《大规模分布式系统》实验报告

——MR 程序

姓名：刘佰川 专业：计算机科学与技术（数据科学方向）学号：16307130214

0．实验环境

VMware+Ubuntu18.04+Hadoop2.9.1+Java1.8.0_201

1. 实验要求

I. 对 sample.txt 文件统计其中各类文件的数量（按文件名后缀区分类型）

II. 对 sample.txt 按文件的字节数大小降序排序输出文件名

2. 实验过程

I. 函数编写

分别编写两个任务的 mapper 和 reducer 文件，主要的问题在于 sample.txt 文件是一个杂乱无章的文本文件，处理文件文本在这个问题中很重要。

统计其中各类文件的数量的 mapper 和 reducer

```
#!/usr/bin/python
import sys

for line in sys.stdin:
    line = line.split(" ")
    file_name = line[len(line)-1].strip()
    suffix = file_name.split(".")[1]
    print('%s\t%s'%(suffix,1))
~
~
~
~
```

```
#!/usr/bin/python
import sys

current_count = 0
current_word = None

for line in sys.stdin:
    line = line.strip()
    word,count = line.split('\t',1)
    count = int(count)
    if current_word == word:
        current_count += count
    else:
        if current_word:
            print("%s\t%s"%(current_word,current_count))

        current_count = count
        current_word = word
print("%s\t%s"%(current_word,current_count))
~
~
```

用管道进行测试

```
junjin@ubuntu:~$ cat /home/junjin/sample.txt | ./mapper.py | sort -r | ./reduce
r.py
pdf     43
jpg     9
dwg     32
docx    1
```

对文件大小进行排序的 mapper 和 reducer

```
#!/usr/bin/python
import sys

for line in sys.stdin:
    line = line.strip()
    text = line.split(" ")
    file_name = text[len(text)-1]
    if text[len(text)-2]:
        size = text[len(text)-3]
    else:size = text[len(text)-2]
    size = int(size.replace(',',''))
    print("%s\t%s"%(int(size),file_name))
~
~
~
```

```
#!/usr/bin/python
import sys
record = []
for line in sys.stdin:
    line = line.strip()
    size,name = line.split('\t',1)
    size = int(size)
    record.append((size,name))
sort_record = sorted(record)
for i in range(len(sort_record)-1,-1,-1):
    print("%s\t%s"%sort_record[i])

~
~
~
```

用管道进行测试

```
junjin@ubuntu:~$ cat /home/junjin/sample.txt | ./mapper1.py | sort -r | ./reduc
er1.py
5272668 ◆◆◆◆-◆◆ŋ00-02.pdf
5203305 ◆◆◆◆-◆◆ŋ00-01.pdf
4520750 ◆◆◆◆Ļé◆◆C.dwg
3395145 h◆◆5◆◆C.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆h◆◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆壩.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆ģ◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆◆◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆l◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆◆◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆◆◆.dwg
3389704 ◆◆◆◆Ŧ◆◆C◆◆◆ˆ◆.dwg
2605370 00-04.pdf
2205340 00-05.pdf
2238600 00-06.pdf
2201666 1-1◆◆◆◆.dwg
2088121 00-11.pdf
1940285 00-12.pdf
1904441 00-08.pdf
1824144 00-09.pdf
1746362 00-13.pdf
1652761 00-03.pdf
1607088 N-A◆◆◆◆C.dwg
1607088 A-N◆◆◆◆C.dwg
1607088 21-15◆◆◆◆C.dwg
1607088 15-21◆◆◆◆C.dwg
1511442 00-10.pdf
```

II. 用 hadoop-streaming 运行

开启 hadoop

```
junjin@ubuntu:/usr/local/hadoop$ ./sbin/start-dfs.sh
```

创建一个新的输入文件夹，将 sample.txt 传入

```
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -mkdir input1
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -copyFromLocal /home/junjin/sa
mple.txt input1/
```

分别运行两个 mapper 和 reducer，分别在输出文件夹查看输出文件情况

```
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop jar /usr/local/hadoop/share/hadoo
p/tools/lib/hadoop-streaming-2.9.1.jar -mapper 'python mapper.py' -file /home/j
unjin/mapper.py -reducer 'python reducer.py' -file /home/junjin/reducer.py -inp
ut input1/* -output output4
```

```
unjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -ls output4
ound 2 items
rw-r--r--   1 junjin supergroup          0 2019-03-29 19:24 output4/_SUCCESS
rw-r--r--   1 junjin supergroup         27 2019-03-29 19:24 output4/part-00000
```

```
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -cat output4/part-00000
docx    1
dwg     32
jpg     9
pdf     43
```

```
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop jar /usr/local/hadoop/share/hadoo
p/tools/lib/hadoop-streaming-2.9.1.jar -mapper 'python mapper1.py' -file /home/
junjin/mapper1.py -reducer 'python reducer1.py' -file /home/junjin/reducer1.py
-input input1/* -output output5
```

```
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -ls output5
Found 2 items
-rw-r--r--   1 junjin supergroup          0 2019-03-29 19:28 output5/_SUCCESS
-rw-r--r--   1 junjin supergroup       2012 2019-03-29 19:28 output5/part-00000
junjin@ubuntu:/usr/local/hadoop/bin$ ./hadoop fs -cat output5/part-00000
5272668 ****-**fj00-02.pdf
5203305 ****-**fj00-01.pdf
4520750 ****Lé**C.dwg
3395145 h**5**C.dwg
3389704 ****T**C***h**.dwg
3389704 ****T**C***壩.dwg
3389704 ****T**C***g*.dwg
3389704 ****T**C******.dwg
3389704 ****T**C***T*.dwg
3389704 ****T**C******.dwg
3389704 ****T**C******.dwg
3389704 ****T**C***^*.dwg
2605370 00-04.pdf
```

由于第二件文件输出较多，故将两个文件从 HDFS 传送到本地，两个输出文件将命名为 count 和 sort 代表两个任务，与实验报告一起提交。