实验 2　　2021sg07　张涵之，林芳麒

1. Map 和 Reduce 的设计思路（含 Key、Value 类型）。

   Map：输入 Value 类型为 Text，包含当前处理的文件中的所有词
   输出 Key 类型为 Text，实际上是由 term 和 filename 拼接成的 pair，Value 类型为 IntWritable，是统计到 term 在 filename 对应文件内出现的次数，计数过程中用到了一个类型为<String, Integer>的 HashTable 来临时储存数据。

   Partitioner：将 Key 所表示 pair 中的 term 取出，根据 term 进行 partition

   Reducer：currentWord 记录当前处理到的单词，wordCount 是这个词在全部文件中出现次数的计数器，fileCount 是含有这个词的文件数量的计数器。All 作为一个 buffer 临时存放<term, filename> pair，鉴于同一个词语的这些 pair 每次都要在完成对单词平均出现次数的统计之后，才进行统一的输出。
   输出的 Key 和 Value 类型都是 Text。其中 reduce 方法每次先检查当前是否处理到一个新单词，如果是则将单词作为 Key，词频和 buffer 中所有的 pair 拼接起来作为 Value 输出，并将 wordCount 和 fileCount 清零，buffer 清空。接着再更新 currentWord，wordCount 和 fileCount 并将 pair 添加到 buffer 中。
   由于 reduce 方法每次读到新单词才对上一个单词进行输出，reducer 完成当前节点上所有任务后，还需要在 cleanup 中对最后一个单词进行输出。

2. MapReduce 中 Map 和 Reduce 的伪代码。

```
class InvertedIndexMapper
    method map(Text value, Context context)
        get filename from context
        get filecontent from value
        F ← new HashTable<String, Integer>
        for all term t in filecontent do
            if F no has key t, F{t} ← 1
            else F{t} ← F{t} + 1
        for all pair <term t, count cnt> in F do
            pair p ← <t, filename>
            write (p, cnt) to context
            // counts the occurrence of a term in a specific file

class NewPartitioner
    method getPartition(Text key, IntWritable value)
        get pair<term t, filename fn> p from key
        get count cnt from value
        perform HashPartitioner on (t, cnt)
        // performs partition only regarding the term, leaves filename alone
```

```
class InvertedIndexReducer
    method setup()
        currentWord ← null
        wordCount ← 0
        fileCount ← 0
        all ← new String Builder (to buffer <term, filename> pairs)

    method reduce(Text(pair) key, IntWritable[] values, Context context)
        get pair<term t, filename fn> p from key
        get count cnt from value(s)
        if currentWord is null or currentWord != t
            // finish counting the current term, update to context
            average ← wordCount / fileCount
            write (term, average and all <term, filename> pairs) to context
            reset wordCount, fileCount to 0 and clear all buffered pairs
        fileCount ← fileCount + 1
        wordCount ← wordCount + cnt
        add <t, fn> pair to all buffer
        update currentWord to t

    method cleanup(Context context)
        // finish counting all terms on this Reducer, update to context
        average ← wordCount / fileCount
        write (term, average and all <term, filename> pairs) to context
```

3. 输出结果文件的部分截图。输出结果文件在 HDFS 上的路径。

hadoop jar InvertedIndexer.jar /data/chinese_novels /user/2021sg07/invert_output
输出结果文件在 HDFS 上的路径为 user/2021sg07/invert_output

4. “我们”、“什么”两个单词的输出结果。

输出结果为从网页上下载文件后，使用查找功能定位并复制到报告中：

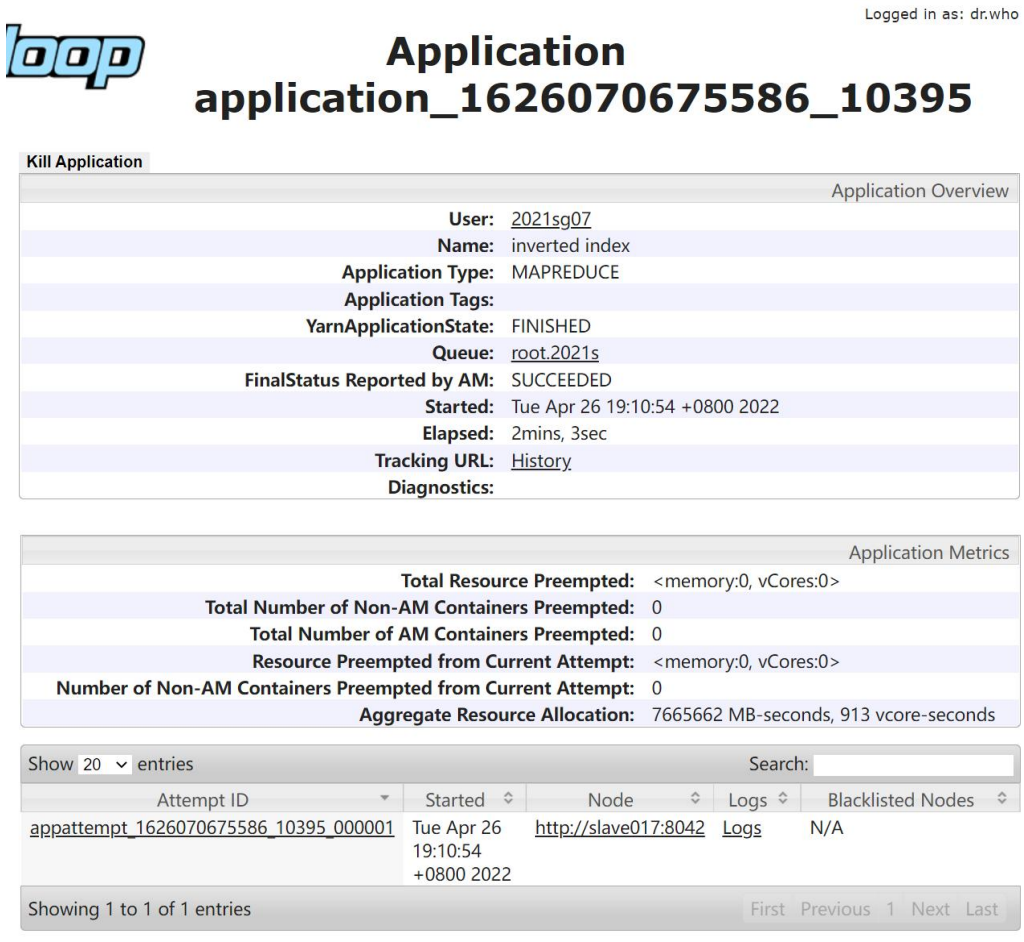
我们 5.719 围城-一:1; 围城-七:11; 围城-三:5; 围城-三十:6; 围城-三十一:1; 围城-三十三:1; 围城-三十二:6; 围城-三十四:7; 围城-九:13; 围城-二:3; 围城-二十:4; 围城-二十一:2; 围城-二十七:2; 围城-二十三:5; 围城-二十九:2; 围城-二十二:3; 围城-二十五:5; 围城-二十八:4; 围城-二十六:2; 围城-二十四:11; 围城-五:8; 围城-八:7; 围城-六:3; 围城-十:5; 围城-十一:5; 围城-十七:12; 围城-十三:14; 围城-十九:5; 围城-十二:2; 围城-十五:1; 围城-十八:8; 围城-十六:8; 围城-十四:3; 围城-四:4; 围城-零:5; 城南旧事-一:18; 城南旧事-七:20; 城南旧事-三:12; 城南旧事-九:23; 城南旧事-二:11; 城南旧事-五:26; 城南旧事-八:29; 城南旧事-六:23; 城南旧事-四:17; 城南旧事-零:17; 平凡的世界-一:2; 平凡的世界-一百:5; 平凡的世界-一百一十:3; 平凡的世界-一百一十一:6; 平凡的世界-一百一十七:16; 平凡的世界-一百一十三:5; 平凡的世界-一百一十九:4; 平凡的世界-一百一十二:5; 平凡的世界-一百一十五:6; 平凡的世界-一百一十八:8; 平凡的世界-一百一十六:5; 平凡的世界-一百一十四:29; 平凡的世界-一百二十:3; 平凡的世界-一百二十一:3; 平凡的世界-一百二十七:5; 平凡的世界-一百二十三:2; 平凡的世界-一百二十九:6; 平凡的世界-一百二十二:5; 平凡的世界-一百二十五:22; 平凡的世界-一百二十八:10; 平凡的世界-一百二十六:3; 平凡的世界-一百二十四:4; 平凡的世界-一百零一:9; 平凡的世界-一百零七:11; 平凡的世界-一百零三:5; 平凡的世界-一百零九:6; 平凡的世界-一百零二:4; 平凡的世界-一百零五:2; 平凡的世界-一百零八:14; 平凡的世界-一百零六:5; 平凡的世界-一百零四:3; 平凡的世界-七:1; 平凡的世界-七十:4; 平凡的世界-七十一:7; 平凡的世界-七十七:6; 平凡的世界-七十三:14; 平凡的世界-七十九:7; 平凡的世界-七十二:7; 平凡的世界-七十五:9; 平凡的世界-七十八:2; 平凡的世界-七十六:3; 平凡的世界-七十四:1; 平凡的世界-三十:5; 平凡的世界-三十一:4; 平凡的世界-三十七:1; 平凡的世界-三十三:1; 平凡的世界-三十九:5; 平凡的世界-三十二:1; 平凡的世界-三十五:5; 平凡的世界-三十八:12; 平凡的世界-三十六:1; 平凡的世界-九:3; 平凡的世界-九十:4; 平凡的世界-九十一:6; 平凡的世界-九十七:14; 平凡的世界-九十三:6; 平凡的世界-九十九:4; 平凡的世界-九十五:9; 平凡的世界-九十八:1; 平凡的世界-九十六:1; 平凡的世界-九十四:10; 平凡的世界-二:1; 平凡的世界-二十:6; 平凡的世界-二十一:4; 平凡的世界-二十七:3; 平凡的世界-二十三:8; 平凡的世界-二十九:11; 平凡的世界-二十二:3; 平凡的世界-二十五:11; 平凡的世界-二十八:9; 平凡的世界-二十六:8; 平凡的世界-二十四:2; 平凡的世界-五十:3; 平凡的世界-五十一:5; 平凡的世界-五十七:7; 平凡的世界-五十九:8; 平凡的世界-五十二:2; 平凡的世界-五十五:4; 平凡的世界-五十八:4; 平凡的世界-五十六:2; 平凡的世界-五十四:2; 平凡的世界-八:3; 平凡的世界-八十:12; 平凡的世界-八十一:1; 平凡的世界-八十七:6; 平凡的世界-八十三:22; 平凡的世界-八十九:3; 平凡的世界-八十二:4; 平凡的世界-八十五:7; 平凡的世界-八十八:3; 平凡的世界-八十六:1; 平凡的世界-八十四:7; 平凡的世界-六十:4; 平凡的世界-六十七:2; 平凡的世界-六十三:3; 平凡的世界-六十九:2; 平凡的世界-六十二:7; 平凡的世界-六十五:3; 平凡的世界-六十八:3; 平凡的世界-六十六:1; 平凡的世界-六十四:5; 平凡的世界-十:2; 平凡的世界-十一:2; 平凡的世界-十七:1; 平凡的世界-十三:1; 平凡的世界-十二:2; 平凡的世界-十五:4; 平凡的世界-十六:1; 平凡的世界-十四:5; 平凡的世界-四:1; 平凡的世界-四十一:1; 平凡的世界-四十七:6; 平凡的世界-四十九:1; 平凡的世界-四十二:5; 平凡的世界-四十五:6; 平凡的世界-四十八:6; 平凡的世界-四十六:11; 平凡的世界-四十四:5; 平凡的世界-零:2; 白鹿原-七十:3; 白鹿原-七十三:5; 白鹿原-七十四:1; 白鹿原-三十:3; 白鹿原-三十一:1; 白鹿原-三十六:1; 白鹿原-二十一:8; 白鹿原-二十三:12; 白鹿原-二十二:4; 白鹿原-二十五:6; 白鹿原-二十四:6; 白鹿原-五十七:2; 白鹿原-五十三:5; 白鹿

原-五十九:1; 白鹿原-五十五:1; 白鹿原-五十六:2; 白鹿原-八:1; 白鹿原-六十:3; 白鹿原-六十一:7; 白鹿原-六十三:2; 白鹿原-六十九:1; 白鹿原-六十五:1; 白鹿原-十二:1; 白鹿原-十八:2; 白鹿原-四十七:1; 白鹿原-四十三:7; 白鹿原-四十九:2; 白鹿原-四十二:4; 白鹿原-四十八:15; 白鹿原-四十六:10; 白鹿原-四十四:1; 边城-七:3; 边城-三:2; 边城-九:4; 边城-二:5; 边城-五:2; 边城-八:2; 边城-六:2; 边城-四:8

什么 8.723 围城-一:6; 围城-七:11; 围城-三:12; 围城-三十:9; 围城-三十一:14; 围城-三十三:15; 围城-三十二:18; 围城-三十四:10; 围城-九:17; 围城-二:14; 围城-二十:13; 围城-二十一:11; 围城-二十七:11; 围城-二十三:5; 围城-二十九:14; 围城-二十二:4; 围城-二十五:10; 围城-二十八:9; 围城-二十六:12; 围城-二十四:14; 围城-五:12; 围城-八:7; 围城-六:16; 围城-十:11; 围城-十一:12; 围城-十七:5; 围城-十三:10; 围城-十九:13; 围城-十二:5; 围城-十五:9; 围城-十八:8; 围城-十六:9; 围城-十四:3; 围城-四:10; 围城-零:4; 城南旧事-一:25; 城南旧事-七:19; 城南旧事-三:18; 城南旧事-九:2; 城南旧事-二:13; 城南旧事-五:32; 城南旧事-八:15; 城南旧事-六:22; 城南旧事-四:18; 城南旧事-零:21; 平凡的世界-一:9; 平凡的世界-一百:7; 平凡的世界-一百一十:7; 平凡的世界-一百一十一:4; 平凡的世界-一百一十七:9; 平凡的世界-一百一十三:8; 平凡的世界-一百一十九:8; 平凡的世界-一百一十二:6; 平凡的世界-一百一十五:6; 平凡的世界-一百一十八:11; 平凡的世界-一百一十六:8; 平凡的世界-一百一十四:7; 平凡的世界-一百二十:10; 平凡的世界-一百二十一:12; 平凡的世界-一百二十七:17; 平凡的世界-一百二十三:13; 平凡的世界-一百二十九:3; 平凡的世界-一百二十二:4; 平凡的世界-一百二十五:8; 平凡的世界-一百二十八:6; 平凡的世界-一百二十六:10; 平凡的世界-一百二十四:9; 平凡的世界-一百零一:10; 平凡的世界-一百零七:13; 平凡的世界-一百零三:8; 平凡的世界-一百零九:2; 平凡的世界-一百零二:9; 平凡的世界-一百零五:13; 平凡的世界-一百零八:6; 平凡的世界-一百零六:8; 平凡的世界-一百零四:6; 平凡的世界-七:12; 平凡的世界-七十:10; 平凡的世界-七十一:15; 平凡的世界-七十七:8; 平凡的世界-七十三:11; 平凡的世界-七十九:22; 平凡的世界-七十二:14; 平凡的世界-七十五:6; 平凡的世界-七十八:11; 平凡的世界-七十六:12; 平凡的世界-七十四:19; 平凡的世界-三:26; 平凡的世界-三十:12; 平凡的世界-三十一:12; 平凡的世界-三十七:8; 平凡的世界-三十三:5; 平凡的世界-三十九:8; 平凡的世界-三十二:7; 平凡的世界-三十五:8; 平凡的世界-三十八:8; 平凡的世界-三十六:6; 平凡的世界-三十四:9; 平凡的世界-九:7; 平凡的世界-九十:9; 平凡的世界-九十一:6; 平凡的世界-九十七:6; 平凡的世界-九十三:13; 平凡的世界-九十九:9; 平凡的世界-九十二:7; 平凡的世界-九十五:11; 平凡的世界-九十八:9; 平凡的世界-九十六:5; 平凡的世界-九十四:11; 平凡的世界-二:12; 平凡的世界-二十:5; 平凡的世界-二十一:9; 平凡的世界-二十七:5; 平凡的世界-二十三:12; 平凡的世界-二十九:8; 平凡的世界-二十二:6; 平凡的世界-二十五:7; 平凡的世界-二十八:8; 平凡的世界-二十六:5; 平凡的世界-二十四:9; 平凡的世界-五:9; 平凡的世界-五十:4; 平凡的世界-五十一:9; 平凡的世界-五十七:10; 平凡的世界-五十三:8; 平凡的世界-五十九:20; 平凡的世界-五十二:5; 平凡的世界-五十五:8; 平凡的世界-五十八:5; 平凡的世界-五十六:17; 平凡的世界-五十四:16; 平凡的世界-八:9; 平凡的世界-八十:5; 平凡的世界-八十一:11; 平凡的世界-八十七:7; 平凡的世界-八十三:14; 平凡的世界-八十九:13; 平凡的世界-八十二:14; 平凡的世界-八十五:19; 平凡的世界-八十八:16; 平凡的世界-八十六:17; 平凡的世界-八十四:9; 平凡的世界-六:18; 平凡的世界-六十:8; 平凡的世界-六十一:19; 平凡的世界-六十七:12; 平凡的世界-六十三:13; 平凡的世界-六十九:10; 平凡的世界-六十二:13; 平凡的世界-六十五:13; 平凡的世界-六十八:16; 平凡的世界-六十六:7; 平凡的世界-六十四:4; 平凡的世界-十:24; 平凡的世界-十一:9; 平凡的世界-十七:5; 平凡的世界-十三:13; 平凡的世界-十九:16; 平凡的世界-十二:10; 平凡的世界-十五:9;

平凡的世界-十八:15; 平凡的世界-十六:14; 平凡的世界-十四:15; 平凡的世界-四:12; 平凡的世界-四十:14; 平凡的世界-四十一:10; 平凡的世界-四十七:13; 平凡的世界-四十三:7; 平凡的世界-四十九:15; 平凡的世界-四十二:6; 平凡的世界-四十五:13; 平凡的世界-四十八:9; 平凡的世界-四十六:16; 平凡的世界-四十四:11; 平凡的世界-零:7; 白鹿原-一:7; 白鹿原-七:1; 白鹿原-七十:3; 白鹿原-七十一:2; 白鹿原-七十三:4; 白鹿原-七十二:5; 白鹿原-七十四:2; 白鹿原-三:1; 白鹿原-三十一:2; 白鹿原-三十七:3; 白鹿原-三十三:2; 白鹿原-三十九:2; 白鹿原-三十二:3; 白鹿原-三十五:5; 白鹿原-三十八:3; 白鹿原-三十六:2; 白鹿原-三十四:1; 白鹿原-九:5; 白鹿原-二:1; 白鹿原-二十一:2; 白鹿原-二十七:1; 白鹿原-二十九:2; 白鹿原-二十二:5; 白鹿原-二十五:1; 白鹿原-二十六:2; 白鹿原-二十四:2; 白鹿原-五:4; 白鹿原-五十:2; 白鹿原-五十一:1; 白鹿原-五十七:5; 白鹿原-五十三:2; 白鹿原-五十九:4; 白鹿原-五十二:4; 白鹿原-五十五:2; 白鹿原-五十八:3; 白鹿原-五十六:2; 白鹿原-五十四:2; 白鹿原-八:6; 白鹿原-六:2; 白鹿原-六十:7; 白鹿原-六十一:1; 白鹿原-六十七:4; 白鹿原-六十三:3; 白鹿原-六十二:3; 白鹿原-六十五:2; 白鹿原-六十八:2; 白鹿原-六十六:1; 白鹿原-六十四:1; 白鹿原-十:3; 白鹿原-十一:3; 白鹿原-十七:4; 白鹿原-十三:5; 白鹿原-十九:6; 白鹿原-十二:8; 白鹿原-十五:3; 白鹿原-十八:3; 白鹿原-十六:3; 白鹿原-十四:7; 白鹿原-四十:2; 白鹿原-四十一:3; 白鹿原-四十七:8; 白鹿原-四十三:3; 白鹿原-四十九:5; 白鹿原-四十五:4; 白鹿原-四十八:5; 白鹿原-四十六:9; 白鹿原-四十四:2; 白鹿原-零:7; 边城-一:8; 边城-七:10; 边城-三:24; 边城-九:7; 边城-二:11; 边城-五:18; 边城-八:15; 边城-六:8; 边城-四:5; 边城-零:2

5. 在集群上执行作业后，Yarn Resource Manager 的 WebUI 执行报告内容。

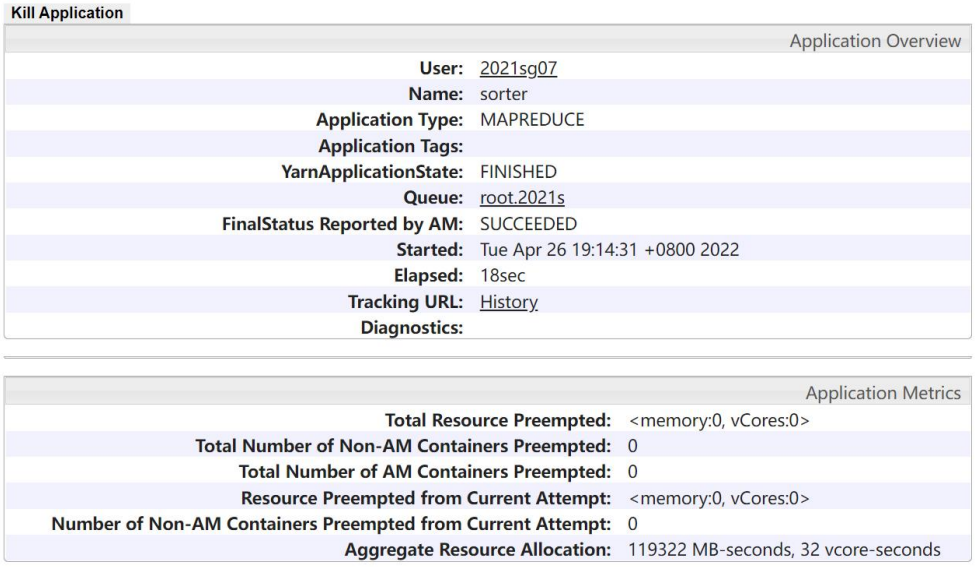


6. 选做内容

1) 对每个词语的平均出现次数进行从大到小全局排序。

按照课件上的 MapReduce 排序算法，输入为必做部分倒排索引的结果，输出为排好序的词语及平均出现次数。其中 Mapper 和 Reducer 的作用类似 Identify Function，使用 Hadoop 提供的 TotalOrderPartitioner，由于这种 Partitioner 实现的是从小到大排序，故 Mapper 先对平均出现次数取负存在 key 中用于反向排序，排好序后，Reducer 再次取负将平均数作为 value 输出。

hadoop jar Sort.jar /user/2021sg07/invert_output /user/2021sg07/sort_output
输出结果文件在 HDFS 上的路径为 user/2021sg07/sort_output



2) 为每个作品计算每个词语的 TF-IDF。

输入为必做部分倒排索引的结果，输出为每个作品中每个词语的 TF-IDF。其中同一部作品的词语在一起，但词语本身是无序的。Mapper 按行读入词语和各文件中的出现次数，采用 HashTable 统计该词在每部作品中的次数，统计完毕后以 workName 为 key，<term, tf-idf> pair 为 value，再由 Reducer 汇总按格式输出。语料库文档总数仍然从原始数据文件获得，因此在执行 jar 包时除了输入和输出，还有第三个参数是 chinese_novel 所在的路径。

hadoop jar TF-IDF.jar /user/2021sg07/invert_output /user/2021sg07/tf-idf_output /data/chinese_novels
输出结果文件在 HDFS 上的路径为 user/2021sg07/tf-idf_output

Kill Application

Application Overview	
User:	2021sg07
Name:	TF-IDFer
Application Type:	MAPREDUCE
Application Tags:	
YarnApplicationState:	FINISHED
Queue:	root.2021s
FinalStatus Reported by AM:	SUCCEEDED
Started:	Tue Apr 26 19:18:14 +0800 2022
Elapsed:	18sec
Tracking URL:	History
Diagnostics:	

Application Metrics	
Total Resource Preempted:	<memory:0, vCores:0>
Total Number of Non-AM Containers Preempted:	0
Total Number of AM Containers Preempted:	0
Resource Preempted from Current Attempt:	<memory:0, vCores:0>
Number of Non-AM Containers Preempted from Current Attempt:	0
Aggregate Resource Allocation:	129653 MB-seconds, 33 vcore-seconds