Wan D. Bae - Data Science (ITE4005)
컴퓨터전공 2013012289 한기훈

12 June 2017

Report for Assignment4 - Recommender

1. Summary of algorithm.

The recommender algorithm I implemented follows the following procedure.

- Read training data and test data from files.
- Generate similarity list between each users.
- Predict rating of given items.
- Write into output files.

2. Detailed description of code.

This code has three major parts.

A. readData()

Read training data and test data from given path. Each line contains user_id, item_id, rating, timestamp. I abandon timestamp because I think timestamp is useless in predicting process.

In this function, I define

```
map<int, map<int, int> > data
```

and store training data and test data into that data structure.

For training data and test data, function is called respectively.

B. processPrediction()

In this function, it follows the following procedure.

- Read entire training data and test data to determine maximum user_id and maximum item_id. These values will be used to construct rating table and rating value empty check table(2D array).
- Then, get training data and assign rating value into previously constructed 2D array.

- Construct ‘`user_avg_ratings`’ 1D array which stores average rating value of each users and store corresponding values.
- Construct ‘`map<int, vector< pair<double, int> > > similarity_list`’. Key is user_id and value is a list of similarity. In list, each object contains similarity value and corresponding other user_id.
- Get similarity value of different 2 users. If value is not NaN, store it into previously defined map.
- After we get similarity list, proceed to get predicted ratings.
- Each prediction for user_id and item_id, get predicted rating value. If rating value is smaller than 1, assign 1. Else if rating value is bigger than 5, assign 5. Else, assign predicted rating value.

C. `printResult()`

Print each user_id, item_id, predicted_rate (with rounded to integer) in one line.

3. How to compile this code.

Just type ‘make’ in terminal. Or, please type below line.

```
$ g++ -O2 -o recommender.exe main.cpp --std=c++11
```

Above task will generate ‘recommender.exe’ file into same directory.

Compiler must support C++11 standard.

4. Any other specification.

Using sample data, test program generates following results (RMSE).

- u1 : 0.8462269
- u2 : 0.8396428
- u3 : 0.8397023
- u4 : 0.8387789
- u5 : 0.84