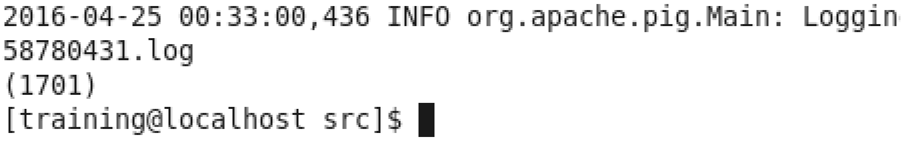
CSE427 Milestone2 Report

Zuyan Jiang(446225)
Yan Tong(444184)
Ren Zhou(445125)

A movie recommendation system is used for predicting movie ratings of the specific users. The recommendation system is based on a item-item collaborative filtering algorithm implemented in Hadoop MapReduce and Pig. We will find the similarity between the movies using Pearson-Correlation similarity, and predicting the ratings a user will give to a movie. The training data set contains 3.25 million ratings provided by Netflix.

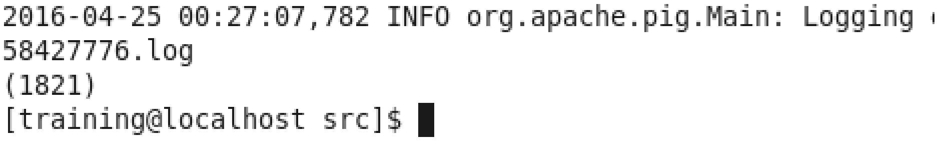
1.

In this problem, our goal is to compute total number of items and users in TesingRatings.txt and TreainingRatins.txt.
Since we only need to compute the number of items, we choose to use pig system.
At first, we compute the total numbers of items. We begin with loading the data into the pig system. Accordingly, the fields are movie_id, user_id and ratings. Then we group the tuples by movie_id. And then count the total numbers of tuples by using COUNT operation. The output will be the total number of items. As shows in Figure 1.1 and Figure 1.2.
The second step is to compute the total numbers of users. In this part, we group the tuples by User_id. And counting the total number of tuples by using COUNT operation. Then the output will be the total number of items. As shown in Figure 1.3 and Figure 1.4.



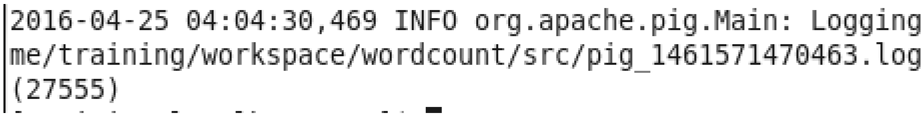
```
2016-04-25 00:33:00,436 INFO org.apache.pig.Main: Loggin
58780431.log
(1701)
[training@localhost src]$ █
```

Figure 1.1 Total numer of movies in TestingRatings.txt



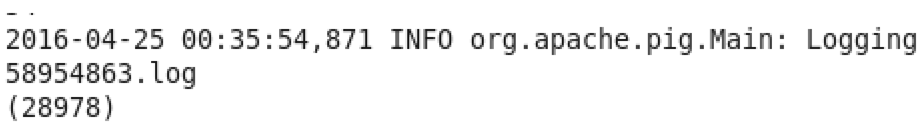
```
2016-04-25 00:27:07,782 INFO org.apache.pig.Main: Logging
58427776.log
(1821)
[training@localhost src]$ █
```

Figure 1.2 The number of movies in TrainingRatings.txt



```
2016-04-25 04:04:30,469 INFO org.apache.pig.Main: Logging
me/training/workspace/wordcount/src/pig_1461571470463.log
(27555)
```

Figure 1.3 Total number of users in TestingRatings.txt



```
2016-04-25 00:35:54,871 INFO org.apache.pig.Main: Logging
58954863.log
(28978)
```

Figure 1.4 Total number of user in TrainingRatings.txt

2.

In the user based approaches, let’s use the expected space complexity of correlation similarity to evaluate the performance of the ‘user to user’ and ‘item to item’ model.

Firstly, let’s recall the computation process:

The formula of the ‘correlation similarity’ is shown as below:

$$P(a,u) = \frac{\sum_{i\in Sa\cap Su}(Vai - \overline{Vai}) \times (Vui - \overline{Vui})}{\sqrt{\sum_{i\in Sa\cap Su}(Vai - \overline{Vai})^2 \sum_{i\in Sa\cap Su}(Vui - \overline{Vui})^2}}$$

Where U is a set of N users and I is a set of M items. Vui denotes the rating of user u ∈ U on item $i \in I$, and $Su\ \in I$ stands for the set of items that user u has rated.

And then, the neighborhood size K we have choosen is 60, so the space complexity is $O(N \times K)$.

And in the subproblem a, we have computed the expected N, it is

Hence, the expected space complexity in TestingRatings.txt is about

$$27555 \times 60 = 1{,}653{,}300$$

And the expected space complexity in TrainingRatings.txt is about

$$28978 \times 60 = 1{,}738{,}680$$

The item-based approaches predict the rating of a given user on a giver item using the ratings of the user on the items considered as similar to the target item.

The Pearson correlation formula as shows below, which corresponds to the cosine of items deviation from the user mean rating:

And then, the neighborhood size K we have choosen is 60, so the space complexity

$$P(i,j)=\frac{\sum_{\{u\in U\ /\ i\in S_u \& j\in S_u\}}(v_{ui}-\overline{v_u})(v_{uj}-\overline{v_u})}{\sqrt{\sum_{\{u\in U\ /\ i\in S_u \& j\in S_u\}}(v_{ui}-\overline{v_u})^2 \sum_{\{u\in U\ /\ i\in S_u \& j\in S_u\}}(v_{uj}-\overline{v_u})^2}}$$

is $O(M \times K)$.

And in the subproblem a, we have computed the expected N, it is

Hence, the expected space complexity in TestingRatings.txt is about

$$1701 \times 60 = 102{,}060$$

And the expected space complexity in TrainingRatings.txt is about

$$1821 \times 60 = 109{,}260$$

3.

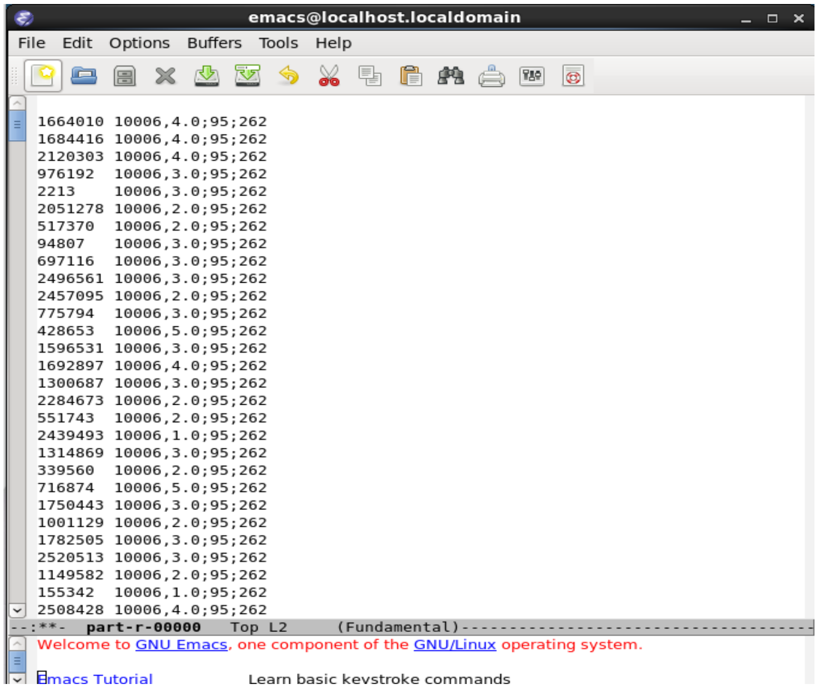
As result of the problem (c), for user-user based, the space complexity is O(NK), and the space complexity is O(MK). (In TestingRatings.txt, they are 1,653,300 and 102,060 respectively and In TrainingRatings.txt 1,738,680 and 109,260). The number of movies are increasing, but not as rapidly as the number of users. In addition, given that the total number of users outnumber the number of movies, we can get O(NK) > O(MK). Therefore,item based provides better quality than user based at all sparsity levels we may focus on scalability.

4.

The preprocessing MapReduce job does two things: a. Get number of rated items per user x, numRatings. B. Compute sum of all rating of user x, sumRatings.
They are needed for Jaccard and correlation to compute average rx=sumRatings/numRatings.

Generated data using Mapper with following format: (_, (Movie_id, User_id, Rating))
The output are files in which each line would be like: (Movie_id, (User_id,Rating)

Generated data using Reducer with following format: (MovieId, list of (User_id,Rating))
Generated data using Reducer with following format:
(UserId, Movie_id,Rating;num_ratings;sum_ratings)


emacs@localhost.localdomain
File Edit Options Buffers Tools Help

```
1664010 10006,4.0;95;262
1684416 10006,4.0;95;262
2120303 10006,4.0;95;262
976192  10006,3.0;95;262
2213    10006,3.0;95;262
2051278 10006,2.0;95;262
517370  10006,2.0;95;262
94807   10006,3.0;95;262
697116  10006,3.0;95;262
2496561 10006,3.0;95;262
2457095 10006,2.0;95;262
775794  10006,3.0;95;262
428653  10006,5.0;95;262
1596531 10006,3.0;95;262
1692897 10006,4.0;95;262
1300687 10006,3.0;95;262
2284673 10006,2.0;95;262
551743  10006,2.0;95;262
2439493 10006,1.0;95;262
1314869 10006,3.0;95;262
339560  10006,2.0;95;262
716874  10006,5.0;95;262
1750443 10006,3.0;95;262
1001129 10006,2.0;95;262
1782505 10006,3.0;95;262
2520513 10006,3.0;95;262
1149582 10006,2.0;95;262
155342  10006,1.0;95;262
2508428 10006,4.0;95;262
--:**-  part-r-00000   Top L2     (Fundamental)------------------------------------
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

Emacs Tutorial               Learn basic keystroke commands
```

In summary, the result shows that the prediction we have made in the milestone 1 is correct. Because, the item based approach has obviously small space comeplexity.