FBDB-lab3 实验报告

一、实验需求：

需求 1：针对股票新闻数据集中的新闻标题，编写 WordCount 程序，统计所有除 Stop-word（如“的”，“得”，“在”等）出现次数 k 次以上的单词计数，最后的结果按照词频从高到低排序输出。

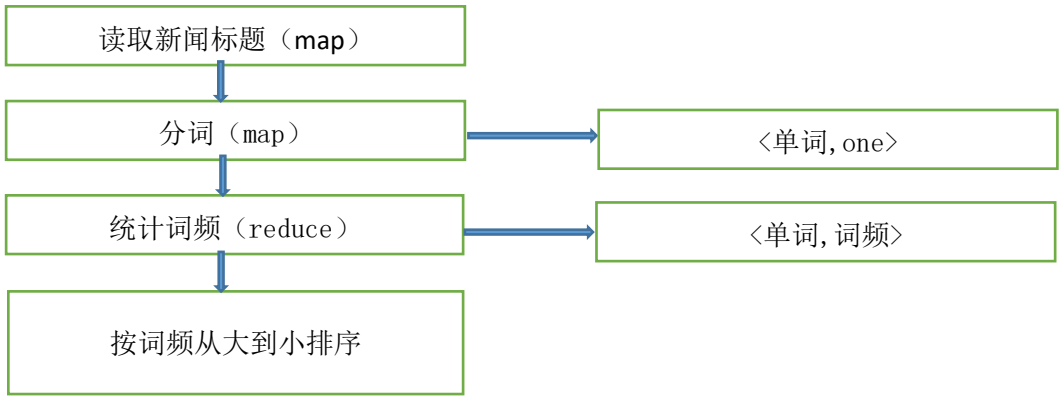
需求 2：针对股票新闻数据集，以新闻标题中的词组为 key，编写带 URL 属性和词频的文档倒排索引程序，并按照词频从大到小排序，将结果输出到指定文件。

注 1：可以用提供的 Stop-word 列表，也可以自行建立一个 Stop-word 列表，其中包含部分停词即可，不需要列出全部停词；参数 k 作为输入参数动态制定（如 k=10）

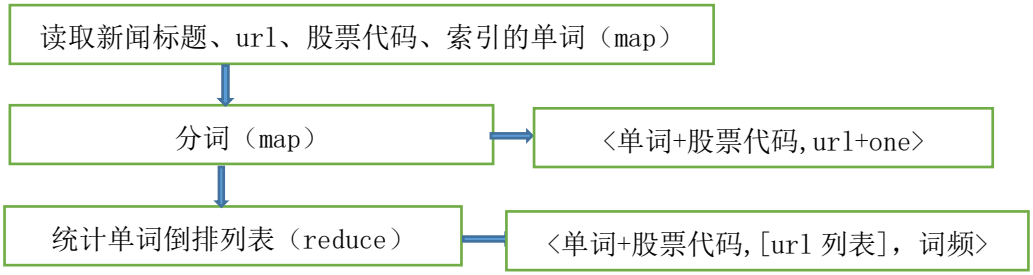
注 2：可以先在一个小数据集上调试通过，然后再对完整数据集进行倒排索引处理。

二、代码设计：

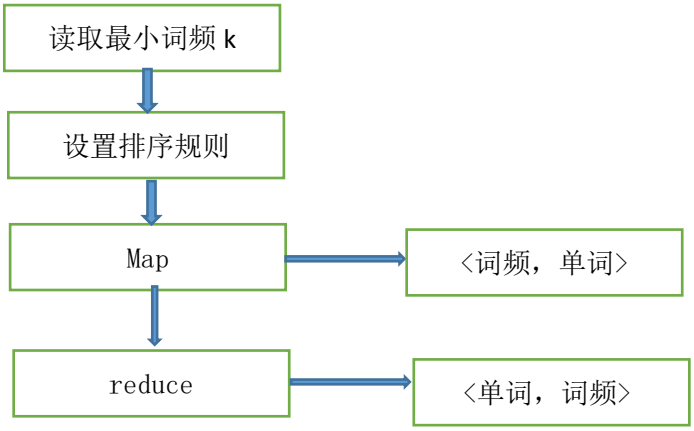
SegmentTool 类：完成对 fulldata.txt 的读取，读取全部标题文本，调用 hanlp 包进行分词，统计词频输出。



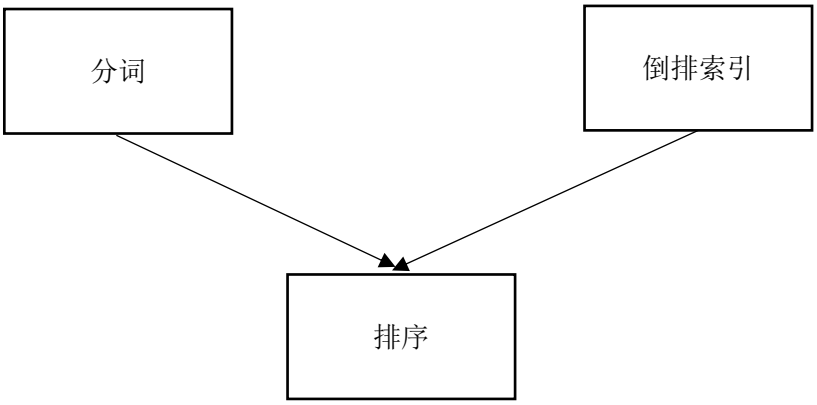
Invertindex 类，读取 fulldata.txt 并解析出新闻标题、url、股票代码。接受索引的单词，调用 hanlp 包进行分词，以<单词+股票代码，url+one>为<key,value>进行 map，以<单词+股票代码+[url 列表]，词频>作为<key,value>作为 reduce 输出。



SortByFrequency 类：调用 WritableComparator 类设置为倒序，设置 DescComparator 类制定比较规则，以上两个类的输出文件为输入，调换<key,value>为<词频，文本>,map,reduce<文本，词频>,接收参数 k，将词频大于 k 的输出。



三、实验设计思路：



四、实验过程及截图：

1.编译运行 SegmentTools，SortBySequence，输入参数 20，倒序输出词频大于 20 的词语到输出文件

```
[root@localhost hadoop-2.9.1]# bin/hadoop jar Sort.jar SortByFrequency /lab3/seg_output
 /lab3/sortedseg_output
Usage: seg.SortByFrequency <input> <output> k
[root@localhost hadoop-2.9.1]# bin/hadoop jar Sort.jar SortByFrequency /lab3/seg_output
 /lab3/sortedseg_output 20
18/11/25 16:56:14 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0
.0.1:8032
18/11/25 16:56:16 INFO input.FileInputFormat: Total input files to process : 1
18/11/25 16:56:16 INFO mapreduce.JobSubmitter: number of splits:1
18/11/25 16:56:16 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-p
ublisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
18/11/25 16:56:16 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_154090421
1849_0096
18/11/25 16:56:17 INFO impl.YarnClientImpl: Submitted application application_154090421
1849_0096
18/11/25 16:56:17 INFO mapreduce.Job: The url to track the job: http://localhost:8088/p
roxy/application_1540904211849_0096/
18/11/25 16:56:17 INFO mapreduce.Job: Running job: job_1540904211849_0096
18/11/25 16:56:27 INFO mapreduce.Job: Job job_1540904211849_0096 running in uber mode :
 false
18/11/25 16:56:27 INFO mapreduce.Job:  map 0% reduce 0%
18/11/25 16:56:34 INFO mapreduce.Job:  map 100% reduce 0%
18/11/25 16:56:41 INFO mapreduce.Job:  map 100% reduce 100%
```

2.查看结果

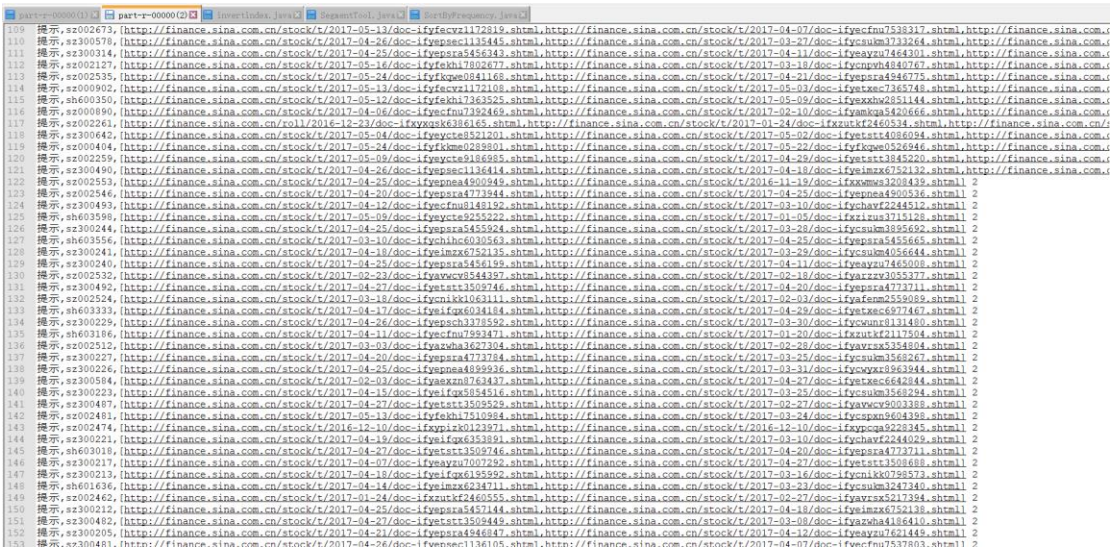
part-r-00000(1) | part-r-00000(2) | invertindex.java | SegmentTool.java | SortByFrequency.java

```
3043  拓维    21
3044  构建    21
3045  型材    21
3046  空降    21
3047  神马    21
3048  海药    21
3049  减仓    21
3050  观察    21
3051  风能    21
3052  谨防    21
3053  冰轮    21
3054  国盛    21
3055  按期    21
3056  海特    21
3057  性价比   21
3058  瑞丰    21
3059  人气    21
3060  安琪    21
3061  开元    21
3062  敢死队   21
3063  百倍    21
3064  京新    21
3065  批文    21
3066  马云    21
3067  红箭    21
3068  一元    21
3069  榕基    21
3070  成就    21
3071  控制系统    21
3072  圈钱    21
3073  巨龙    21
3074  贝尔    21
3075  探访    21
3076  特尔佳   21
3077  许继    21
3078  病因    21
3079  不俗    21
3080  宜昌    21
3081  总监    21
3082  自控    21
3083  抢占    21
3084  华鑫    21
3085  哈尔斯   21
3086  包装机   21
3087  普林    21
3088
```

3.编译运行 invertindex 和 SortBySequency， 接收参数“提示”，获取词语“提示”的倒排索引列表。

```
[root@localhost hadoop-2.9.1]# bin/hadoop jar Sort.jar SortByFrequency /lab3/invert_output /lab3/sortedinvert_output 0
18/11/25 17:07:50 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
18/11/25 17:07:53 INFO input.FileInputFormat: Total input files to process : 1
18/11/25 17:07:54 INFO mapreduce.JobSubmitter: number of splits:1
18/11/25 17:07:54 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
18/11/25 17:07:55 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1540904211849_0098
18/11/25 17:07:55 INFO impl.YarnClientImpl: Submitted application application_1540904211849_0098
18/11/25 17:07:55 INFO mapreduce.Job: The url to track the job: http://localhost:8088/proxy/application_1540904211849_0098/
18/11/25 17:07:55 INFO mapreduce.Job: Running job: job_1540904211849_0098
18/11/25 17:08:06 INFO mapreduce.Job: Job job_1540904211849_0098 running in uber mode : false
18/11/25 17:08:06 INFO mapreduce.Job:  map 0% reduce 0%
18/11/25 17:08:12 INFO mapreduce.Job:  map 100% reduce 0%
18/11/25 17:08:19 INFO mapreduce.Job:  map 100% reduce 100%
18/11/25 17:08:19 INFO mapreduce.Job: Job job_1540904211849_0098 completed successfully
[root@localhost hadoop-2.9.1]# bin/hadoop jar invert.jar invertindex /lab3/input /lab3/invert_output 提示
18/11/25 17:05:10 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
18/11/25 17:05:11 INFO input.FileInputFormat: Total input files to process : 1
18/11/25 17:05:11 INFO mapreduce.JobSubmitter: number of splits:1
18/11/25 17:05:11 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
18/11/25 17:05:12 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1540904211849_0097
18/11/25 17:05:12 INFO impl.YarnClientImpl: Submitted application application_1540904211849_0097
18/11/25 17:05:12 INFO mapreduce.Job: The url to track the job: http://localhost:8088/proxy/application_1540904211849_0097/
18/11/25 17:05:12 INFO mapreduce.Job: Running job: job_1540904211849_0097
18/11/25 17:05:22 INFO mapreduce.Job: Job job_1540904211849_0097 running in uber mode : false
18/11/25 17:05:22 INFO mapreduce.Job:  map 0% reduce 0%
18/11/25 17:05:41 INFO mapreduce.Job:  map 57% reduce 0%
18/11/25 17:05:42 INFO mapreduce.Job:  map 100% reduce 0%
18/11/25 17:05:49 INFO mapreduce.Job:  map 100% reduce 100%
18/11/25 17:05:49 INFO mapreduce.Job: Job job_1540904211849_0097 completed successfully
18/11/25 17:05:50 INFO mapreduce.Job: Counters: 49
```

4.查看结果：



五、实验结果分析：

1.分词效果较好，可以识别长度较长的词且基本没有无意义词，但有个别的其他符号没有处理掉。

2.有的高频词分布的文件比较分散，属于财经新闻常用的词；有的高频词分布计较集中，属于个别股票专有词。

六、实验总结：

1.实验考验对数据文件的解析与噪音的处理，通过判断每行数据数量判断和正则匹配清除不规则数据和乱码的干扰。

2.程序的设计需要同时传入 String 参数和 int 参数，需要灵活应用 conf.set()和 conf.get()

3.要深入理解 mapreduce 的工作原理，才能灵活的设定 map 的键值对，达到希望的效果。比如倒排索引 invertindex 类，股票代码和词语为定值，url 和词频为需要 map 的值，因此设置为<单词+股票代码，url+one>

4.开源的中文分词类 hanlp 只需在官网下载相关文件，然后引用相关类即可，本实验使用基础分词方法，HanLP.segment()函数。

5. 当 map 和 reduce 的输出类型不同时，比如 invertindex 中 Mapper<LongWritable,Text,Text,Text>和 Reducer<Text,Text,Text,IntWritable>，需要先设置 job.setReducerClass 和 job.setOutputKeyClass，再设置 map 的输出类型 job.setMapOutputKeyClass 和 job.setMapOutputValueClass 来覆盖，并将 job.setCombinerClass(SegReducer.class);注释掉才能正常运行。

6.排序时调用 WritableComparator 类，制定比较规则来获得希望的排序结果。

7.程序存在运行流程复杂且每次倒排索引都要全部运行一遍的缺点，有待改进。每次都要输出完文件后才能重新读入排序，造成性能的不足。

七、代码及注释：

已上传到 github，见文件夹 src