

实验2-MapReduce编程

221275037刘紫艺

任务一

设计思路

1.Mapper阶段

将第2列的数据作为 `date`，资金流入与流出量分别从第5列和第9列提取，若该列为空，则默认值为 `0.0`，否则将其解析为 `double` 类型。对于每一行数据，输出一个 `<date, amounts>` 键值对，其中 `date` 是日期，`amounts` 是一个包含资金流入量和流出量的字符串（格式为 `流入量,流出量`）

2.Reduce阶段

通过遍历每个 `value`，将其中的流入量和流出量进行累加。每一条记录的流入量和流出量会加到相应的累计变量 `totalInflow` 和 `totalOutflow` 中。

在计算完成后，输出该日期的总流入量和总流出量。

不足和可改进之处

1、不足：在 Reducer 中，每次都会进行 `totalInflow += Double.parseDouble(amounts[0])` 和 `totalOutflow += Double.parseDouble(amounts[1])` 操作。对于较大的数据集来说，可能会增加计算的复杂性。

2、改进：

- 并行处理：因为流入量和流出量之间的计算是独立的，可以考虑将它们分别交给不同的 Reducer 进行计算，减少计算的串行化过程。
- 合并计算：通过设置更合理的 `Partitioner`，将相同日期的数据聚集到同一个 Reducer 中，减少数据传输。

运行结果

```
root@h01:/usr/local/hadoop# ./bin/hadoop jar hadoop-task-1.0-SNAPSHOT.jar com.e
xample.DailyFundFlow /input/user_balance_table.csv /output/lab2
2024-11-04 07:25:03,431 INFO client.DefaultNoHARMFailoverProxyProvider: Connect
ing to ResourceManager at h01/172.18.0.2:8032
2024-11-04 07:25:04,927 WARN mapreduce.JobResourceUploader: Hadoop command-line
option parsing not performed. Implement the Tool interface and execute your ap
plication with ToolRunner to remedy this.
2024-11-04 07:25:05,006 INFO mapreduce.JobResourceUploader: Disabling Erasure C
oding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1730687134049_0002
2024-11-04 07:25:06,238 INFO input.FileInputFormat: Total input files to proces
s : 1
2024-11-04 07:25:07,977 INFO mapreduce.JobSubmitter: number of splits:2
2024-11-04 07:25:08,556 INFO mapreduce.JobSubmitter: Submitting tokens for job:
job_1730687134049_0002
2024-11-04 07:25:08,556 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-11-04 07:25:09,224 INFO conf.Configuration: resource-types.xml not found
2024-11-04 07:25:09,227 INFO resource.ResourceUtils: Unable to find 'resource-t
ypes.xml'.
2024-11-04 07:25:11,800 INFO impl.YarnClientImpl: Submitted application applica
tion_1730687134049_0002
2024-11-04 07:25:12,092 INFO mapreduce.Job: The url to track the job: http://h0
1:8088/proxy/application_1730687134049_0002/
2024-11-04 07:25:12,094 INFO mapreduce.Job: Running job: job_1730687134049_0002
2024-11-04 07:25:43,920 INFO mapreduce.Job: Job job_1730687134049_0002 running
in uber mode : false
2024-11-04 07:25:43,924 INFO mapreduce.Job: map 0% reduce 0%
2024-11-04 07:26:22,566 INFO mapreduce.Job: map 7% reduce 0%
2024-11-04 07:26:24,639 INFO mapreduce.Job: map 40% reduce 0%
2024-11-04 07:26:26,822 INFO mapreduce.Job: map 57% reduce 0%
2024-11-04 07:26:28,865 INFO mapreduce.Job: map 67% reduce 0%
2024-11-04 07:26:34,270 INFO mapreduce.Job: map 82% reduce 0%
2024-11-04 07:26:40,882 INFO mapreduce.Job: map 83% reduce 0%
2024-11-04 07:26:46,140 INFO mapreduce.Job: map 100% reduce 0%
2024-11-04 07:27:07,595 INFO mapreduce.Job: map 100% reduce 100%
```

```
root@h01:/usr/local/hadoop# ./bin/hadoop fs -ls /output/lab2
Found 2 items
-rw-r--r--  2 root supergroup          0 2024-11-04 07:27 /output/lab2/_SUCCESS
-rw-r--r--  2 root supergroup    14515 2024-11-04 07:27 /output/lab2/part-r-00000
root@h01:/usr/local/hadoop# ./bin/hadoop fs -cat /output/lab2/part-r-00000
20130701      3.2488348E7,5525022.0
20130702      2.903739E7,2554548.0
20130703      2.727077E7,5953867.0
20130704      1.8321185E7,6410729.0
20130705      1.1648749E7,2763587.0
20130706      3.6751272E7,1616635.0
20130707      8962232.0,3982735.0
20130708      5.7258266E7,8347729.0
20130709      2.6798941E7,3473059.0
20130710      3.0696506E7,2597169.0
20130711      4.4075197E7,3508800.0
20130712      3.4183904E7,8492573.0
20130713      1.5164717E7,3482829.0
20130714      2.2615303E7,2784107.0
20130715      4.8128555E7,1.3107943E7
20130716      5.0622847E7,1.1864981E7
20130717      2.9015682E7,1.0911513E7
20130718      2.4234505E7,1.1765356E7
20130719      3.3680124E7,9244769.0
20130720      2.0439079E7,4601143.0
20130721      2.1142394E7,2681331.0
20130722      4.0448896E7,1.9144267E7
20130723      5.8136147E7,2.4404051E7
20130724      4.8422518E7,3.6258592E7
20130725      5.7433418E7,3.8212836E7
20130726      4.4721817E7,3.9192369E7
20130727      1.7194451E7,1.5058893E7
20130728      3.6255382E7,7683211.0
```

Show 25 entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rw-r--r--	root	supergroup	0 B	Nov 04 15:27	2	128 MB	_SUCCESS	<input type="checkbox"/>
<input type="checkbox"/>	-rw-r--r--	root	supergroup	14.17 KB	Nov 04 15:27	2	128 MB	part-r-00000	<input type="checkbox"/>

任务二

设计思路

1.Mapper 部分

解析每条输入记录，将日期转换为星期几，将资金流入量和流出量存储为键值对 `<weekday, "流入量,流出量">`

2.Reducer 部分

对于同一 `weekday` 的所有记录，计算总流入量和总流出量，并计数记录数 `count`。

用总流入量和总流出量分别除以 `count` 计算平均值。

利用 `TreeMap` 按流入量从大到小的顺序排序，并在 `cleanup` 方法中将排序后的结果输出。

不足和可改进之处

1.不足：当前实现使用单一的 Reducer 进行汇总和排序，随着数据量增加，Reducer 的负载将会显著增加，可能无法充分利用集群的并行计算能力。

2.改进：

- **增加并行度**：通过自定义 `Partitioner` 类，基于特定日期或其他字段进行数据分区，分配给多个 Reducer。这样可以利用更多的并行处理能力，提升整体作业性能。
- **分区策略优化**：根据实际数据分布情况（如不同日期分布）设置合理的分区策略，均衡各个 Reducer 的工作负载。

运行结果

```
root@h01:/usr/local/hadoop# ./bin/hadoop fs -cat /output/lab2_2/part-r-00000
Tuesday 2.6358205886885247E8,1.9176914462295082E8
Monday 2.6030581E8,2.174638654918033E8
Wednesday 2.5416260783606556E8,1.946394465081967E8
Thursday 2.3642559403278688E8,1.764666748852459E8
Friday 1.9940792306557378E8,1.6646796019672132E8
Sunday 1.5591455193442622E8,1.3242720506557377E8
Saturday 1.4808806829508197E8,1.1286894208196722E8
```

/output/lab2_2Go!

Show25entriesSearch:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	root	supergroup	0 B	Nov 06 16:55	2	128 MB	_SUCCESS	
<input type="checkbox"/>	-rw-r--r--	root	supergroup	339 B	Nov 06 16:55	2	128 MB	part-r-00000	

任务三

设计思路

1、Mapper：

读取每行数据，判断用户是否为活跃状态，如果是，则将 <用户ID，日期> 输出到 Reducer。

2、Reducer：

对于每个用户 ID，计算唯一的活跃日期总数（活跃天数）。

将活跃天数和用户 ID 存入 `TreeMap`，按活跃天数降序排序。

在 `cleanup` 中输出排序后的结果。

不足和可改进之处

扩展性问题

1.不足：如果少数用户的活跃记录特别多，可能会造成某些 Mapper 或 Reducer 负载不均衡。

2.改进：可以在 Mapper 中进行预处理，通过 `Combiner` 类在 Map 端部分聚合，以减少传输到 Reducer 的数据量。

运行结果

```
root@h01:/usr/local/hadoop# ./bin/hdfs dfs -cat /output/lab2_3/part-r-00000
7629      384
11818     359
21723     334
19140     332
24378     315
26395     297
25147     295
27719     293
20515     291
5016      287
27751     285
25951     280
2521      277
13435     268
5284      262
4561      260
24259     257
7848      251
24474     249
7320      240
1431      236
3059      228
19796     226
1676      225
22005     224
21437     223
18468     220
```

Show 25 entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	root	supergroup	0 B	Nov 06 20:03	2	128 MB	_SUCCESS	
<input type="checkbox"/>	-rw-r--r--	root	supergroup	1.97 KB	Nov 06 20:03	2	128 MB	part-r-00000	

任务四

设计内容

使用mfd_day_share_interest.csv中的数据计算一周七天中每天平均的万份收益和七日年化收益，并按照万份收益的大小从大到小排序。

程序部分和任务二相类似，不再复述

运行结果

```
root@h01:/usr/local/hadoop# ./bin/hdfs dfs -cat /output/lab2_4/part-r-00000
Tuesday 1.3640,5.095
Monday  1.3635,5.101
Friday  1.3630,5.079
Wednesday 1.3543,5.089
Thursday  1.3527,5.084
Sunday   1.3414,5.069
Saturday  1.3413,5.074
```

/output/lab2_4

Go!

Show

25

entries

Search:

<input type="checkbox"/>	<div></div> Permission	<div></div> Owner	<div></div> Group	<div></div> Size	<div></div> Last Modified	<div></div> Replication	<div></div> Block Size	<div></div> Name	<div></div>
<input type="checkbox"/>	-rw-r--r--	root	supergroup	0 B	Nov 07 01:05	2	128 MB	_SUCCESS	<div></div>
<input type="checkbox"/>	-rw-r--r--	root	supergroup	148 B	Nov 07 01:05	2	128 MB	part-r-00000	<div></div>

结论

与任务二的结果相对比，发现两者的排序结果相同，说明每日的收益率与每日资金流入之间可能存在正相关的关系。