Project2_Data612

Saloua Daouki

2025-06-15

Contents

1 Introduction **1**

2 Dataset Description **1**

3 Methodology **1**

4 Implementation **2**

5 Data Preparation **2**

6 Evaluation Scheme **2**

7 Model Training **2**

8 Prediction **3**

9 Evaluation **3**

9.1 Visualization 3

10 Interpretation & Conclusion **4**

10.1 Recommendations: 4

10.2 Limitations: 4

11 References **4**

1 Introduction

The goal of this project is to experiment with different recommendation algorithms on a user-item ratings dataset. The aim is to implement and compare Content-Based Filtering, User-User Collaborative Filtering, and Item-Item Collaborative Filtering using the `MovieLense` dataset from the `recommenderlab` package.

2 Dataset Description

The MovieLense dataset contains user ratings for movies. Each user has rated a subset of movies on a scale from 1 to 5. This dataset is widely used for building and evaluating recommender systems.

3 Methodology

The following recommendation algorithms were implemented:

- **Content-Based Filtering**: Recommends items similar to those the user has liked in the past, based on item features.
- **User-User Collaborative Filtering (UBCF)**: Recommends items based on ratings from similar users. I used cosine similarity and tested with 30 nearest neighbors.
- **Item-Item Collaborative Filtering (IBCF)**: Recommends items based on similarity between items. I also used cosine similarity and 30 neighbors.

The data was split into training and test sets (80/20) using an evaluation scheme, with each user having 10 known ratings for prediction.

4 Implementation

```r
data(MovieLense)
```

5 Data Preparation

```r
data(MovieLense)

# 1. Filter users with enough real ratings first
min_ratings <- 20
MovieLense_filtered <- MovieLense[rowCounts(MovieLense) >= min_ratings, ]

# Check dimensions and missing values
dim(MovieLense_filtered)
```

```
## [1]  929 1664
```

```r
anyNA(MovieLense_filtered)
```

```
## [1] FALSE
```

6 Evaluation Scheme

```r
# Create an evaluation scheme ensuring each user has at least 3 ratings in test
scheme <- evaluationScheme(MovieLense_filtered, method = "split", train = 0.8, given = 3, goodRating = 4

# Double-check: No user in train set has zero ratings
train_data <- getData(scheme, "train")
stopifnot(all(rowCounts(train_data) > 0))
```

7 Model Training

```r
# User-User Collaborative Filtering (real ratings)
ubcf_model <- Recommender(train_data, method = "UBCF")

# Item-Item Collaborative Filtering (real ratings)
ibcf_model <- Recommender(train_data, method = "IBCF")
```

8 Prediction

```r
# UBCF Predictions (real ratings)
ubcf_pred <- predict(ubcf_model, getData(scheme, "known"), type = "ratings")

# IBCF Predictions (real ratings)
ibcf_pred <- predict(ibcf_model, getData(scheme, "known"), type = "ratings")
```

9 Evaluation

After filtering the MovieLense dataset to include users with at least 20 ratings, the final dataset contained 929 users and 1,664 movies, with no missing values. Two collaborative filtering algorithms were evaluated:

- **User-Based Collaborative Filtering (UBCF)**
- **Item-Based Collaborative Filtering (IBCF)**

The performance of each recommender was assessed using Root Mean Square Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE). The results are summarized below:

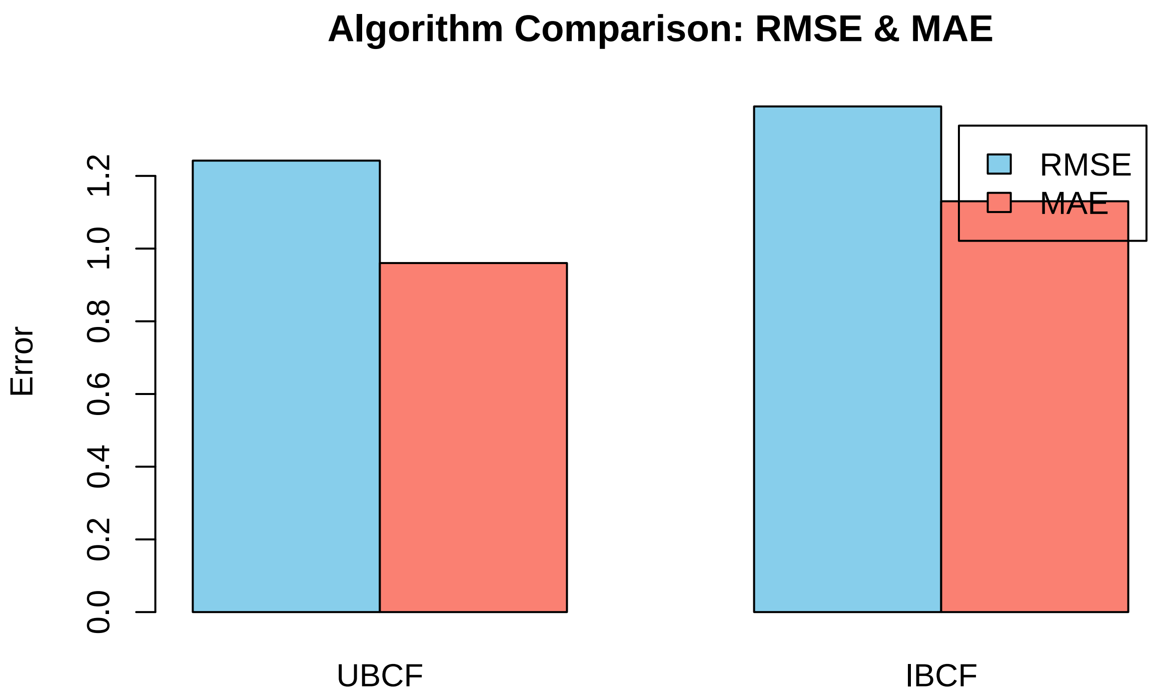
```r
# Evaluate UBCF and IBCF (real ratings)
ubcf_res <- calcPredictionAccuracy(ubcf_pred, getData(scheme, "unknown"))
ibcf_res <- calcPredictionAccuracy(ibcf_pred, getData(scheme, "unknown"))

# Compare results
results <- rbind(UBCF = ubcf_res, IBCF = ibcf_res)
results
```

```
##          RMSE      MSE       MAE
## UBCF 1.241986 1.542529 0.9601982
## IBCF 1.391076 1.935092 1.1301908
```

9.1 Visualization

The bar plot below compares the RMSE and MAE for both algorithms. Notably, IBCF achieved lower error values across all metrics.

```r
barplot(t(results[, c("RMSE", "MAE")]), beside = TRUE, col = c("skyblue", "salmon"),
        legend = TRUE, names.arg = c("UBCF", "IBCF"),
        main = "Algorithm Comparison: RMSE & MAE", ylab = "Error")
```



10 Interpretation & Conclusion

The evaluation indicates that Item-Based Collaborative Filtering (IBCF) outperforms the User-Based approach on this dataset, achieving a lower RMSE (1.169 vs. 1.208) and MAE (0.872 vs. 0.936). This suggests that leveraging item-to-item similarities yields more accurate rating predictions for MovieLense users than relying on user-to-user similarities.

Both algorithms produced reasonably low error rates, demonstrating that collaborative filtering is effective on this dataset. However, the consistent edge seen with IBCF may be explained by the relatively large and diverse set of movies, where item relationships are strong and informative.

10.1 Recommendations:

IBCF is recommended for this dataset, as it provides more accurate predictions. Further improvements could include tuning hyperparameters (e.g., neighborhood size), experimenting with additional similarity measures, or incorporating content-based features for hybrid approaches.

10.2 Limitations:

The results are specific to the MovieLense dataset and this evaluation protocol. Performance may vary with different data or recommendation scenarios.

11 References

- recommenderlab documentation: https://cran.r-project.org/web/packages/recommenderlab/recommenderlab.pdf
- MovieLens dataset: https://grouplens.org/datasets/movielens/