

**课 程 实 验 报 告**

**课程名称： 大数据分析**

**专业班级： CS1806**

**学 号： U201814670**

**姓 名： 李田田**

**指导教师： 崔金华**

**报告日期： 2020.12.10**

**计算机科学与技术学院**

**目录**

[实验一 wordCount算法及其实现 1](#_Toc58252267)

**[1.1实验目的](#_Toc58252268)** [1](#_Toc58252268)

**[1.2 实验内容](#_Toc58252269)** [1](#_Toc58252269)

**[1.3 实验过程](#_Toc58252270)** [1](#_Toc58252270)

[1.3.1 编程思路 1](#_Toc58252271)

[1.3.2 遇到的问题及解决方式 2](#_Toc58252272)

[1.3.3 实验测试与结果分析 2](#_Toc58252273)

**[1.4 实验总结](#_Toc58252274)** [5](#_Toc58252274)

# 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

Map的过程：

通过9个线程，每个线程对应一个source文件，输出到一个map文件；9个线程的参数不一样，但是运行的线程函数一样。

在线程函数里，打开读取文件，从读取文件中获取每一行，将该行使用strip函数去掉最后的回车，再用split函数将该行在‘， ’出分隔开，每个单词为一行；将每行的单词以格式（word， 1）写入map文件。

Shuffle的过程：

通过9个线程将9个map文件的每个文件按照首字母进行分开，不同的首字母进入不同的shuffle中，一共有三个shuffle文件。注意此处打开写入文件的方式应该是‘a’，每一次都在文件后面增加，而不是重新写入。

此处的过程是为了对最后的统计结果去重。

Reduce的过程：

Reduce的过程使用三个线程，每个线程对应一个输入文件shuffle，一个输出文件reduce。

使用python的字典将读取文件中的每一个单词提取出来存入字典，如果字典中含有该元素，则加一，如果没有，则将该元素加入字典中，key值记为1。遍历完成后将得到的字典进行排序，完成后将字典按照（word， num）的格式写入文件。

对每个线程采用time.clock函数进行计时。

### 1.3.2 遇到的问题及解决方式

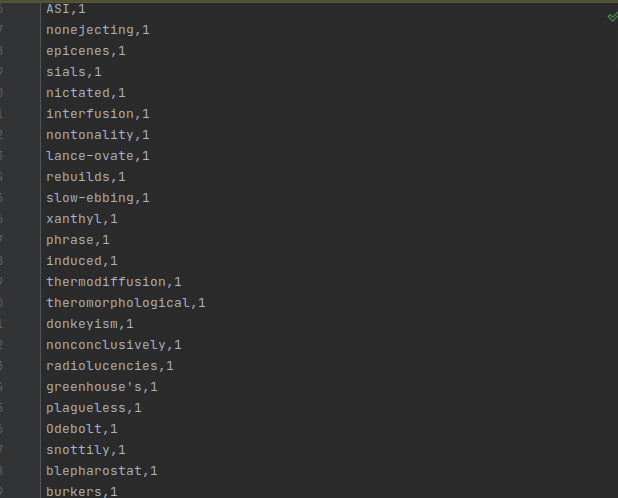
在该实验刚开始写的时候，用到python语言之前并没有接触过，写起来会有些困难，主要是库太多以及数据使用不太熟悉，比如字典、列表、set都不是特别熟练，对其构造优劣也不是特别清楚，造成了极大的麻烦。

解决方法：只能对着网上的python教程硬啃，一点一点地学，这点别无他法。

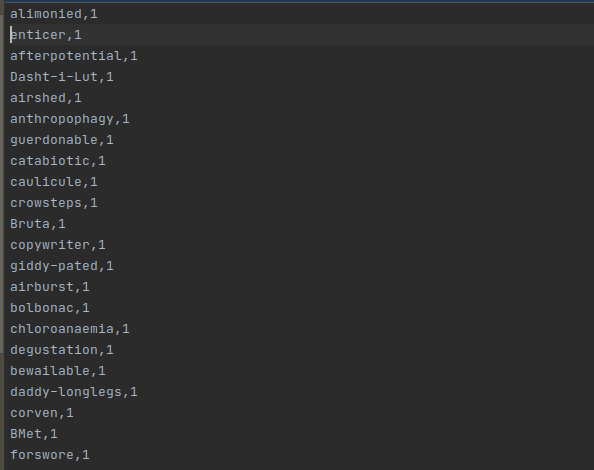
### 1.3.3 实验测试与结果分析

测试：

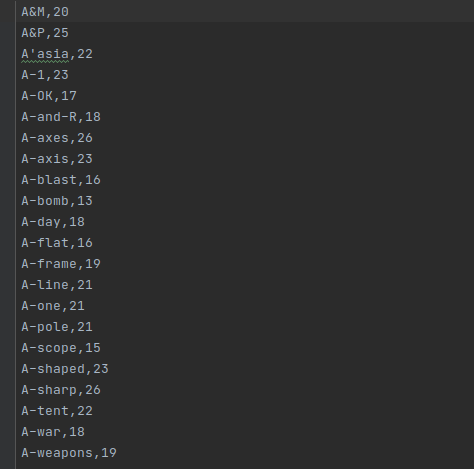
Map文件：以<word, count>的形式将文件里的所有单词统计进行写入到map文件。



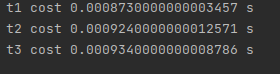
Shuffle文件：对每9个map文件的单词进行分类成3个shuffle文件。结果如下图所示。



Reduce文件：



时间计算：



结果分析：

最后得到的计数为该单词在9个source文件中出现的次数。该结果符合预期结果。

## **1.4 实验总结**

通过实现map-reduce的过程，将整个算法的过程理解了一边，在该过程中，对combine和shuffle的过程进行了体会和融合，形成最后的实验代码和结果。此外，在该过程体会并学到了wordcount的核心内容，之前只在hadoop上面运行过map-reduce，知道大概内容。在该实验进行的过程中，也观察到了线程对整个程序的影响，各个线程的差异倒不是很大，但是对整个程序的性能的影响还是显而易见的。