

# MapReduce FindCommonFriends

**Definition:** < person> , < friend1>< friend2 >...< friendn >

friend1和friend2的共同好友不一定含有person，该list只能说明person作为friend的好友出现，因此要首先对给出的friend list进行反向处理，保证value组中的两两组合共同好友即为key对应的用户。

## 基本数据类型实现

即利用int及IntWritable类型读入用户名

- MapReduce job1实现

- Map：读入数据

eg. 100, 200 300 400 500 600

200, 100 300 400

<key,value>:<200, 100> <300, 100><400, 100>...

- 此时的键值对含义为：key值用户的好友中都有value值用户

- Reduce

- 通过查看样例发现，**好友关系是单向的**（即100的好友列表中没有600，但600的好友列表中有100，此时100和600的任意一好友共同好友则不能为600），因此reduce阶段就是将题目中的好友列表反向读入，保证value中用户的好友中都有key中的用户

eg. 100, 200 300 400 500 600

200, 100 300 400

<key,value>: <300, 100> <300, 200> 在reduce中规约为 300 100,200

通过第一步mapreduce实现用户列表反转之后，才可以保证遍历value中的两两用户，其共同好友为key对应的用户

遇见问题：最开始将第一个person以intWritable读入，通过报错提醒修改为longWritable（但是后来还是报错说不能将LongWritable转为Text，于是索性直接改为Text进行了hhh

```
Error: java.io.IOException: Type mismatch in key from map: expected org.apache.hadoop.io.IntWritable, received org.apache.hadoop.io.LongWritable
    at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.collect(MapTask.java:1072)
    at org.apache.hadoop.mapred.MapTask$NewOutputCollector.write(MapTask.java:715)
    at org.apache.hadoop.mapreduce.task.TaskInputOutputContextImpl.write(TaskInputOutputContextImpl.java:89)
    at org.apache.hadoop.mapreduce.lib.map.WrappedMapper$Context.write(WrappedMapper.java:112)
    at org.apache.hadoop.mapreduce.Mapper.map(Mapper.java:125)
    at org.apache.hadoop.mapreduce.Mapper.run(Mapper.java:146)
    at org.apache.hadoop.mapred.MapTask.runNewMapper(MapTask.java:787)
    at org.apache.hadoop.mapred.MapTask.run(MapTask.java:341)
    at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:164)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1657)
    at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:158)
```

- 第一个mapreduce之后的返回结果

```
600,500,400,200,300,100,
100 200,400,300,600,500,
200 400,100,300,
300 400,200,100,500,
400 100,200,300,
500 100,300,
~
```

- 发现第一行的key居然是空的

- 解决方案：出现一个空值对应全部person，查阅input文件发现是因为分割符号“，”后面还有一个空格。。如果不在split中处理的话会在mapper阶段读入n
  - ull作为friend
- friendlist最后会有一个逗号
  - 解决方案：在reduce中先对返回的字符串处理一下，去掉最后一个字符
- 最终的mapreduce job 1 结果：

```
100      600,500,300,400,200
200      400,100,300
300      400,200,100,500
400      100,300,200
500      300,100
```

- MapReduce job2实现
  - Map：读入上一步的reduce数据 在value中形成两两组合的用户对并输出其共同好友  
key: [200,300] value: 100, 400
  - Reduce：合并value  
key: [200,300] value: [100, 400]
- 遇到一个报错：

```
20/11/02 07:09:52 INFO mapreduce.Job: Task Id : attempt_1604284178194_0009_m_000000_1, Status : FAILED
Error: java.lang.ClassCastException: org.apache.hadoop.io.Text cannot be cast to org.apache.hadoop.io.LongWritable
    at hyy.nju.edu.cn.FindFriend$CommonRegMapper.map(FindFriend.java:64)
    at org.apache.hadoop.mapreduce.Mapper.run(Mapper.java:146)
    at org.apache.hadoop.mapred.MapTask.runNewMapper(MapTask.java:787)
    at org.apache.hadoop.mapred.MapTask.run(MapTask.java:341)
    at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:164)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1657)
    at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:158)
```

- 找了很久代码似乎都没有问题。。后来在main函数中发现可能job1输出和job2输入都设定了sequencefileformat，于是猜测是此处有问题，上网查找发现如果不进行设定将默认以TextFormat作为输入输出格式，于是去掉setinput和outformat的设定之后重试居然好了。。目前还不知道是什么原因
- 最终结果见output/part-r-00000