FBDP作业五

maven的熟悉与操作

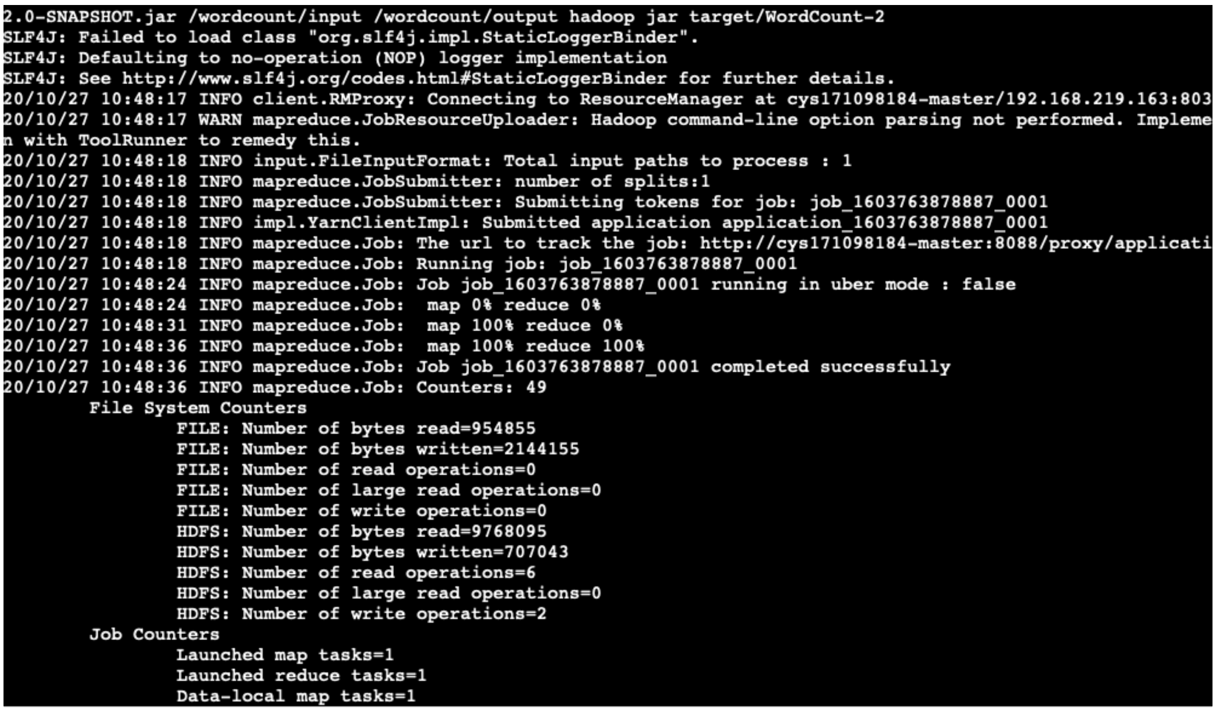
1. 首先在ide如vscode或者intellij上构建maven类，创建
2. 修改pom.xml，载入Hadoop，hdfs等插件；pom.xml内需要填写dependence以及build的插件信息
3. `mvn clean`先对文件进行晴空
4. `mvn package`对文档进行编译，得到jar文件
5. 发现报错`Exception in thread "main" java.lang.ClassNotFoundException: cys.nju.edu.cn.WordCount`原因是在pom里面要对主要的class进行定位，定位信息应该是packageName.className，若定位不准确则会报错

在bdkit上运行

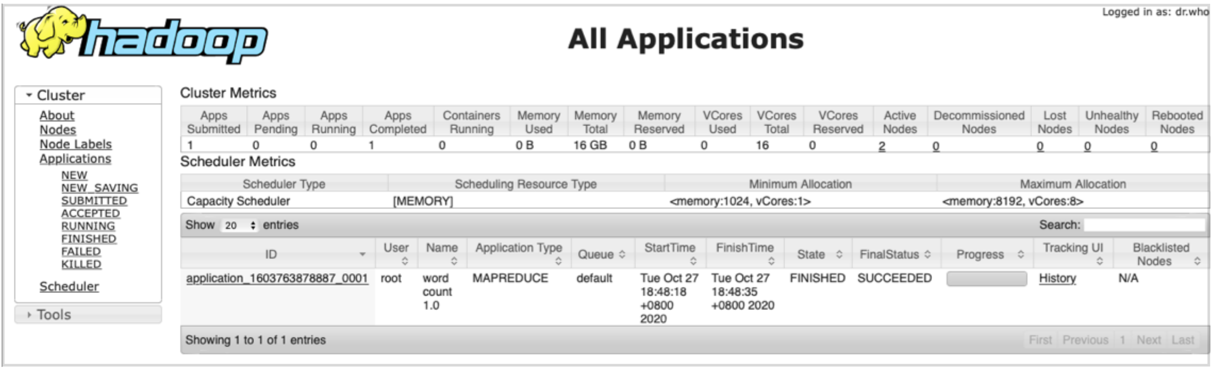
Hdfs运行的几个代码（往后熟悉就会记住）

```
hdfs dfs -mkdir /wordcount

hdfs dfs -put /input /wordcount

hadoop jar target/jarFileName.jar /wordcount/input /wordcount/output
```

首先运行的是wordcount1.0版本，确定熟悉mapreduce的基本编写以及maven的用法、hadoop的运行等



```
2.0-SNAPSHOT.jar /wordcount/input /wordcount/output hadoop jar target/WordCount-2
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
20/10/27 10:48:17 INFO client.RMProxy: Connecting to ResourceManager at cys171098184-master/192.168.219.163:803
20/10/27 10:48:17 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Impleme
n with ToolRunner to remedy this.
20/10/27 10:48:18 INFO input.FileInputFormat: Total input paths to process : 1
20/10/27 10:48:18 INFO mapreduce.JobSubmitter: number of splits:1
20/10/27 10:48:18 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1603763878887_0001
20/10/27 10:48:18 INFO impl.YarnClientImpl: Submitted application application_1603763878887_0001
20/10/27 10:48:18 INFO mapreduce.Job: The url to track the job: http://cys171098184-master:8088/proxy/applicati
20/10/27 10:48:18 INFO mapreduce.Job: Running job: job_1603763878887_0001
20/10/27 10:48:24 INFO mapreduce.Job: Job job_1603763878887_0001 running in uber mode : false
20/10/27 10:48:24 INFO mapreduce.Job:  map 0% reduce 0%
20/10/27 10:48:31 INFO mapreduce.Job:  map 100% reduce 0%
20/10/27 10:48:36 INFO mapreduce.Job:  map 100% reduce 100%
20/10/27 10:48:36 INFO mapreduce.Job: Job job_1603763878887_0001 completed successfully
20/10/27 10:48:36 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=954855
                FILE: Number of bytes written=2144155
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=9768095
                HDFS: Number of bytes written=707043
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
```



Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
1	0	0	1	0	0 B	16 GB	0 B	0	16	0	2	0	0	0	0

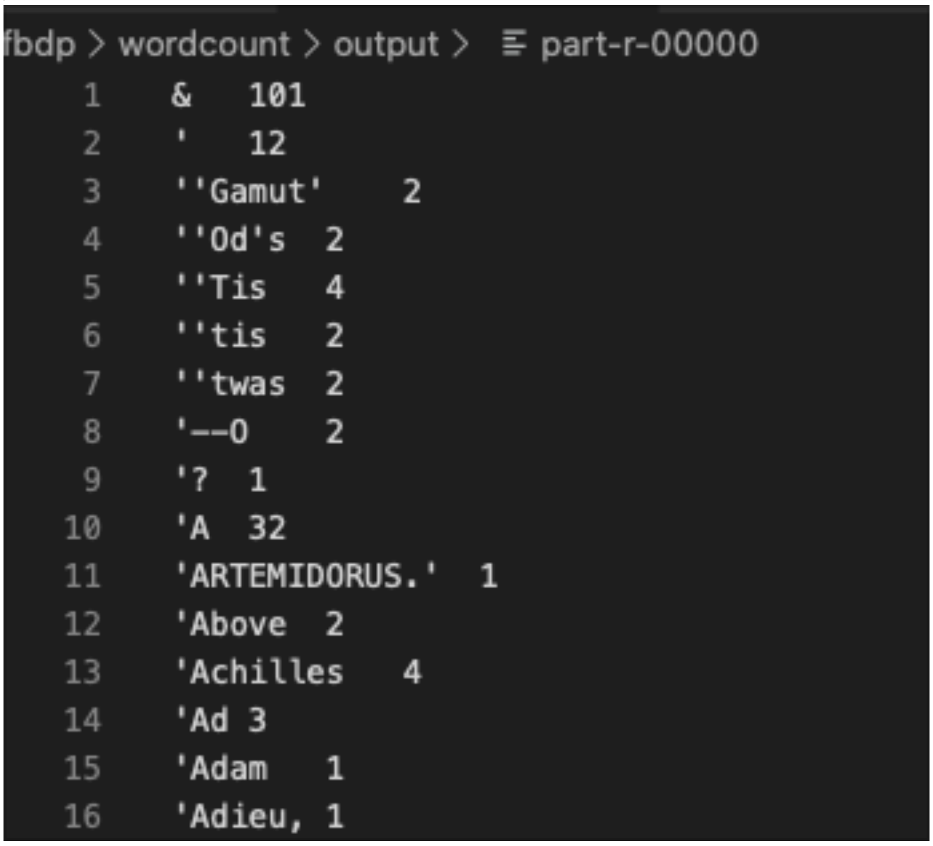
Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:8>

Show 20 entries　　Search:

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Progress	Tracking UI	Blacklisted Nodes
application_1603763878887_0001	root	word count 1.0	MAPREDUCE	default	Tue Oct 27 18:48:18 +0800 2020	Tue Oct 27 18:48:35 +0800 2020	FINISHED	SUCCEEDED		History	N/A

Showing 1 to 1 of 1 entries　　First Previous 1 Next Last



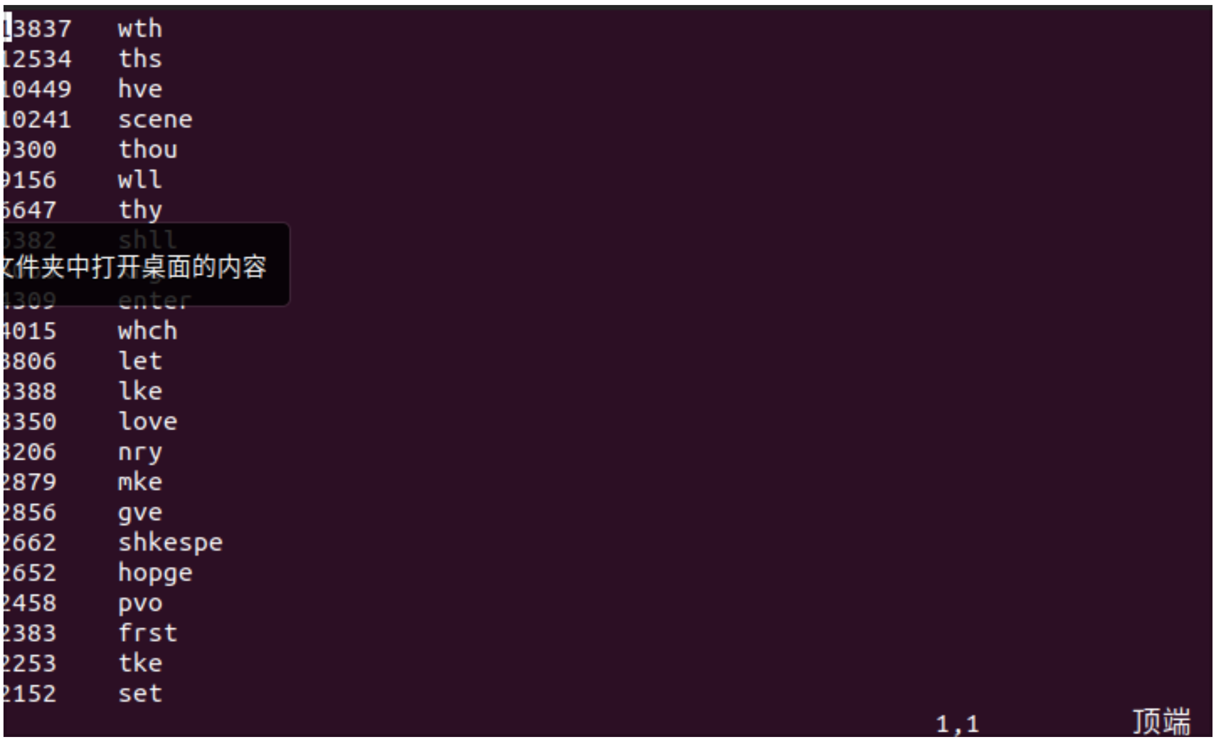
```
fbdp > wordcount > output > ≡ part-r-00000
&    101
'    12
''Gamut'    2
''Od's  2
''Tis   4
''tis   2
''twas  2
'--O    2
'?  1
'A  32
'ARTEMIDORUS.'  1
'Above  2
'Achilles   4
'Ad 3
'Adam   1
'Adieu, 1
```

2.0 版本

此为1.0版本结果截图，接着开始wordcount2.0版本，要求达到：忽略分词、停词，不区分大小写，忽略数字等。

由此，对1.0版本所作的修改为：

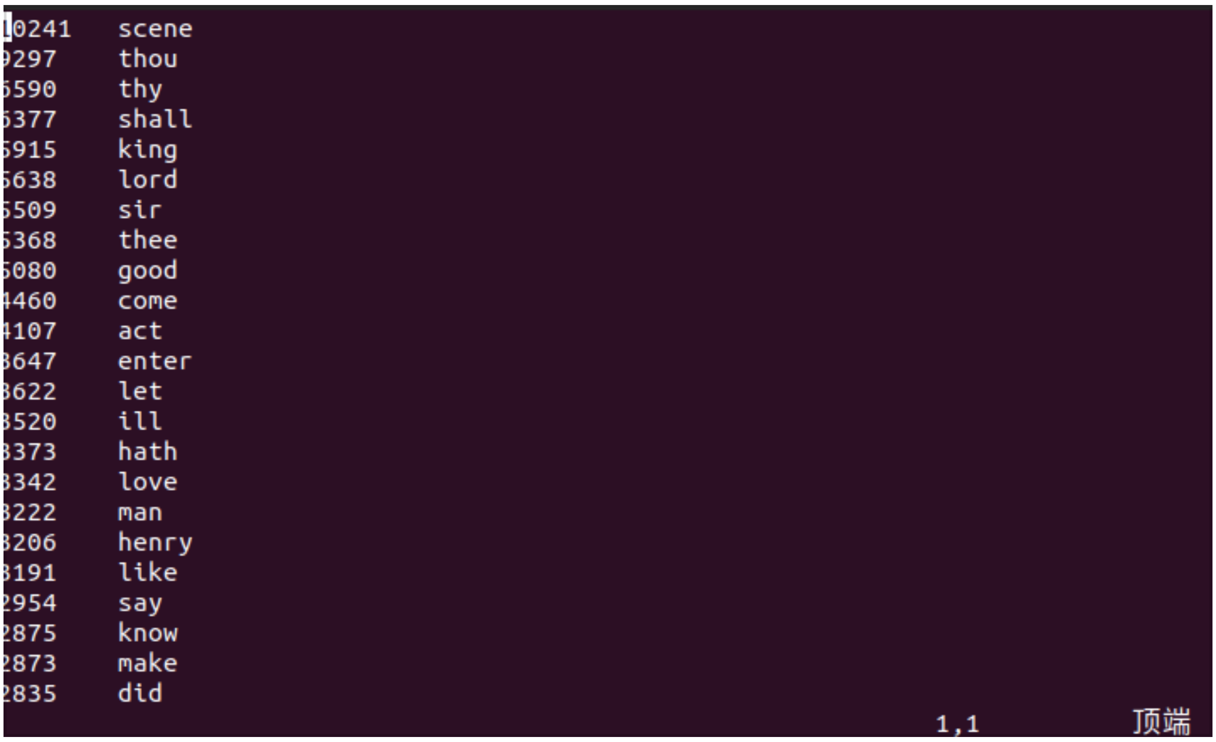
1. +setup() – 处理参数的输入，如caseSensitive的选择，skip文件的来源等
2. 在map()方法进行修改，利用正则表达式去除数字、字母以及排除停词，但是得到如下的结果，单词缺失字母。因为在stop-words-list中需要去除如'a'，'i'等短词，若使用replaceall方法，则会把长单词中此类字母也一起去掉。



```
13837   wth
12534   ths
10449   hve
10241   scene
9300    thou
9156    wll
6647    thy
5382    shll
文件夹中打开桌面的内容
4309    enter
4015    whch
3806    let
3388    lke
3350    love
3206    nry
2879    mke
2856    gve
2662    shkespe
2652    hopge
2458    pvo
2383    frst
2253    tke
2152    set
                                                   1,1           顶端
```

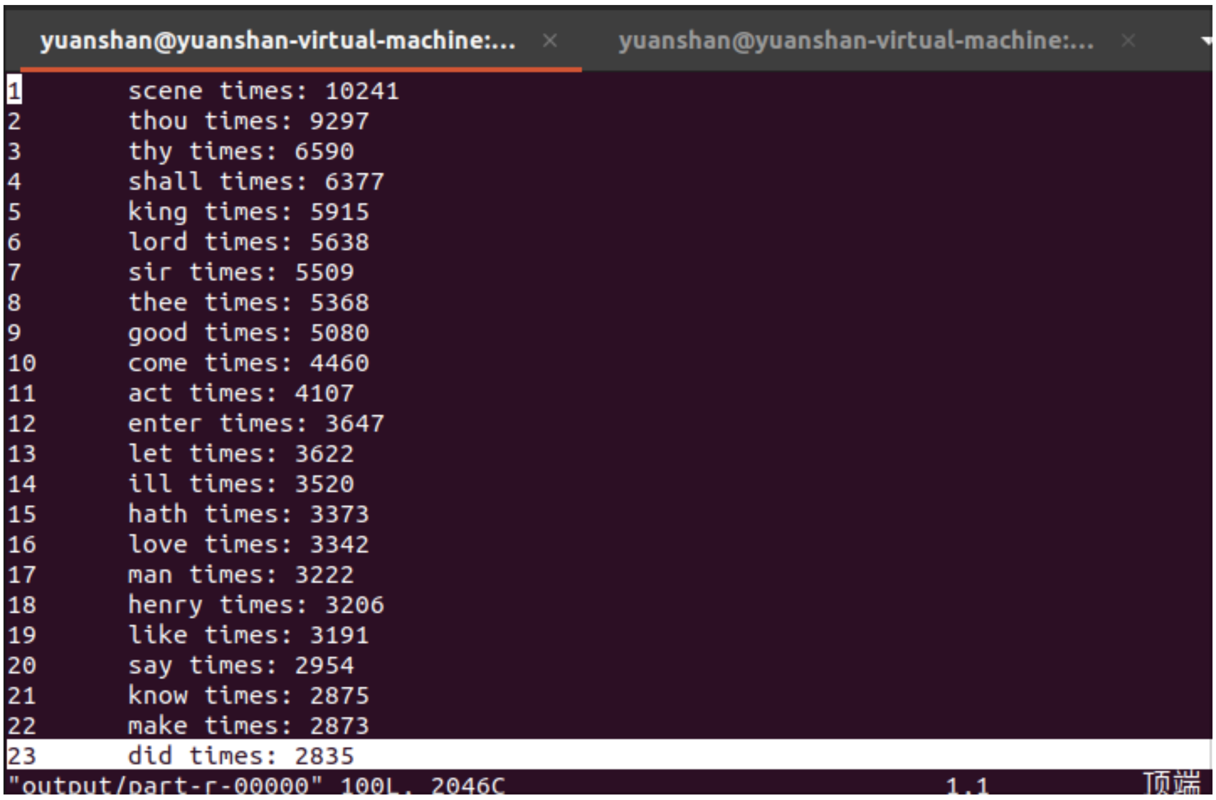
解决方法：利用正则表达式去掉字母和符号，利用tokenizer的特性可以分离不同的词，分离后判断单词长度且与停词比较是否要去掉或纳入。

```
String num = "[0-9]+";
String p = "\\pP";
line = line.replaceAll(num,"");
line = line.replaceAll(p,"");
StringTokenizer itr = new StringTokenizer(line);
while (itr.hasMoreTokens()) {
    boolean founded = false;
    String s = itr.nextToken();
    for (String pattern : patternToSkip) {
        if (s.equals(pattern)) {
            founded = true;
            break;
        }
    }
if (s.length() >= 3 && !founded) {
    word.set(s);
    context.write(word, one);
}
```



```
10241   scene
9297    thou
6590    thy
6377    shall
5915    king
5638    lord
5509    sir
5368    thee
5080    good
4460    come
4107    act
3647    enter
3622    let
3520    ill
3373    hath
3342    love
3222    man
3206    henry
3191    like
2954    say
2875    know
2873    make
2835    did
                                                              1,1          顶端
```

3. 实现倒序排列输出。原本按照1.0的方法得到一个输出，存储在临时文件夹，重新利用map读取原结果，利用inversemapper class可以改变自动sort按照单词字典序排序，转为按照value即单词计数来排序。同时在比较时引入`IntWritableDecreasingComparator`的override版本，正向排序可变为逆向排序（比较结果改为负数即可）
4. 按格式要求输出，则需要修改reducer类别，使输出的的key为排名，value为单词与计数的string。并修改main的输出reducer class。


yuanshan@yuanshan-virtual-machine:...
yuanshan@yuanshan-virtual-machine:...

```
1       scene times: 10241
2       thou times: 9297
3       thy times: 6590
4       shall times: 6377
5       king times: 5915
6       lord times: 5638
7       sir times: 5509
8       thee times: 5368
9       good times: 5080
10      come times: 4460
11      act times: 4107
12      enter times: 3647
13      let times: 3622
14      ill times: 3520
15      hath times: 3373
16      love times: 3342
17      man times: 3222
18      henry times: 3206
19      like times: 3191
20      say times: 2954
21      know times: 2875
22      make times: 2873
23      did times: 2835
"output/part-r-00000" 100L, 2046C                           1,1          顶端
```

本次实验的收获

1. 熟悉了maven创建操作以及bdkit的使用
2. 熟悉了git仓库操作与使用
3. 熟悉了mapreduce程序编写的方法