|  |  |  |
| --- | --- | --- |
| 属性名称 | 默认值 | 属性说明 |
| spark.reducer.maxSizeInFlight | 48m | reduce task的buffer缓冲，代表了每个reduce task每次能够拉取的map side数据最大大小，如果内存充足，可以考虑加大，从而减少网络传输次数，提升性能 |
| spark.shuffle.blockTransferService | netty | shuffle过程中，传输数据的方式，两种选项，netty或nio，spark 1.2开始，默认就是netty，比较简单而且性能较高，spark 1.5开始nio就是过期的了，而且spark 1.6中会去除掉 |
| spark.shuffle.compress | true | 是否对map side输出的文件进行压缩，默认是启用压缩的，压缩器是由spark.io.compression.codec属性指定的，默认是snappy压缩器，该压缩器强调的是压缩速度，而不是压缩率 |
| spark.shuffle.consolidateFiles | false | 默认为false，如果设置为true，那么就会合并map side输出文件，对于reduce task数量特别的情况下，可以极大减少磁盘IO开销，提升性能 |
| spark.shuffle.file.buffer | 32k | map side task的内存buffer大小，写数据到磁盘文件之前，会先保存在缓冲中，如果内存充足，可以适当加大，从而减少map side磁盘IO次数，提升性能 |
| spark.shuffle.io.maxRetries | 3 | 网络传输数据过程中，如果出现了网络IO异常，重试拉取数据的次数，默认是3次，对于耗时的shuffle操作，建议加大次数，以避免full gc或者网络不通常导致的数据拉取失败，进而导致task lost，增加shuffle操作的稳定性 |
| spark.shuffle.io.retryWait | 5s | 每次重试拉取数据的等待间隔，默认是5s，建议加大时长，理由同上，保证shuffle操作的稳定性 |
| spark.shuffle.io.numConnectionsPerPeer | 1 | 机器之间的可以重用的网络连接，主要用于在大型集群中减小网络连接的建立开销，如果一个集群的机器并不多，可以考虑增加这个值 |
| spark.shuffle.io.preferDirectBufs | true | 启用堆外内存，可以避免shuffle过程的频繁gc，如果堆外内存非常紧张，则可以考虑关闭这个选项 |
| spark.shuffle.manager | sort | ShuffleManager，Spark 1.5以后，有三种可选的，hash、sort和tungsten-sort，sort-based ShuffleManager会更高效实用内存，并且避免产生大量的map side磁盘文件，从Spark 1.2开始就是默认的选项，tungsten-sort与sort类似，但是内存性能更高 |
| spark.shuffle.memoryFraction | 0.2 | 如果spark.shuffle.spill属性为true，那么该选项生效，代表了executor内存中，用于进行shuffle reduce side聚合的内存比例，默认是20%，如果内存充足，建议调高这个比例，给reduce聚合更多内存，避免内存不足频繁读写磁盘 |
| spark.shuffle.service.enabled | false | 启用外部shuffle服务，这个服务会安全地保存shuffle过程中，executor写的磁盘文件，因此executor即使挂掉也不要紧，必须配合spark.dynamicAllocation.enabled属性设置为true，才能生效，而且外部shuffle服务必须进行安装和启动，才能启用这个属性 |
| spark.shuffle.service.port | 7337 | 外部shuffle服务的端口号，具体解释同上 |
| spark.shuffle.sort.bypassMergeThreshold | 200 | 对于sort-based ShuffleManager，如果没有进行map side聚合，而且reduce task数量少于这个值，那么就不会进行排序，如果你使用sort ShuffleManager，而且不需要排序，那么可以考虑将这个值加大，直到比你指定的所有task数量都打，以避免进行额外的sort，从而提升性能 |
| spark.shuffle.spill | true | 当reduce side的聚合内存使用量超过了spark.shuffle.memoryFraction指定的比例时，就进行磁盘的溢写操作 |
| spark.shuffle.spill.compress | true | 同上，进行磁盘溢写时，是否进行文件压缩，使用spark.io.compression.codec属性指定的压缩器，默认是snappy，速度优先 |