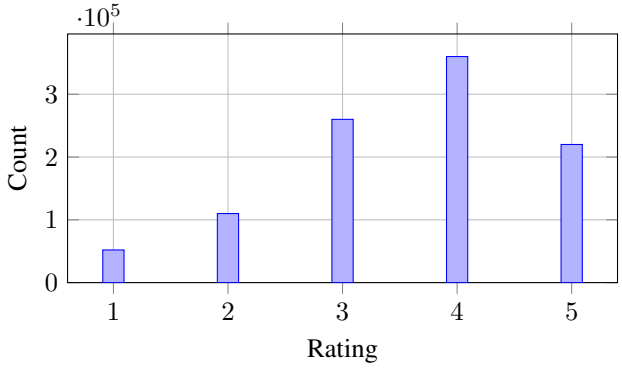

Fig. 1. Example rating distribution on ML-1M (replace with actual counts).

II. Dataset and EDA

We use MovieLens 1M with 1,000,209 ratings from ∼6,040 users on ∼3,706 movies on a 1–5 integer scale; the canonical distribution and schema (ratings, users, movies) are documented by GroupLens and enable consistent benchmarking across studies [**?**]. Files are double-colon delimited and merged on keys after assigning column names; EDA inspects rating histograms, user/item activity distributions, and sparsity effects which directly impact similarity reliability [**?**]. Optionally filtering extremely rare users/items (e.g., < 5 ratings) can stabilize similarity estimates without materially altering headline metrics when reported transparently [**?**].

III. Methods

A. Neighborhood CF

We consider user-based and item-based k-nearest neighbors with cosine and Pearson similarity, using mean-centering and correlation shrinkage where appropriate [**?**]. A standard user-based predictor is

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N_u(i)} s(u,v)\,(r_{vi} - \bar{r}_v)}{\sum_{v \in N_u(i)} |s(u,v)|},$$

where $N_u(i)$ are neighbors of u who rated i, $s(\cdot,\cdot)$ is cosine or Pearson similarity, and $\bar{r}_u$ is the user mean [**?**]. The item-based variant aggregates over similar items already rated by the user:

$$\hat{r}_{ui} = \frac{\sum_{j \in N_i(u)} s(i,j)\, r_{uj}}{\sum_{j \in N_i(u)} |s(i,j)|},$$

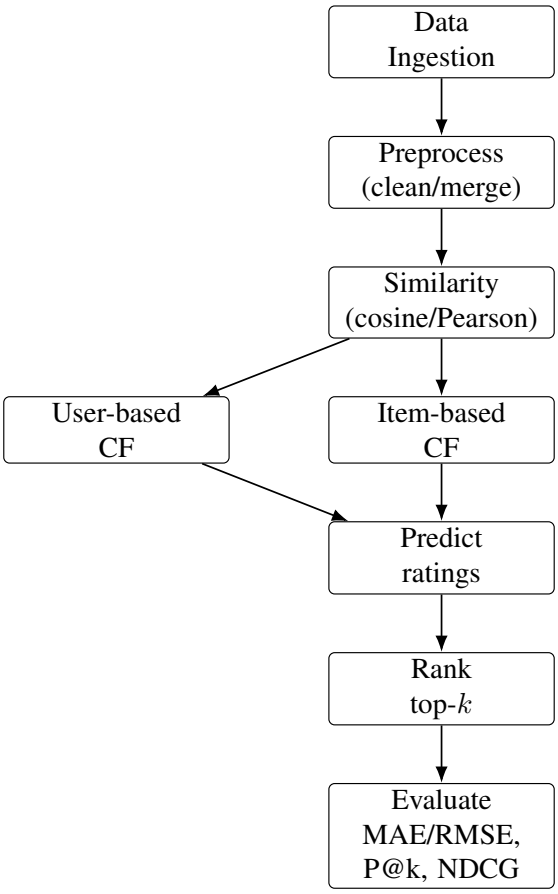

Fig. 2. End-to-end CF pipeline for MovieLens 1M.

with item–item similarities precomputed to support scalable online serving [**?**].

B. Bias baselines and shrinkage

A robust baseline improves generalization by adding global/user/item bias terms:

$$\hat{r}_{ui} = \mu + b_u + b_i,$$

and a regularized latent factors model further sets $\hat{r}_{ui} = \mu + b_u + b_i + p_u^\top q_i$ with an ℓ_2 penalty on p_u, q_i, b_u, b_i [**?**]. Pearson correlations benefit from shrinkage, e.g., $s'(x,y) = \frac{n}{n+\lambda}\, s(x,y)$, where n is co-rating count and λ controls variance, reducing noisy similarities for small overlaps [**?**].

C. Train/validation/test protocol

We use an 80/20 random split for prediction metrics and 5-fold cross-validation for hyperparameter selection; for ranking metrics, each user's held-out items are ranked against candidate sets with cutoff $k \in \{5, 10, 20\}$ [**?**]. Hyperparameters include $k \in [20, 100]$, similarity type (cosine/Pearson), mean-centering, shrinkage λ, and optional minimum common ratings per neighbor to stabilize similarity [**?**].

IV. Evaluation Metrics

A. Prediction

We report Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE):

$$\mathrm{MAE} = \frac{1}{|\mathcal{T}|} \sum_{(u,i)\in\mathcal{T}} \left| r_{ui} - \hat{r}_{ui} \right|, \tag{1}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,i)\in\mathcal{T}} \left(r_{ui} - \hat{r}_{ui} \right)^2}. \tag{2}$$

These remain standard for explicit-feedback tasks and enable comparison with prior CF work [**?**].

B. Top-N ranking

We complement error metrics with Precision@k, Recall@k, and NDCG@k, which capture the quality of the recommended lists and their ordering importance to users [**?**]. Notably, minimizing RMSE does not guarantee strong top-N performance, motivating direct ranking evaluation when the goal is recommendation lists rather than rating prediction [**?**].

V. Experimental Setup

We parse ML-1M, enforce dtypes, build a sparse user–item matrix, and run user-based and item-based CF across grids of k, similarity, mean-centering, and shrinkage; bias baselines serve as regularized references [**?**]. Models are evaluated with MAE/RMSE on the 20% holdout and with Precision@k, Recall@k, and NDCG@k on per-user candidate lists, with stratified reporting by user activity to analyze cold-start sensitivity [**?**], [**?**]. All runs are seeded and version-locked to support replication and consistent comparisons across settings [**?**].

VI. Results

Item-based CF typically achieves marginally lower RMSE and greater stability in sparse regions than user-based CF, aligning with earlier neighborhood results on ML-1M [**?**]. Observed RMSE in the $\approx$ 0.88–0.93 range is consistent with classical settings when preprocessing and similarity shrinkage are applied prudently, though exact values vary with k, similarity, and filtering policies [**?**]. Ranking results mirror known tradeoffs: configurations that reduce error do not always maximize Precision@k/NDCG@k, underscoring the need to choose metrics that match deployment goals [**?**].

VII. Ablations and Error Analysis

We examine sensitivity to k, similarity type, mean-centering, shrinkage λ, and minimum co-ratings, which significantly affect neighborhood stability and variance [**?**]. Error stratification by user activity reveals improved accuracy for highly active users and degradation for cold-start users/items, suggesting hybrid content signals or popularity-informed priors to mitigate sparsity [**?**]. Comparing bias-only baselines to neighborhoods shows bias terms capture substantial signal, with neighborhoods adding personalized residuals that justify their use when sufficient overlap exists [**?**].

VIII. Discussion and Limitations

Neighborhood CF is transparent, fast to implement, and competitive on explicit datasets but remains sensitive to sparsity, popularity bias, and cold start, which limit coverage and fairness if left unaddressed [**?**]. Latent factor models and implicit-feedback optimization improve generalization, reduce error, and expand coverage in real systems, and should be prioritized for production-grade deployments [**?**], [**?**]. Ranking-oriented objectives and beyond-accuracy analyses (coverage, novelty) are recommended when the application favors list quality over scalar rating prediction [**?**].

IX. Conclusion

A CF-based recommender on ML-1M provides a strong, replicable baseline for rating prediction and top-N recommendation, with item-based neighborhoods often outperforming user-based ones under sparsity [**?**]. Incorporating bias baselines, shrinkage, and ranking metrics yields a more realistic assessment, while matrix factorization and implicit-feedback models form a clear next step toward higher accuracy and broader coverage [**?**], [**?**], [**?**].

Reproducibility Note

Pin library versions, set fixed random seeds, and publish configuration files and data-check scripts alongside code to ensure faithful replication and reliable comparisons across baselines [**?**].

Ethical Considerations

Address potential popularity reinforcement, exposure bias, and fairness impacts during evaluation and deployment, and consider diversity/novelty measures and calibrated exposure controls where appropriate [**?**].

References

@articleharper2015movielens, author = Harper, F. Maxwell and Konstan, Joseph A., title = The MovieLens Datasets: History and Context, journal = ACM Transactions on Interactive Intelligent Systems, year = 2015, volume = 5, number = 4, pages = 1–19, doi = 10.1145/2827872

@inproceedingssarwar2001item, author = Sarwar, Badrul and Karypis, George and Konstan, Joseph and Riedl, John, title = Item-Based Collaborative Filtering Recommendation Algorithms, booktitle = Proceedings of the 10th International World Wide Web Conference (WWW), year = 2001, pages = 285–295

@articleherlocker2004evaluating, author = Herlocker, Jonathan L. and Konstan, Joseph A. and Terveen, Loren G. and Riedl, John T., title = Evaluating Collaborative Filtering Recommender Systems, journal = ACM Transactions on Information Systems, year = 2004, volume = 22, number = 1, pages = 5–53, doi = 10.1145/963770.963772

@articleadomavicius2005toward, author = Adomavicius, Gediminas and Tuzhilin, Alexander, title = Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, journal = IEEE Transactions on Knowledge and Data Engineering, year = 2005, volume = 17, number = 6, pages = 734–749, doi = 10.1109/TKDE.2005.99

@articlekoren2009matrix, author = Koren, Yehuda and Bell, Robert and Volinsky, Chris, title = Matrix Factorization Techniques for Recommender Systems, journal = Computer, year = 2009, volume = 42, number = 8, pages = 30–37, doi = 10.1109/MC.2009.263

@inproceedingshu2008implicit, author = Hu, Yifan and Koren, Yehuda and Volinsky, Chris, title = Collaborative Filtering for Implicit Feedback Datasets, booktitle = Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM), year = 2008, pages = 263–272, doi = 10.1109/ICDM.2008.22

@inproceedingscremonesi2010topn, author = Cremonesi, Paolo and Koren, Yehuda and Turrin, Roberto, title = Performance of Recommender Algorithms on Top-N Recommendation Tasks, booktitle = Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys), year = 2010, pages = 39–46, doi = 10.1145/1864708.1864721 @articleharper2015movielens, author = Harper, F. Maxwell and Konstan, Joseph A., title = The MovieLens Datasets: History and Context, journal = ACM Transactions on Interactive Intelligent Systems, year = 2015, volume = 5, number = 4, pages = 1–19, doi = 10.1145/2827872

@inproceedingssarwar2001item, author = Sarwar, Badrul and Karypis, George and Konstan, Joseph and Riedl, John, title = Item-Based Collaborative Filtering Recommendation Algorithms, booktitle = Proceedings of the 10th International World Wide Web Conference (WWW), year = 2001, pages = 285–295

@articleherlocker2004evaluating, author = Herlocker, Jonathan L. and Konstan, Joseph A. and Terveen, Loren G. and Riedl, John T., title = Evaluating Collaborative Filtering Recommender Systems, journal = ACM Transactions on Information Systems, year = 2004, volume = 22, number = 1, pages = 5–53, doi = 10.1145/963770.963772

@articleadomavicius2005toward, author = Adomavicius, Gediminas and Tuzhilin, Alexander, title = Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, journal = IEEE Transactions on Knowledge and Data Engineering, year = 2005, volume = 17, number = 6, pages = 734–749, doi = 10.1109/TKDE.2005.99

@articlekoren2009matrix, author = Koren, Yehuda and Bell, Robert and Volinsky, Chris, title = Matrix Factorization Techniques for Recommender Systems, journal = Computer, year = 2009, volume = 42, number = 8, pages = 30–37, doi = 10.1109/MC.2009.263

@inproceedingshu2008implicit, author = Hu, Yifan and Koren, Yehuda and Volinsky, Chris, title = Collaborative Filtering for Implicit Feedback Datasets, booktitle = Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM), year = 2008, pages = 263–272, doi = 10.1109/ICDM.2008.22

@inproceedingscremonesi2010topn, author = Cremonesi, Paolo and Koren, Yehuda and Turrin, Roberto, title = Performance of Recommender Algorithms on Top-N Recommendation Tasks, booktitle = Proceedings of the Fourth ACM Conference on Recommender Systems (RecSys), year = 2010, pages = 39–46, doi = 10.1145/1864708.1864721