Movie Recommendation System using the MovieLens data set

Instructor: Prof. Pawlicki
Haoyu Li &Zenghe Huang

Outline

- Objective
- Data Preprocessing
- Technical Approach
- Results
- Future Expectation

Objective

- The main objective of this project is to build the Movie Recommendation Systems using the MovieLens data set.
- The recommendation system shall predict the ratings of a movie that the user haven’t seen yet.
- Using both user-user approach and item-item approach

Data Preprocessing

- The dataset we used is MovieLens 20M Dataset, which includes tag genome data with 12 million relevance scores across 1,100 tags.
- Relabeling, Shrink the Dataset, Convert to HashMaps
- Get rid of useless information, Time Stamp in this case
- We only use rating.csv and movie.csv

Data Preprocessing-Step 1

- Using rating.csv
- Reassigning the Movie IDs
  - The movie IDs are not sequential, they go from 1-100k
  - There're only around 20k movies
  - Loop through the entire dataset, make new mapping that goes from 0 to 20k (Make the MovieID consecutive)

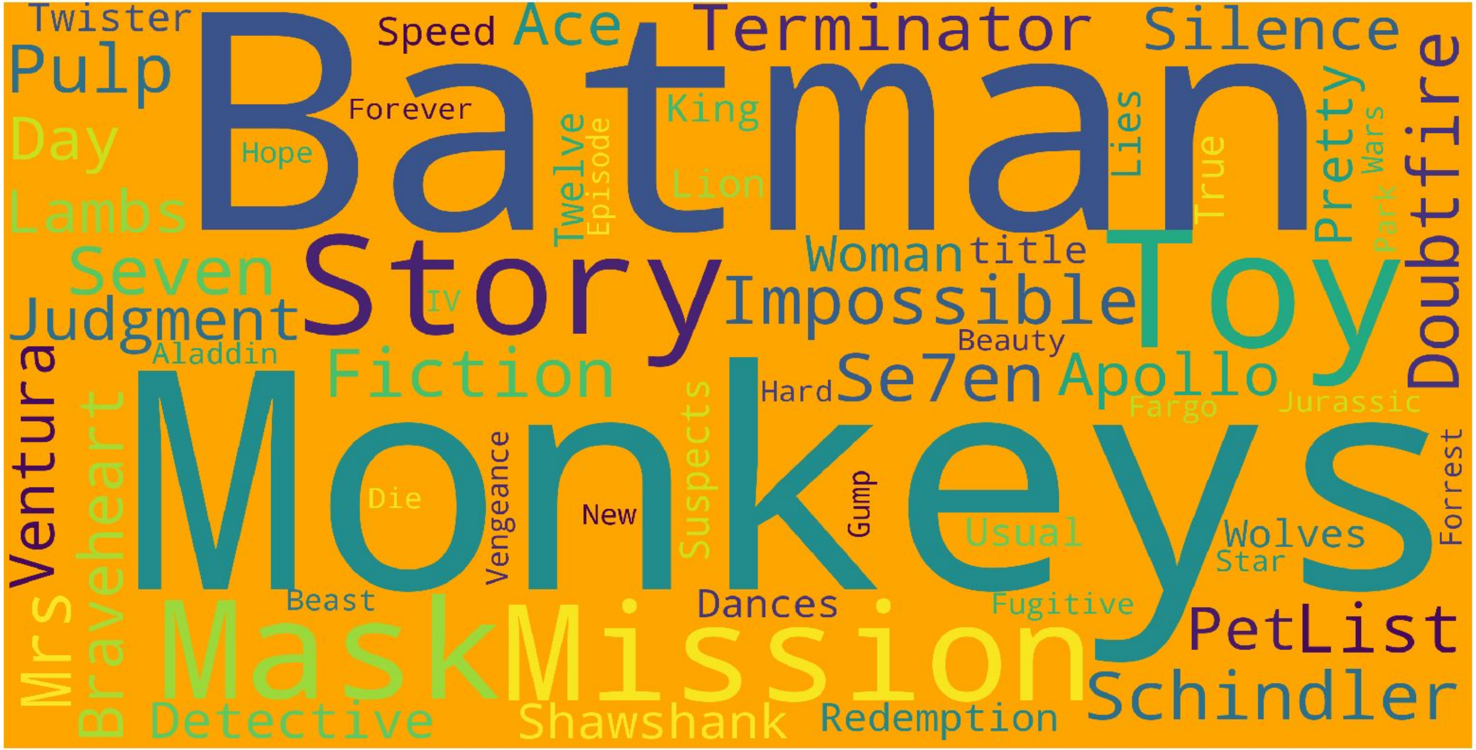
Data Preprocessing-Step 2

- Shrinking the dataset
  - The dataset is too big to perform an O(N^2) algorithm
  - Shrinking the dataset by:
    - Selecting subset of users and movies (# of users n: 1000, # of movies m: 50)
    - Top n users who rated the most movies
    - Top m movies who've been rated by most users

Data Preprocessing-Step 3

- Convert table to HashMaps, so that we can use the key-value pairs to lookup the data
  - Table is not an ideal data structure to access the data, we use hashmaps
  - Given user i, which movies j did they rate?
    - userToMovie(i)
  - Given movie j, which users i have rated it before?
    - movieToUser(j)
  - Given user i and movie j, what is the rating?
    - userMovieToRating(i,j)
  - Given movie j, what's the corresponding title?
    - movieToTitle(j)
    - Inner join the shrinked rating table and the movie table by matching common movieId

WordCloud: Shrinked Dataset Exploration



User-User Collaborative Filtering

- **Collaborative filtering** based systems use the actions of users to recommend other items.
- **User-User Collaborative Filtering** uses that logic and recommends items by finding similar users to the active user to whom we are trying to recommend a movie.

User-user Collaborative Filtering

$$s(i,j) = \bar{r}_i + \frac{\sum_{i' \in \Omega_j} w_{ii'} \{ r_{i'j} - \bar{r}_{i'} \}}{\sum_{i' \in \Omega_j} |w_{ii'}|}$$

- The score for user i and movie j can be expressed by 2 parts
    - User i's own average rating
    - Weighted average deviation for movie j

Using Pearson Correlation Coefficient

$$w_{ii'} = \frac{\sum_{j \in \Psi_{ii'}} (r_{ij} - \bar{r}_i)(r_{i'j} - \bar{r}_{i'})}{\sqrt{\sum_{j \in \Psi_i} (r_{ij} - \bar{r}_i)^2} \sqrt{\sum_{j \in \Psi_{i'}} (r_{i'j} - \bar{r}_{i'})^2}}$$

$\Psi_i = \textit{set of movies that user } i \textit{ has rated}$

$\Psi_{ii'} = \textit{set of movies both user } i \textit{ and } i' \textit{ have rated}$ **Threshold >=5**

$\Psi_{ii'} = \Psi_i \cap \Psi_{i'}$

Neighborhood

- In practice, we don't sum over all users who rated movie j, it takes too long to run the code
- We only want to sum over the ones with highest weights
    - We just keep track of k most similar users to each user (K nearest neighbors)
    - We use an ordered list to achieve that, only the K highest weights can be maintained in the list

Item-item collaborative filtering

- Similar to user-user collaborative filtering
- Difference between user-user and item-item
  - User-user CF: choose movies for a user, because those movies have been liked by similar users
  - Item-item CF: choose items for a user, because this user has liked similar items in the past
- Item-Item CF runs much faster: $O(M^2 \times N)$
  - There are M^2 item-item weights, and each vector is length N
  - $N \gg M$

Results

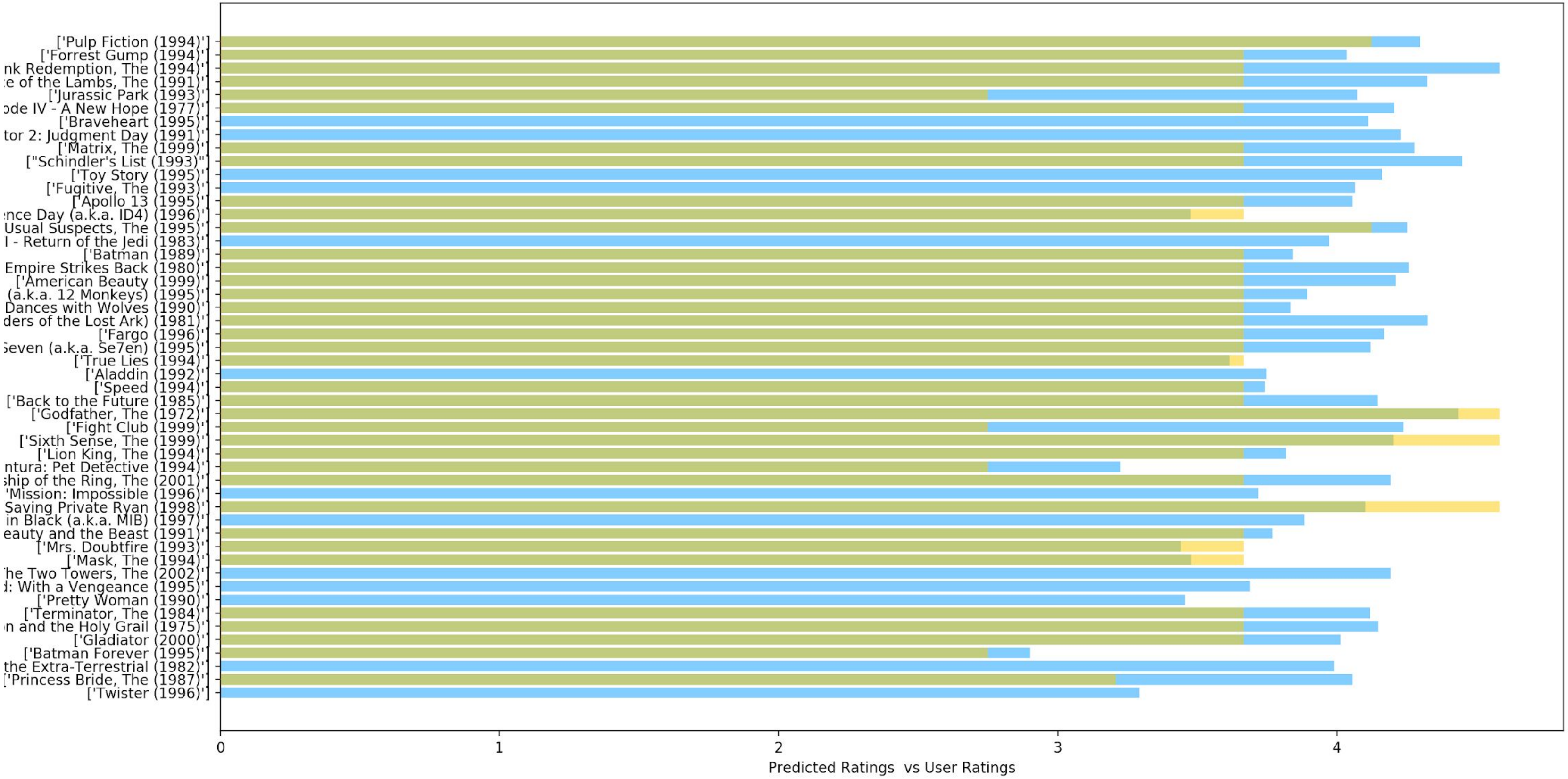
```
Actual
movie: ['Pulp Fiction (1994)'] : 4.5
movie: ['Forrest Gump (1994)'] : 4.0
movie: ['Shawshank Redemption, The (1994)'] : 4.0
movie: ['Silence of the Lambs, The (1991)'] : 4.0
movie: ['Jurassic Park (1993)'] : 3.0
movie: ['Star Wars: Episode IV - A New Hope (1977)'] : 4.0
movie: ['Matrix, The (1999)'] : 4.0
movie: ["Schindler's List (1993)"] : 4.0
movie: ['Apollo 13 (1995)'] : 4.0
movie: ['Independence Day (a.k.a. ID4) (1996)'] : 4.0
movie: ['Usual Suspects, The (1995)'] : 4.5
movie: ['Batman (1989)'] : 4.0
movie: ['Star Wars: Episode V - The Empire Strikes Back (1980)'] : 4.0
movie: ['American Beauty (1999)'] : 4.0
movie: ['Twelve Monkeys (a.k.a. 12 Monkeys) (1995)'] : 4.0
movie: ['Dances with Wolves (1990)'] : 4.0
movie: ['Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)'] : 4.0
movie: ['Fargo (1996)'] : 4.0
movie: ['Seven (a.k.a. Se7en) (1995)'] : 4.0
movie: ['True Lies (1994)'] : 4.0
movie: ['Speed (1994)'] : 4.0
movie: ['Back to the Future (1985)'] : 4.0
movie: ['Godfather, The (1972)'] : 5.0
movie: ['Fight Club (1999)'] : 3.0
movie: ['Sixth Sense, The (1999)'] : 5.0
movie: ['Lion King, The (1994)'] : 4.0
movie: ['Ace Ventura: Pet Detective (1994)'] : 3.0
movie: ['Lord of the Rings: The Fellowship of the Ring, The (2001)'] : 4.0
movie: ['Saving Private Ryan (1998)'] : 5.0
movie: ['Beauty and the Beast (1991)'] : 4.0
movie: ['Mrs. Doubtfire (1993)'] : 4.0
movie: ['Mask, The (1994)'] : 4.0
movie: ['Terminator, The (1984)'] : 4.0
movie: ['Monty Python and the Holy Grail (1975)'] : 4.0
movie: ['Gladiator (2000)'] : 4.0
movie: ['Batman Forever (1995)'] : 3.0
movie: ['Princess Bride, The (1987)'] : 3.5
```

User-based vs Item-based

```
Predicted:
movie: ['Pulp Fiction (1994)'] : 4.73
movie: ['Forrest Gump (1994)'] : 4.04
movie: ['Shawshank Redemption, The (1994)'] : 4.5
movie: ['Silence of the Lambs, The (1991)'] : 4.56
movie: ['Jurassic Park (1993)'] : 3.53
movie: ['Star Wars: Episode IV - A New Hope (1977)'] : 4.52
movie: ['Braveheart (1995)'] : 4.1
movie: ['Terminator 2: Judgment Day (1991)'] : 4.17
movie: ['Matrix, The (1999)'] : 4.11
movie: ["Schindler's List (1993)"] : 4.65
movie: ['Toy Story (1995)'] : 4.19
movie: ['Fugitive, The (1993)'] : 3.99
movie: ['Apollo 13 (1995)'] : 3.87
movie: ['Independence Day (a.k.a. ID4) (1996)'] : 3.61
movie: ['Usual Suspects, The (1995)'] : 4.41
movie: ['Star Wars: Episode VI - Return of the Jedi (1983)'] : 4.08
movie: ['Batman (1989)'] : 3.66
movie: ['Star Wars: Episode V - The Empire Strikes Back (1980)'] : 4.34
movie: ['American Beauty (1999)'] : 4.2
movie: ['Twelve Monkeys (a.k.a. 12 Monkeys) (1995)'] : 3.85
movie: ['Dances with Wolves (1990)'] : 3.98
movie: ['Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)'] : 4.5
movie: ['Fargo (1996)'] : 4.45
movie: ['Seven (a.k.a. Se7en) (1995)'] : 4.17
movie: ['True Lies (1994)'] : 3.62
movie: ['Aladdin (1992)'] : 3.69
movie: ['Speed (1994)'] : 3.71
movie: ['Back to the Future (1985)'] : 4.02
movie: ['Godfather, The (1972)'] : 4.92
movie: ['Fight Club (1999)'] : 3.23
movie: ['Sixth Sense, The (1999)'] : 4.45
movie: ['Lion King, The (1994)'] : 3.94
movie: ['Ace Ventura: Pet Detective (1994)'] : 2.16
movie: ['Lord of the Rings: The Fellowship of the Ring, The (2001)'] : 4.34
movie: ['Mission: Impossible (1996)'] : 3.5
movie: ['Saving Private Ryan (1998)'] : 4.65
movie: ['Men in Black (a.k.a. MIB) (1997)'] : 3.8
movie: ['Beauty and the Beast (1991)'] : 4.08
movie: ['Mrs. Doubtfire (1993)'] : 3.8
movie: ['Mask, The (1994)'] : 3.51
movie: ['Lord of the Rings: The Two Towers, The (2002)'] : 4.25
movie: ['Die Hard: With a Vengeance (1995)'] : 3.6
movie: ['Pretty Woman (1990)'] : 3.48
movie: ['Terminator, The (1984)'] : 4.14
movie: ['Monty Python and the Holy Grail (1975)'] : 4.31
movie: ['Gladiator (2000)'] : 4.03
movie: ['Batman Forever (1995)'] : 2.14
movie: ['E.T. the Extra-Terrestrial (1982)'] : 4.32
movie: ['Princess Bride, The (1987)'] : 3.83
movie: ['Twister (1996)'] : 3.36
```

```
Predicted:
movie: ['Pulp Fiction (1994)'] : 4.3
movie: ['Forrest Gump (1994)'] : 4.04
movie: ['Shawshank Redemption, The (1994)'] : 4.58
movie: ['Silence of the Lambs, The (1991)'] : 4.32
movie: ['Jurassic Park (1993)'] : 4.07
movie: ['Star Wars: Episode IV - A New Hope (1977)'] : 4.21
movie: ['Braveheart (1995)'] : 4.11
movie: ['Terminator 2: Judgment Day (1991)'] : 4.23
movie: ['Matrix, The (1999)'] : 4.28
movie: ["Schindler's List (1993)"] : 4.45
movie: ['Toy Story (1995)'] : 4.16
movie: ['Fugitive, The (1993)'] : 4.07
movie: ['Apollo 13 (1995)'] : 4.06
movie: ['Independence Day (a.k.a. ID4) (1996)'] : 3.47
movie: ['Usual Suspects, The (1995)'] : 4.25
movie: ['Star Wars: Episode VI - Return of the Jedi (1983)'] : 3.97
movie: ['Batman (1989)'] : 3.84
movie: ['Star Wars: Episode V - The Empire Strikes Back (1980)'] : 4.26
movie: ['American Beauty (1999)'] : 4.21
movie: ['Twelve Monkeys (a.k.a. 12 Monkeys) (1995)'] : 3.89
movie: ['Dances with Wolves (1990)'] : 3.83
movie: ['Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)'] : 4.32
movie: ['Fargo (1996)'] : 4.17
movie: ['Seven (a.k.a. Se7en) (1995)'] : 4.12
movie: ['True Lies (1994)'] : 3.62
movie: ['Aladdin (1992)'] : 3.75
movie: ['Speed (1994)'] : 3.74
movie: ['Back to the Future (1985)'] : 4.15
movie: ['Godfather, The (1972)'] : 4.43
movie: ['Fight Club (1999)'] : 4.24
movie: ['Sixth Sense, The (1999)'] : 4.2
movie: ['Lion King, The (1994)'] : 3.82
movie: ['Ace Ventura: Pet Detective (1994)'] : 3.23
movie: ['Lord of the Rings: The Fellowship of the Ring, The (2001)'] : 4.19
movie: ['Mission: Impossible (1996)'] : 3.72
movie: ['Saving Private Ryan (1998)'] : 4.1
movie: ['Men in Black (a.k.a. MIB) (1997)'] : 3.88
movie: ['Beauty and the Beast (1991)'] : 3.77
movie: ['Mrs. Doubtfire (1993)'] : 3.44
movie: ['Mask, The (1994)'] : 3.48
movie: ['Lord of the Rings: The Two Towers, The (2002)'] : 4.19
movie: ['Die Hard: With a Vengeance (1995)'] : 3.69
movie: ['Pretty Woman (1990)'] : 3.46
movie: ['Terminator, The (1984)'] : 4.12
movie: ['Monty Python and the Holy Grail (1975)'] : 4.15
movie: ['Gladiator (2000)'] : 4.01
movie: ['Batman Forever (1995)'] : 2.9
movie: ['E.T. the Extra-Terrestrial (1982)'] : 3.99
movie: ['Princess Bride, The (1987)'] : 4.05
movie: ['Twister (1996)'] : 3.29
```

Item-based Recommendation System



Evaluations:

- This is essentially a regression problem, we use mean square error(MSE) as our evaluation method

$$MSE = \frac{1}{|\Omega|} \sum_{i,j \in \Omega} (r_{ij} - \hat{r}_{ij})^2$$

Ω = Set of pairs (i,j) where user i has rated movie j

- User-user collaborative filtering: MSE = **0.602**
- Item-item collaborative filtering: MSE = **0.578**

Future Works

- Cold-Start Problem, if we do not have enough data, there is no way for us to calculate the correlations
- we could using Bayesian approach by putting a prior to the average
- Optimize the time complexity of the algorithm

Thank You!