SI650 Project Update

Yu Xia
xiayuu@umich.edu

1. Introduction

For beer lover, drinking difference kinds of beer has been one of the most relaxing activities. However, not all beer lovers are experts in beer and the names of the beer usually do not tell much information. Hence, every time when they would like to try a new flavor of beer, they get stuck on the all those different names of beer and have no clue which one to choose. Searching them one by one online before making a decision would result in much inconvenience. What if there is a system that can recommend beer? Motivated by the above consideration, I would like to build a beer recommender system based on collaborative filtering methods.

2. Data

Data Source

The dataset "BeerAdvocate" is obtained from Dr. Julian McAuley's ICDM 2012 paper, Learning Attitudes and Attributes from Multi-aspect Reviews. The link to the dataset is https://cseweb.ucsd.edu/~jmcauley/datasets.html#multi_aspect.

The dataset holds 1586615 interaction records between 33388 users and 66055 beers. Specifically, each record of the dataset contains the name, brewer and other information of a beer as well as the ratings given by a user. A sample record of the original dataset is shown below:

brewery_id, brewery_name, review_time, review_overall, review_aroma, review_appearance, review_profilename, beer_style, review_palate, review_taste, beer_name, beer_abv, beer_beerid

10325, Vecchio Birraio, 1234817823, 1.5, 2, 2.5, stcules, Hefeweizen, 1.5, 1.5, Sausa Weizen, 5, 47986

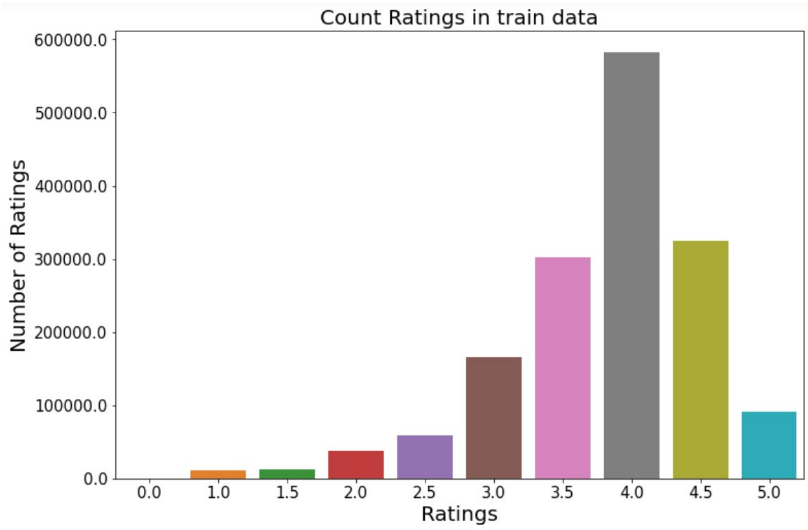
From all the fields, I will use only three of them, *review_profilename, beer_name* and *review_overall* to build my recommender system.

Preprocessing Steps
To improve the usability of the data, I make the following preprocessing steps.

1. Mapp the user names and beer names into integer indexes for simplicity and efficiency of the model.

2. Identify and drop duplicate and null records and alter the data type of rating from string to float.

3. Visualize the distribution of the ratings across the entire dataset



4. Collect statistics of how many ratings each user makes and how many ratings each beer has.

```
ratings_per_user = df.groupby("
ratings_per_user.describe()

count    33388.000000
mean        47.520427
std        182.604079
min          1.000000
25%          1.000000
50%          3.000000
75%         16.000000
max       5817.000000
Name: rating, dtype: float64
```

```
ratings_per_item = df.groupby("
ratings_per_item.describe()

count    66055.000000
mean        24.019559
std        110.864137
min          1.000000
25%          1.000000
50%          2.000000
75%          7.000000
max       3290.000000
Name: rating, dtype: float64
```

5. Filter out users that make less than 30 ratings and beers that has less than 15 ratings, which reduces the size of users to 6150 and the size of items to 10115.

6. Split the data into train and test set with a proportion of 80% train data and 20% test data. Since the test dataset contains the user ratings, no extra annotation is needed.

3. Related Work

Beer recommendation has been a topic studied by a few articles. Huitzil [3] proposed a beer recommender called GimmeHop using ontology reasoners and fuzzy logic. Allen [2] developed a beer recommender called BeerMe based on content-based filtering. Chinchanachokchai [1] built the system using review data from existing online community to test the hypotheses.

Collaborative filtering methods have been widely applied to recommender system in order to make better rating prediction. Resnick [5] first introduced user-based collaborative filtering which considers similar user sharing similar preferences. Sarwar [4] first introduced item-based collaborative filtering which considers item-item similarities when making predictions. Koren [6] first devised matrix factorization method to collaborative filtering recommender system.

4. Methodology

The beer recommender system will be built on collaborative filtering methods. There are generally two approaches of collaborative filtering methods, memory-based and model-based. Memory-based approach uses user rating data to compute the similarities between users or between items, which can also be classified as user-based and item-based collaborative filtering method. Model-based approach uses machine learning algorithm like matrix factorization to compute the ratings of unrated items.

In this project, I would explore methods in both approaches and evaluate their performances on beer recommendation scenario. Specifically, user-based and item-based K-nearest neighbor algorithm will be applied and being compared. Furthermore, I would try matrix factorization-based algorithm to evaluate whether machine learning method would help beer recommendation.

5. Evaluation and Results

The performances on test data of different approaches of collaborative filtering method will be measured by Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE).

There are two types of baselines. The first one is a naive baseline, which predicts a random rating based on the distribution of the training set. The second type of baseline would be item-based and user-based k-NN collaborative filtering algorithms. The current results of these baselines on the beer dataset are shown in the table below.

	RMSE	MAE
Random	0.9684	0.7618
Item-based k-NN	0.5843	0.4387
User-based k-NN	0.5953	0.4503

6. Work Plan

The tentative work plan is shown below, where I am currently working on step 4.

1) data preprocessing – week1
2) recommender system implementation – week 2-3
3) baseline algorithms evaluation – week 4
4) new method evaluation and discussion – week 5-6
5) conclusion and project report writing – week 7

References

[1] Chinchanachokchai, Sydney, Pipat Thontirawong, and Punjaporn Chinchanachokchai. "A tale of two recommender systems: The moderating role of consumer expertise on artificial intelligence based product recommendations." Journal of Retailing and Consumer Services 61 (2021): 102528.

[2] Allen, Alessandro, and Ryan Wetherbee. "BeerMe: A Beer Recommendation System."

[3] Huitzil, Ignacio, Fernando Alegre, and Fernando Bobillo. "GimmeHop: A recommender system for mobile devices using ontology reasoners and fuzzy logic." Fuzzy Sets and Systems 401 (2020): 55-77.

[4] Sarwar, Badrul, et al. "Item-based collaborative filtering recommendation algorithms." Proceedings of the 10th international conference on World Wide Web. 2001.

[5] Resnick, Paul, et al. "Grouplens: An open architecture for collaborative filtering of netnews." Proceedings of the 1994 ACM conference on Computer supported cooperative work. 1994.

[6] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42.8 (2009): 30-37.