華中科技大学

课 程 实 验 报 告

课程名称： 大数据分析

专业班级： 物联网工程 1801 班

学　　号： U201814500

姓　　名： 王英嘉

指导教师： 崔金华

报告日期： 2020.12.9

计算机科学与技术学院

目录

实验一 wordCount 算法及其实现

1.1 实验目的

1、理解 map-reduce 算法思想与流程；
2、应用 map-reduce 思想解决 wordCount 问题；
3、（可选）掌握并应用 combine 与 shuffle 过程。

1.2 实验内容

提供 9 个预处理过的源文件（source01-09）模拟 9 个分布式节点，每个源文件中包含一百万个由英文、数字和字符（不包括逗号）构成的单词，单词由逗号与换行符分割。

要求应用 map-reduce 思想，模拟 9 个 map 节点与 3 个 reduce 节点实现 wordCount 功能，输出对应的 map 文件和最终的 reduce 结果文件。由于源文件较大，要求使用多线程来模拟分布式节点。

学有余力的同学可以在 map-reduce 的基础上添加 combine 与 shuffle 过程，并可以计算线程运行时间来考察这些过程对算法整体的影响。

提示：实现 shuffle 过程时应保证每个 reduce 节点的工作量尽量相当，来减少整体运行时间。

1.3 实验过程

1.3.1 编程思路

实现 wordcount 主要依托代码中的 map 和 reduce 两个功能。

map 功能由 9 个模拟节点实现，分别对应 source 目录下的一个文件，通过多线程方式让 9 个模拟节点同时进行处理，将实验提供的文本转化成单词作为 key，value 为 1 的键值对形式，按行保存在 map 文件中，标号 map01-map09。

reduce 功能由 3 个模拟节点实现，1 个模拟节点对应 3 个生成的 map 文件，同样通过多线程方式同时进行处理，将 map 文件中每行键值对中的单词实现合并计数，最终生成单词作为 key，value 可能大于 1 的键值对形式，并按行保存在 reduce 文件中，标号 reduce00-reduce02。

在这之后，对 reduce 功能生成的三个文件进行整合，再执行一次类似 reduce 的过程，最终生成仍然是键值对形式的名为 wordcount 文件。

在此基础上，combine 功能即在 map 文件生成之前对键值对按 key 先作一次合并计数，例如 10 行 ok 1 将以 ok 10 在一行出现，以此减少 map 文件大小，提高效率；shuffle 功能实现了按 key 排序，方便后续 reduce 的读取和处理，这里使用了 Python 中 collections 模块里的 OrderedDict 来自动对加入的键值对进行排序。

1.3.2 遇到的问题及解决方式

在 reduce 过程生成文件后，不应该在 reduce 函数内部进行 merge，因为 reduce 是通过多线程方式进行的，如果在内部 merge 将重复写文件三次，并且导致每次写入的内容只是该线程处理的内容，即内容缺失。

上述问题通过在 reduce 函数外部实现一个 merge 函数，在外部调用即可。

1.3.3 实验测试与结果分析

分别在 common 模式和 combine and shuffle 模式下进行测试，结果如图 1。

```
yingjia@yingjia-Vostro-5370:~/文档/vscode/python/lab1$ python wordcount.py
In common mode...
Map done. Time Cost = 5.16s
Reduce done. Time Cost = 9.20s
timeSum = 14.36s
yingjia@yingjia-Vostro-5370:~/文档/vscode/python/lab1$ python wordcount.py
In combine and shuffle mode...
Map done. Time Cost = 11.52s
Reduce done. Time Cost = 6.07s
timeSum = 17.59s
```

图 1

在 combine 和 shuffle 模式下，map 函数耗时增加，reduce 函数耗时减少，这是因为在 map 函数中需要维护 OrderedDict，但同时生成的 map 文件内容更少，所以 reduce 函数中处理时间自然会减少。

1.4 实验总结

通过完成本次实验，进一步加深了对 mapreduce 原理的理解，也更加体会了 mapreduce 的并行性和高性能是如何实现的，同时还查阅了一些资料，对内部 combine 和 shuffle 的原理也有了一定的了解。