

数据倾斜问题

reduce 阶段卡在 99.99%不动；各种 container 报错 OOM（内存溢出）；读写数据量很大，超过其他正常 reduce

数据倾斜的原理

在进行 shuffle 的时候，必须将各个节点上相同的 key 拉取到某个节点上的一个 task 来进行处理，比如按照 key 进行聚合或 join 等操作。此时如果某个 key 对应的数据量特别大的话，就会发生数据倾斜

发生数据倾斜的原因在于 Task 的数据分配不均衡

分为两种情况：

- 1：数据本身就是倾斜的，数据中某种数据出现的次数过多；
- 2：分区规则导致这些相同的数据都分配给了同一个 Task，导致这个 Task 拿到了大量的数据，而其他 Task 数据量比较少，所以运行起来较慢

数据倾斜的处理

1.group by 产生数据倾斜

【解决方法】：

(1)开启 Map 端聚合参数设置（详见上文 Group By）。

(2)或者根据业务，合理调整分组维度

2. count(distinct)产生数据倾斜

【解决方法】：

(1)使用 sum...group by 代替。如 select a,sum(1) from (select a, b from t group by a,b) group by a; (详见上文 Count(Distinct)去重统计)

(2)在业务逻辑优化效果的不大情况下，有些时候是可以将倾斜的数据单独拿出来处理，最后 union 回去

3. 大表和小表 join 产生数据倾斜

【解决方法】： 使用 mapjoin 将小表加载到内存中。（详见上文小表 join 大表）

4. 空值产生数据倾斜

【解决方法】：

(1) id 为空的不参与关联：

```
select * from log a join users b
  on a.user_id is not null and a.user_id = b.user_id
union all
select * from log a where a.user_id is null;
```

(2) 给空值分配随机的 key 值：

```
select * from log a left outer join users b
  on case when a.user_id is null then
      concat('hive',rand() ) else a.user_id end = b.user_id;
```

一般分配随机 key 值得方法更好一些。

5. 合理设置 map 和 reduce 数避免数据倾斜

【调整 Map 数】

1.小文件进行合并，减少 map 数（默认开启）

在 map 执行前合并小文件，减少 map 数：

```
set hive.input.format=
org.apache.hadoop.hive ql.io.CombineHiveInputFormat;
```

注: `CombineHiveInputFormat` 具有对小文件进行合并的功能（系统默认的模式）。但是 `HiveInputFormat` 没有对小文件合并功能

2. 复杂文件增加 Map 数

当 `input` 的文件都很大，任务逻辑复杂，`map` 执行非常慢的时候，可以考虑增加 Map 数，来使得每个 `map` 处理的数据量减少，从而提高任务的执行效率

MapReduce 中没有办法直接控制 `map` 数量，可以通过设置每个 `map` 中处理的数据量进行设置。

#HDFS 默认数据块大小 128M、最小分片大小 1、最大分片大小 256M

```
set dfs.blocksize=134217728;
```

```
set mapreduce.input.fileinputformat.split.minsize=1;
```

```
set mapreduce.input.fileinputformat.split.maxsize=256000000;
```

增加 `map` 的方法为：根据公式，调整 `maxSize` 最大值：

```
computeSliteSize(Math.max(minSize,Math.min(maxSize,blocksize))=blocksize=128M
```

让 `maxSize` 最大值低于 `blocksize` 就可以增加 `map` 的个数

【调整 Reduce 数】

调整 `reduce` 个数方法一：

每个 `Reduce` 处理的数据量默认是 256MB

```
hive.exec.reducers.bytes.per.reducer=256000000
```

#每个任务最大的 `reduce` 数，默认为 1009

```
hive.exec.reducers.max=1009
```

#计算 `reducer` 数的公式：

```
N=min(hive.exec.reducers.max, 总输入数据  
量/hive.exec.reducers.bytes.per.reducer)
```

调整 `reduce` 个数方法二：

在 `hadoop` 的 `mapred-default.xml` 文件中修改设置每个 `job` 的 `Reduce` 个数

```
set mapreduce.job.reduces = 15;
```

