

计算机科学与技术学院_大数据管理与分析_课程实验报告

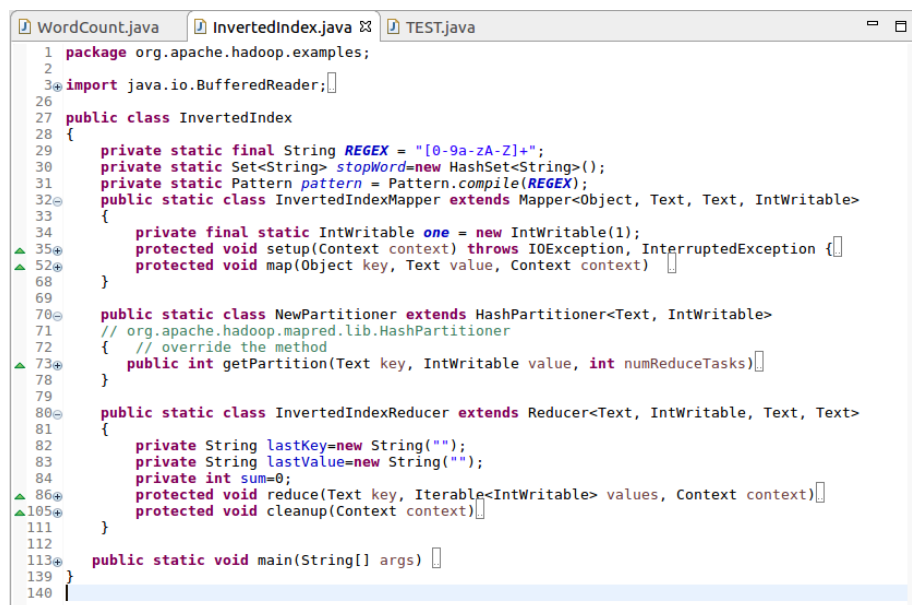
实验题目：文档倒排索引算法实现		学号：201605130116
日期：2019.4.17	班级：2016 级泰山学堂	姓名：杜洪超
Email： 1503345074@qq.com		
<p>实验目的：</p> <p>倒排索引（Inverted Index）被用来存储在全文搜索下某个单词在一个文档或者一组文档中的存储位置的映射，是目前几乎所有支持全文索引的搜索引擎都需要依赖的一个数据结构。通过对倒排索引的编程实现，熟练掌握 MapReduce 程序在集群上的提交与执行过程，加深对 MapReduce 编程框架的理解。</p>		
<p>实验软件和硬件环境：</p> <p>软件环境：</p> <p>系统：Ubuntu16.04 LTS 64 位，集群环境为 centos6.5</p> <p>软件：openjdk-8-jre, openjdk-8-jdk, java1.8.0_191</p> <p>Hadoop 2.9.2，集群为 2.9.0</p> <p>Eclipse, ssh</p> <p>硬件环境：</p> <p>CPU：Intel® Core™ i5-6260U CPU @ 1.80GHz × 4</p> <p>磁盘：121.8 GB</p> <p>内存：7.7 GiB</p>		
<p>实验原理和方法：</p> <ol style="list-style-type: none">1. 本地编写倒排索引算法的 MapReduce 程序，导出 jar 包后提交到集群执行；2. 比较集群执行结果和标准答案，验证程序是否正确		
<p>实验步骤：（不要求罗列完整源代码）</p> <ol style="list-style-type: none">1. 本地编写程序和调试 <p>文档倒排索引算法的 MapReduce 程序主要有以下几个要点：</p> <ol style="list-style-type: none">(1) 设置合适的分词器；Java 中的 StringTokenizer 类的分词功能不够灵活，可以使用正则表达式对文本的每一行进行模式匹配，实现更精细的分词(2) Map 的输出；为了实现按文档 id 排序，Map 输出的 key 为单词+文档名，这样在进入 Reduce 前同一单词的 key-value 对会自动按文档排序；value 设置为 1，用于计数；(3) 定制 Partitioner；(2) 中为了实现排序修改了 key 的定义，会导致 reduce 时相同单词被 hash 到不同节点，导致结果错误，因此使用定制 Partitioner 类来解决这个问题；Partitioner 类处理的 key 仍然是单词+文档名个格式，但 getPartition 时只处理单词部分，从而保证了相同单		

词都映射到同一 Reduce 节点;

- (4) 为了把不同文件中的相同单词拼接在一起输出,借助 Partitioner 处理后的 key-value 对序列按单词和文档排序的特性,设置两个私有变量保存当前处理的单词以及目前次单词出现的文档和词频信息,如果下一个 key-value 对仍属于这个单词,就将改文档和词频信息添加到私有变量中,否则证明现在处理的单词以及处理完毕,需要输出,然后更新当前处理的单词和文档及词频信息;这样的作法会导致最后一个单词无法输出,在 Reduce 类的 cleanup 函数中输出当前单词和文档和词频信息即可;

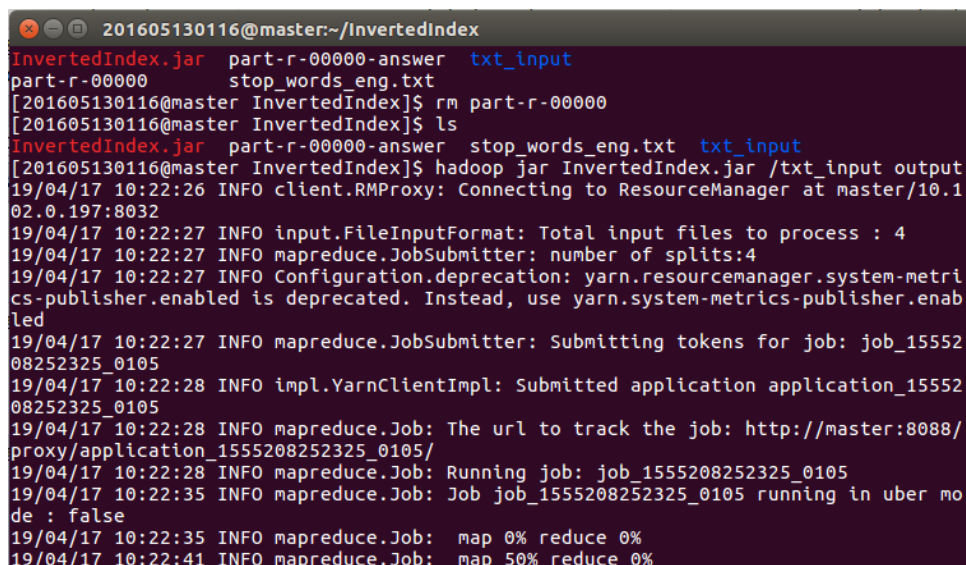
- (5) 去停用词;为了去掉结果中无用的停用词,通过在 Map 中判断当前单词是否存在与停用词表中来实现筛选操作;把停用词表设置为缓存文件,在 Map 的 setup 函数中读取停用词表;

全部类和函数如下:



```
1 package org.apache.hadoop.examples;
2
3 import java.io.BufferedReader;
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27 public class InvertedIndex
28 {
29     private static final String REGEX = "[0-9a-zA-Z]+";
30     private static Set<String> stopWord=new HashSet<String>();
31     private static Pattern pattern = Pattern.compile(REGEX);
32     public static class InvertedIndexMapper extends Mapper<Object, Text, Text, IntWritable>
33     {
34         private final static IntWritable one = new IntWritable(1);
35         protected void setup(Context context) throws IOException, InterruptedException {}
36         protected void map(Object key, Text value, Context context) {}
37     }
38
39     public static class NewPartitioner extends HashPartitioner<Text, IntWritable>
40     {
41         // org.apache.hadoop.mapred.lib.HashPartitioner
42         // override the method
43         public int getPartition(Text key, IntWritable value, int numReduceTasks){}
44     }
45
46     public static class InvertedIndexReducer extends Reducer<Text, IntWritable, Text, Text>
47     {
48         private String lastKey=new String("");
49         private String lastValue=new String("");
50         private int sum=0;
51         protected void reduce(Text key, Iterable<IntWritable> values, Context context){}
52         protected void cleanup(Context context){}
53     }
54
55     public static void main(String[] args) {}
56 }
```

2. 在集群上提交作业并执行;



```
201605130116@master:~/InvertedIndex
InvertedIndex.jar part-r-00000-answer txt_input
part-r-00000 stop_words_eng.txt
[201605130116@master InvertedIndex]$ rm part-r-00000
[201605130116@master InvertedIndex]$ ls
InvertedIndex.jar part-r-00000-answer stop_words_eng.txt txt_input
[201605130116@master InvertedIndex]$ hadoop jar InvertedIndex.jar /txt_input output
19/04/17 10:22:26 INFO client.RMProxy: Connecting to ResourceManager at master/10.102.0.197:8032
19/04/17 10:22:27 INFO input.FileInputFormat: Total input files to process : 4
19/04/17 10:22:27 INFO mapreduce.JobSubmitter: number of splits:4
19/04/17 10:22:27 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
19/04/17 10:22:27 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1555208252325_0105
19/04/17 10:22:28 INFO impl.YarnClientImpl: Submitted application application_1555208252325_0105
19/04/17 10:22:28 INFO mapreduce.Job: The url to track the job: http://master:8088/proxy/application_1555208252325_0105/
19/04/17 10:22:28 INFO mapreduce.Job: Running job: job_1555208252325_0105
19/04/17 10:22:35 INFO mapreduce.Job: Job job_1555208252325_0105 running in uber mode : false
19/04/17 10:22:35 INFO mapreduce.Job: map 0% reduce 0%
19/04/17 10:22:41 INFO mapreduce.Job: map 50% reduce 0%
```

作业执行情况:

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved
89	0	1	88	1	2 GB	24 GB	0 B

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes
3	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:4>

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores	Allocated Memory MB	Reserved CPU Vcores
application_1555208252325_0105	201605130116	invert index	MAPREDUCE	default	0	Wed Apr 17 10:22:27 +0800 2019	N/A	ACCEPTED	UNDEFINED	1	1	2048	0
application_1555208252325_0104	201605130116	invert index	MAPREDUCE	default	0	Tue Apr 16 22:47:20 +0800 2019	Tue Apr 16 22:47:37 +0800 2019	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A
application_1555208252325_0103	201605130116	invert index	MAPREDUCE	default	0	Tue Apr 16 22:45:04 +0800 2019	Tue Apr 16 22:45:23 +0800 2019	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A

验证结果：

Browse Directory

/user/201605130116/output

Go!

Show

25

entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	201605130116	201605130116	0 B	Apr 17 10:22	3	128 MB	_SUCCESS	
<input type="checkbox"/>	-rw-r--r--	201605130116	201605130116	1.84 MB	Apr 17 10:22	3	128 MB	part-r-00000	

Showing 1 to 2 of 2 entries

Previous

1

Next

201605130116@master:~/InvertedIndex

```

Reduce output records=32345
Spilled Records=1290406
Shuffled Maps =4
Failed Shuffles=0
Merged Map outputs=4
GC time elapsed (ms)=545
CPU time spent (ms)=24310
Physical memory (bytes) snapshot=3421491200
Virtual memory (bytes) snapshot=30494150656
Total committed heap usage (bytes)=7687634944

Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0

File Input Format Counters
  Bytes Read=9194714
File Output Format Counters
  Bytes Written=1925475
[201605130116@master InvertedIndex]$ hdfs dfs -get output/part-r-00000
[201605130116@master InvertedIndex]$ diff part-r-00000 part-r-00000-answer
[201605130116@master InvertedIndex]$

```

结论分析与体会：

通过实现文档倒排索引算法，掌握了 MapReduce 程序在集群上的提交与执行过程，加深了对 MapReduce 编程的理解；学习了自定义 Partitioner 的技巧，对 Map 和 Reduce 类的预处理和结束函数的作用有了更深的了解。

就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1—3 道问答题：

1. 分词结果不理想；

通过正则表达式实现分词；

2. 无法设置自定义的 Partitioner 类；

Import 类库错误；

3. 集群读取调用停用词失败；

将停用词文件设置成缓存文件，使各节点能够读取。