

CEG-7380

Cloud Computing

Spring 2016

Project #3

Report File

DHRUVKUMAR NAVINCHANDRA PATEL

U00791652

Finding Common Friends in MapReduce

Following are the different steps which I followed during development of this project

- 1) **Analysis:** In social networking contains huge data because of higher number of users such as Facebook. So idea is to find common (mutual) friends between two users. So, MapReduce is the ideal solution to improve the time computation of the request. Here I implemented Hadoop MapReduce way to find common friends from two user's user ids and friends list.

- 1) **Hadoop MapReduce implementation :**

- i) In Hadoop MapReduce I am taking sample.txt file as an input. I developed this program in our virtual Cloud which has a version of Hadoop 1.2.1. Input file contains User id and their friendslist with FriendId. Here friend id basically the user id of one user. Every user has their unique user id.

Input Format Example:

<UseriDi><,><FriendID1><space><FriendID2>...<FriendIDm>

101,102 103 104 105 106

Here 101 is the user id and from 102 to 106 are the friends list of user 101.

- ii) **Mapper Implementation:** In Mapper I am taking data from input file and pass to the mapper to generate key and value pair from mapper.
 - a) In mapper generate key and value as Text.
 - b) Here I split the data by “,” and generate userid pairs using user and their friendslist.
 - c) So Output from mapper is userid pair as key and friends list as value

For example, following input

101,102 103 104 105 106

Mapper output

Key	value
-----	-------

101,102	102 103 104 105 106
---------	---------------------

101,103	102 103 104 105 106 and so on.
---------	--------------------------------

Here I used compareTo method to compare to user ids so 101,102 and 102,101 works as same Key.

- iii) **Reducer implementation:** In reducer I am taking the key and value from mapper and generate output in key as Text and value as Text

- a) In reducer for each userid pair contains their friends list.
- b) I just created String array that contains two values and send those two string values in another class method called `Findcommonfriends.common (s1, s2)` to find common friends between s1 and s2. Output of that method become the final output of reducer.
- c) In Hadoop 1.2.1 version there is no `TextOutputFormat.SEPERATOR` so in my output there is a space between key and value.

Output from Virtual Cloud:

Which contains space between output key and value pair

```
<UseriDi><,><UserIDj><space><[><FriendID1><,><FriendID2>...<FriendIDm><]>
```

```
101,102 [103,104]
```

Basically output display common (mutual friends) between user 101 and 102. So [103,104] are common friends of 101 and 102.

- d) In Hadoop 2.6.0 version there is `TextOutputFormat.SEPERATOR` so I removed space between key and value and I am getting output exactly what you want.

Output is following format:

```
101,102[103,104]
```

- iv) **Driver class** – It contains all the required parameter to execute Map-Reduce Program.
- v) **Findcommonfriends Class** – This class is used to find common friendIds between two friends id list. This class contains one static method named `common` which basically takes two String and split string by space so I am getting two arrays and then apply Brute-force search approach to find common between two arrays. And collect common friends in this [friends list] format and return when this method called from reducer class.