MapReduce 基础编程实验报告

151180045 侯汶昕

1. 实验目的

利用 MapReduce 编程实现对给定格式的文本文件进行基本的分词，词频统计，倒排索引和二次排序的操作

2. 实验重点与难点

1. 文本分词和去除停用词
2. 倒排索引
3. 降序排序及二次排序

3. 实验环境

实验中使用了两台虚拟机，虚拟机具有相同的环境和配置：

系统：ubuntu 16.04 LTS 64-bit

内存：1.9GB

4. 实验要求

4.1 针对股票新闻数据集中的新闻标题，去除其中的停用词（Stop-Word）后，对出现次数为 k 以上的单词进行词频统计，结果按词频从高到底排序输出。

4.2 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。

规定输出格式为：

高速公路, 10, 股票代码, [url0, url1,...,url9]

5. 实验设计思路

该实验要求实现两个需求，考虑分别使用两个 MapReduce 程序进行实现。

5.1 中文词频统计

第一个需求的基本框架依然是 WordCount，有三个重点：

第一个重点也是一个难点，那就是对中文进行词频统计。词频统计的前提是分词，相对可以简单的依托空格来分词的英文来说，中文的分词明显要难得多。在尝试自己写程序进行分词失败后，我还是决定使用开源的 HanLP 中文自然语言处理包进行分词（感谢 HanLP）。

第二个重点在于统计除停用词以外的出现 k 次以上的单词。去除停用词的功能依然借助了 HanLP 进行实现（再次感谢 HanLP）。只统计 k 次以上的单词是比较简单的，只需要对统计完

的停用词出现次数进行一个条件判断，将出现次数大于 k 次的键值对写入 context 就行。

第三个重点在于要求将最后结果按照词频从高到低进行排序输出。考虑将词频统计完的结果输出到一个临时文件后，再新建一个 Job 对其进行倒序排序。

结合上述分析，最终实现的程序的大体框架是在 Map 阶段对于输入利用 HanLP 进行分词、去除停用词等基本操作，并以<word，1>键值对（Key-Value）的方式传输给 Combiner 和 Reducer。在 Combiner 和 Reducer 中以单词进行划分，对 Value 进行求和得到总出现次数 Sum，并将和大于 k 的键值对<word, Sum>输出得到一个临时文件。再利用系统给出的 InverseMapper 类将临时文件中的<word, Sum>进行键值对交换以便对 Sum 进行排序，再重写一个降序排序的 IntWritableDecreasingComparator 类并利用 setSortComparatorClass 进行倒序排序，最后输出到文件即可实现第一个需求。

5.2 新闻倒排索引

第二个需求相比第一个需求要难一些。由于输出格式受限，我首先就是考虑 Key-Value 对如何设计。

在 Map 阶段，Value 输入格式为：

股票代码 Code\t 股票名称\t 日期\t 时间\t 新闻标题 Title\t 链接 URL

在 Map 阶段对 Title 进行分词、去除停用词等基本操作后得到了 TitleWords，考虑到我下一步要进行词频统计，并且是按照不同的 Code 和不同的 TitleWord 进行统计，所以我决定将 Code 和 TitleWord 进行组合成 Key，URL 和出现次数 1 进行组合作为 Value，剩下的没用信息直接丢弃。于是 Map 的输出键值对为：

<TitleWord#Code, URL#1>

这样可以保证同一个 Code 下的同一个 TitleWord 会被分到一起进行 Reduce。

那么在 Reducer 中，我就可以对每一个股票下的每一个单词进行词频统计得到 Sum，在实施过程中我还发现其实这里可以顺便把 URL 也拼起来，于是 Reducer 的输出键值对为：

<TitleWord#Code#[URL0, URL1, ...], Sum>

这样就得到了最后输出所需要的所有信息了，剩下要做的就是倒序排序和调整输出格式。于是我决定使用和上面第一个需求类似的方法，将刚才 Reducer 的输出放在一个临时文件中，再新建一个 Job（SecondSort）对其进行格式重组和排序。

在新建 Job 的 Map 阶段，输入键值对格式为<Text, IntWritable>，内容和上面的 Reducer 输出一致，考虑反正要格式调整，不如在这里就把格式调一下，于是没有别的操作，仅仅更换了输出时的键值对格式为<Text, NullWritable>，键值对内容为：

<TitleWord#Sum#Code#[URL0, URL1, ...], NullWritable>

因为最后的结果要将 TitleWord 一致的放在一起，所以还需要重写一个 SecondSortPartitioner，依据 TitleWord 而不是全部的 Key 进行划分。然后和第一个需求类似，重写一个基于二次排序并且是降序排序的 SecondSortComparator 类并利用 setSortComparatorClass 进行倒序排序，首先依据字符串形式的 TitleWord 进行排序，在 TitleWord 一致的情况下依据整型的 Sum 进行倒序排序。

在 Reduce 阶段，将 Mapper 输出键值对中的所有"#"替换成",“，再输出到文件即可完成第二个需求。

6. 重要类的设计说明

6.1 中文词频统计

6.1.1 MainReducer 类

MainReducer 类是第一个 Job 的关键任务，主要实现的功能为词频统计和 URL 的拼接，由于便于后续处理的分割，采用“#”作为连接符。

```
public static class MainReducer extends
   Reducer<Text, Text, Text, IntWritable> {
      private IntWritable result = new IntWritable();

      public void reduce(Text key, Iterable<Text> values, Context context)
            throws IOException, InterruptedException {
         int sum = 0;
         Text word = new Text();
         String titleWord = key.toString().split("#")[0];
         String code = key.toString().split("#")[1];
         String URL;
         StringBuilder strBuilder = new StringBuilder("[");
         int Occurrence;
         for (Text val : values) {
           URL = val.toString().split("#")[0];
           strBuilder.append(URL+", ");
           Occurrence = Integer.parseInt(val.toString().split("#")[1]);
           sum += Occurrence;
         }
         strBuilder.replace(strBuilder.length()-2, strBuilder.length(), "]");
         word.set(titleWord + "#" + code + "#" + strBuilder.toString());
         result.set(sum);
         context.write(word, result);
         }
   }
```

6.1.1 InverseMapper 类

由于 MapReduce 默认按照 Key 进行排序，为了使最后结果按词频进行排序，需要调用 InverseMapper 类。该类是系统自带的类，用于调换 Key 和 Value，输出<Value, Key>

该类源码如下：

```
public class InverseMapper<K, V> extends
```

```
        MapReduceBase implements Mapper<K, V, V, K> {
    /** The inverse function.    Input keys and values are swapped.*/
    public void map(K key, V value,
                    OutputCollector<V, K> output, Reporter reporter)
      throws IOException {
      output.collect(value, key);
    }
}
```

6.1.2 IntWritableDecreasingComparator 类

由于 MapReduce 默认对 Key 进行升序排序，因此为了实现降序排序，需要重写一个 IntWritableDecreasingComparator 类。该类通过继承 IntWritable.Comparator 类，将父类 compare 函数返回的结果分别取反，从而实现降序排序。

```
private static class IntWritableDecreasingComparator extends IntWritable.Comparator{
    public int compare(WritableComparable a, WritableComparable b){
        return -super.compare(a, b);
    }
    public int compare(byte[] b1, int s1, int l1, byte[] b2, int s2, int l2){
        return -super.compare(b1, s1, l1, b2, s2, l2);
    }
}
```

6.2 新闻倒排索引

6.2.1 SecondSortPartitioner 类

由于 MapReduce 默认是按照相同的完整的 Key 来分区，而在 Mapper 阶段之后输出的格式为 <TitleWord#Sum#Code#[URL0, URL1, ...], NullWritable>。为了保证最后同一个 TitleWord 能放在一起并且正常排序，这里需要重写一个 Partitioner，实现仅针对 TitleWord 进行分区。

```
public static class SecondSortPartitioner extends HashPartitioner<Text, NullWritable>
  {
    //Partitioned by titleWord
    public int getPartition(Text key, NullWritable value, int numReduceTasks){
        String term = new String();
        term = key.toString().split("#")[0];
        return super.getPartition(new Text(term), value, numReduceTasks);
    }
}
```

6.2.2 SecondSortComparator 类

由于要实现二次排序和依据词频降序排序两个功能，所以这里必须重写一个 Comparator 类。该类首先比较 TitleWord，如果不一致则比较 TitleWord（TitleWord 并不需要反向排序）；如果一致则比较单词的词频，并且返回与正常的 Compare 函数相反的结果从而实现降序排序。

```
private static class SecondSortComparator extends WritableComparator{
        protected SecondSortComparator(){
            super(Text.class, true);
        }
        @Override
        public int compare(WritableComparable a, WritableComparable b) {
                Text o1 = (Text) a;
                Text o2 = (Text) b;
                int occurrence1 = Integer.parseInt(o1.toString().split("#")[1]);
                String titleWord1 = o1.toString().split("#")[0];
                int occurrence2 = Integer.parseInt(o2.toString().split("#")[1]);
                String titleWord2 = o2.toString().split("#")[0];
                if(! titleWord1.equals(titleWord2))
                    return titleWord1.compareTo(titleWord2);
                else
                    return occurrence2 - occurrence1;
          }
}
```

7. 实验结果及截图

7.1 中文分词结果

```
houwenxin@ubuntu:~/hadoop_installs/hadoop-2.9.1$ bin/hadoop jar /home/houwenxin/Desktop/HWX_wcCN.jar wcCN input output 30
18/11/13 23:27:30 INFO client.RMProxy: Connecting to ResourceManager at host-0/127.0.0.1:8032
18/11/13 23:27:32 INFO input.FileInputFormat: Total input files to process : 1
18/11/13 23:27:33 WARN hdfs.DataStreamer: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1252)
        at java.lang.Thread.join(Thread.java:1326)
        at org.apache.hadoop.hdfs.DataStreamer.closeResponder(DataStreamer.java:980)
        at org.apache.hadoop.hdfs.DataStreamer.endBlock(DataStreamer.java:630)
        at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:807)
18/11/13 23:27:33 INFO mapreduce.JobSubmitter: number of splits:1
18/11/13 23:27:33 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
18/11/13 23:27:33 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1542180344850_0001
18/11/13 23:27:34 INFO impl.YarnClientImpl: Submitted application application_1542180344850_0001
18/11/13 23:27:34 INFO mapreduce.Job: The url to track the job: http://ubuntu:8088/proxy/application_1542180344850_0001/
18/11/13 23:27:34 INFO mapreduce.Job: Running job: job_1542180344850_0001
18/11/13 23:27:49 INFO mapreduce.Job: Job job_1542180344850_0001 running in uber mode : false
18/11/13 23:27:49 INFO mapreduce.Job:  map 0% reduce 0%
18/11/13 23:28:10 INFO mapreduce.Job:  map 49% reduce 0%
18/11/13 23:28:15 INFO mapreduce.Job:  map 100% reduce 0%
18/11/13 23:28:38 INFO mapreduce.Job:  map 100% reduce 100%
18/11/13 23:28:39 INFO mapreduce.Job: Job job_1542180344850_0001 completed successfully
18/11/13 23:28:39 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=44272
                FILE: Number of bytes written=484311
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=20980068
                HDFS: Number of bytes written=66003
                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
                Total time spent by all maps in occupied slots (ms)=23666
                Total time spent by all reduces in occupied slots (ms)=18770
                Total time spent by all map tasks (ms)=23666
                Total time spent by all reduce tasks (ms)=18770
                Total vcore-milliseconds taken by all map tasks=23666
                Total vcore-milliseconds taken by all reduce tasks=18770
```

```
houwenxin@ubuntu:~/hadoop_installs/hadoop-2.9.1$ bin/hdfs dfs -cat output/*
28274   公告
18416   股份有限公司
15533   日
12665   公司
9107    上市公司
8686    %
8449    提示
8066    股份
7570    沪
7410    深
7349    股市
7318    业绩
6963    导航
6907    股海
6777    板块
6189    股东
5943    涨停
5796    科技
5036    晚间
4659    机构
4639    速递
4638    集团
4505    成
4443    A
4437    快讯
4431    线
4120    超
3956    概念股
3893    年度
3539    中
3503    拟
3446    资金
3429    新
3211    投资
3168    控股
2940    报告
2914    行业
2856    涨
2840    看好
2814    近
```

```
31      信飞
31      英茂
31      固
31      喜爱
31      门道
31      索具
31      出自
31      红箭
31      售价
31      蠕虫
31      茂硕
31      闻
31      韵
31      横店
31      横盘
31      森美
31      启幕
31      氧
31      果汁
31      服装公司
31      服装业
31      中介
31      名义
31      流感
31      冀凯
31      更大
31      涨升
31      世嘉
31      趣
31      星期六
31      漏洞
31      缓解
31      旭
31      新黄浦
31      灵
31      0
31      风景
31      远望
31      落空
31      新药
31      合金
31      司法
31      整顿
31      特质
31      支出
31      临床试验
31      探索
houwenxin@ubuntu:~/hadoop_installs/hadoop-2.9.1$
```

程序基本实现了词频统计和倒序排序功能，不过由于一些分词的原因，有一些符号也被统计了进来。测试时 k 取值为 30。

7.2 新闻倒排索引结果

```
houwenxin@ubuntu:~/hadoop_installs/hadoop-2.9.1$ bin/hadoop jar /home/houwenxin/Desktop/HWX_InvertedIndex.jar InvertedIndex input output
18/11/15 00:32:45 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
18/11/15 00:32:45 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
18/11/15 00:32:46 INFO input.FileInputFormat: Total input files to process : 1
18/11/15 00:32:46 INFO mapreduce.JobSubmitter: number of splits:1
18/11/15 00:32:47 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local347624016_0001
18/11/15 00:32:47 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
18/11/15 00:32:47 INFO mapreduce.Job: Running job: job_local347624016_0001
18/11/15 00:32:47 INFO mapred.LocalJobRunner: OutputCommitter set in config null
18/11/15 00:32:47 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
18/11/15 00:32:47 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
18/11/15 00:32:47 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
18/11/15 00:32:47 INFO mapred.LocalJobRunner: Waiting for map tasks
18/11/15 00:32:47 INFO mapred.LocalJobRunner: Starting task: attempt_local347624016_0001_m_000000_0
18/11/15 00:32:47 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
18/11/15 00:32:47 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
18/11/15 00:32:48 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
18/11/15 00:32:48 INFO mapred.MapTask: Processing split: hdfs://host-0:9000/user/houwenxin/input/fulldata.txt:0+20979951
18/11/15 00:32:48 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
18/11/15 00:32:48 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
18/11/15 00:32:48 INFO mapred.MapTask: soft limit at 83886080
18/11/15 00:32:48 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
18/11/15 00:32:48 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
18/11/15 00:32:48 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
18/11/15 00:32:48 INFO mapreduce.Job: Job job_local347624016_0001 running in uber mode : false
18/11/15 00:32:48 INFO mapreduce.Job:  map 0% reduce 0%
18/11/15 00:33:00 INFO mapred.LocalJobRunner: map > map
18/11/15 00:33:00 INFO mapred.MapTask: Spilling map output
18/11/15 00:33:00 INFO mapred.MapTask: bufstart = 0; bufend = 71002656; bufvoid = 104857600
18/11/15 00:33:00 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 22993548(91974192); length = 3220849/6553600
18/11/15 00:33:00 INFO mapred.MapTask: (EQUATOR) 74229040 kvi 18557256(74229024)
18/11/15 00:33:00 INFO mapreduce.Job:  map 50% reduce 0%
18/11/15 00:33:05 INFO mapred.MapTask: Finished spill 0
18/11/15 00:33:05 INFO mapred.MapTask: (RESET) equator 74229040 kv 18557256(74229024) kvi 17751700(71006800)
18/11/15 00:33:05 INFO mapred.LocalJobRunner: map > map
18/11/15 00:33:05 INFO mapred.MapTask: Starting flush of map output
18/11/15 00:33:05 INFO mapred.MapTask: Spilling map output
18/11/15 00:33:05 INFO mapred.MapTask: bufstart = 74229040; bufend = 93351536; bufvoid = 104857600
18/11/15 00:33:05 INFO mapred.MapTask: kvstart = 18557256(74229024); kvend = 17690008(70760032); length = 867249/6553600
18/11/15 00:33:06 INFO mapred.LocalJobRunner: map > sort
18/11/15 00:33:06 INFO mapred.MapTask: Finished spill 1
18/11/15 00:33:06 INFO mapred.Merger: Merging 2 sorted segments
18/11/15 00:33:06 INFO mapred.Merger: Down to the last merge-pass, with 2 segments left of total size: 92169187 bytes
18/11/15 00:33:06 INFO mapreduce.Job:  map 67% reduce 0%
```

```
houwenxin@ubuntu:~/hadoop_installs/hadoop-2.9.1$ bin/hdfs dfs -cat output/* | grep 高速公路
高速公路, 23, sh600035, [http://finance.sina.com.cn/stock/t/2017-04-18/doc-ifyeimzx6752178.shtml, http://finance.sina.com.cn/stock/t/2017-01-11/doc-ifxzkfuh6804456.shtml, http://finance.sina.com.cn/stock/t/2017-01-18/doc-ifxzqnva3906319.shtml, http://finance.sina.com.cn/stock/t/2017-04-19/doc-ifyeifqx6354378.shtml, http://finance.sina.com.cn/stock/t/2017-04-20/doc-ifyepnea4220485.shtml, http://finance.sina.com.cn/stock/t/2017-05-11/doc-ifyfecvz0861871.shtml, http://finance.sina.com.cn/stock/t/2017-04-07/doc-ifyeayzu7006668.shtml, http://finance.sina.com.cn/stock/t/2017-04-06/doc-ifyeayzu6860042.shtml, http://finance.sina.com.cn/stock/t/2017-01-24/doc-ifxzunxf1917217.shtml, http://finance.sina.com.cn/stock/t/2017-02-16/doc-ifyarref5397683.shtml, http://finance.sina.com.cn/stock/t/2017-03-31/doc-ifycwyxr8962649.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/t/2017-05-20/doc-ifyfkqwe0379550.shtml, http://finance.sina.com.cn/stock/t/2017-02-07/doc-ifyaexzn9116272.shtml, http://finance.sina.com.cn/stock/t/2017-04-29/doc-ifyetstt3845573.shtml, http://finance.sina.com.cn/stock/t/2017-05-10/doc-ifyexxhw3005167.shtml, http://finance.sina.com.cn/stock/t/2017-02-25/doc-ifyavrsx5018812.shtml, http://finance.sina.com.cn/stock/t/2017-02-24/doc-ifyavwcv8667141.shtml, http://finance.sina.com.cn/stock/t/2017-05-17/doc-ifyfekhi7954651.shtml, http://finance.sina.com.cn/stock/t/2017-02-21/doc-ifyarrcc8300587.shtml, http://finance.sina.com.cn/stock/hyyj/2017-07-31/doc-ifyinwmp0911976.shtml, http://finance.sina.com.cn/stock/t/2017-05-10/doc-ifyexxhw3005167.shtml, http://finance.sina.com.cn/stock/t/2017-04-13/doc-ifyeimzx6070793.shtml]
高速公路, 16, sh600269, [http://finance.sina.com.cn/stock/t/2017-01-13/doc-ifxzqhka2845736.shtml, http://finance.sina.com.cn/stock/t/2017-04-19/doc-ifyeimzx6910966.shtml, http://finance.sina.com.cn/stock/t/2017-04-13/doc-ifyeimzx6070612.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/t/2017-04-11/doc-ifyecfnu7993248.shtml, http://finance.sina.com.cn/stock/t/2017-04-07/doc-ifyeayzu7006974.shtml, http://finance.sina.com.cn/stock/t/2017-03-29/doc-ifycsukm4056361.shtml, http://finance.sina.com.cn/stock/t/2017-03-15/doc-ifychavf2805331.shtml, http://finance.sina.com.cn/stock/t/2017-04-25/doc-ifyepsra5456459.shtml, http://finance.sina.com.cn/stock/t/2017-05-17/doc-ifyfekhi7955241.shtml, http://finance.sina.com.cn/stock/t/2017-02-15/doc-ifyamkqa6211189.shtml, http://finance.sina.com.cn/stock/t/2017-05-12/doc-ifyfekhi7363522.shtml, http://finance.sina.com.cn/stock/t/2016-12-24/doc-ifxyxqsk6501181.shtml, http://finance.sina.com.cn/stock/t/2017-02-08/doc-ifyafqxp5909772.shtml, http://finance.sina.com.cn/stock/t/2017-01-03/doc-ifxzczsu6613247.shtml, http://finance.sina.com.cn/stock/t/2017-01-21/doc-ifxzunxf1642564.shtml]
高速公路, 13, sz000886, [http://finance.sina.com.cn/stock/t/2016-08-23/doc-ifxvctcc8272934.shtml, http://finance.sina.com.cn/stock/t/2017-05-22/doc-ifyfkqwe0527101.shtml, http://finance.sina.com.cn/stock/t/2017-04-25/doc-ifyepnea4901297.shtml, http://finance.sina.com.cn/stock/t/2017-01-19/doc-ifxzunxf1396785.shtml, http://finance.sina.com.cn/roll/2016-11-12/doc-ifxxsfip4528967.shtml, http://finance.sina.com.cn/stock/t/2017-04-15/doc-ifyeifqx5854466.shtml, http://finance.sina.com.cn/stock/t/2016-09-06/doc-ifxvpxua7954891.shtml, http://finance.sina.com.cn/stock/t/2017-04-18/doc-ifyeifqx6196325.shtml, http://finance.sina.com.cn/roll/2016-11-12/doc-ifxxsfip4528967.shtml, http://finance.sina.com.cn/stock/t/2016-10-28/doc-ifxxfuff6998896.shtml, http://finance.sina.com.cn/roll/2016-08-31/doc-ifxvitex9344266.shtml, http://finance.sina.com.cn/stock/hyyj/2017-07-31/doc-ifyinwmp0911976.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml]
高速公路, 12, sh601107, [http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-02-14/doc-ifyameqr7509994.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-08-10/doc-ifyixipt0866329.shtml, http://finance.sina.com.cn/stock/hyyj/2017-07-31/doc-ifyinwmp0911976.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-08-01/doc-ifyinwmp1285001.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-08-14/doc-ifyixipt1668353.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-27/doc-ifyetxec6769975.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-05-11/doc-ifyfeivp5597410.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-21/doc-ifyepnea4485363.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-20/doc-ifyepsra4899237.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-12/doc-ifyeimzx6027637.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-10/doc-ifyecfnu7966161.shtml]
高速公路, 9, sh600012, [http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-06-28/doc-ifyhmtek7867601.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-28/doc-ifyetwtf8618017.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-07-03/doc-ifyhrttz2199302.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-28/doc-ifyetxec6851103.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-19/doc-ifyepnea4144539.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-12/doc-ifyeimzx6022273.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-11/doc-ifyeayzu7575890.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-07/doc-ifyecfnu7641551.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-06-30/doc-ifyhryex5625314.shtml]
高速公路, 8, sh600377, [http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-13/doc-ifyeifqx5633633.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-07-26/doc-ifyinryq6279901.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-07-07/doc-ifyhweua4307597.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-07-04/doc-ifyhrttz2371519.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-06-23/doc-ifyhmtcf2784654.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-28/doc-ifyetxec6953769.shtml, http://finance.sina.com.cn/stock/hkstock/ggscyd/2017-04-28/doc-ifyetstt3822095.shtml]
高速公路, 6, sh601518, [http://finance.sina.com.cn/stock/t/2016-06-06/doc-ifxsvexw8523796.shtml, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/jsy/20160606/113024786705.shtml, http://finance.sina.com.cn/stock/e/2016-11-21/doc-ifxxwrwh4824314.shtml, http://finance.sina.com.cn/stock/t/20151210/023023977809.shtml, http://finance.sina.com.cn/stock/t/20151210/070423980510.shtml]
高速公路, 4, sz000429, [http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/t/2017-04-08/doc-ifyeayzu7179690.shtml, http://finance.sina.com.cn/stock/hyyj/2017-07-31/doc-ifyinwmp0911976.shtml, http://finance.sina.com.cn/stock/t/2017-04-13/doc-ifyeifqx5516507.shtml]
高速公路, 3, sh600548, [http://cj.sina.com.cn/article/detail/583552730/231460, http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml, http://finance.sina.com.cn/stock/hyyj/2017-07-31/doc-ifyinwmp0911976.shtml]
高速公路, 2, sz000885, [http://finance.sina.com.cn/stock/s/2017-07-07/doc-ifyhvyie0431981.shtml, http://finance.sina.com.cn/stock/t/2017-07-06/doc-ifyhvyie0404002.shtml]
高速公路, 1, sz000900, [http://finance.sina.com.cn/roll/2017-07-25/doc-ifyihmmm8448415.shtml]
```

程序基本实现了新闻的倒排索引，并且能够把同一个词放在一起，按照降序排序进行输出。图示为测试时尝试搜索了“高速公路”这个词，得到的结果。

8. 实验中可以改进之处

本实验中，我完成了实验所要求的两个任务：面向中文的词频统计和针对金融新闻的倒排索引，并且基本完成了所有需求。但是该实验还有几个可以改进优化的地方：

1. 这两个需求在的实现和功能都是十分类似的，因而可以考虑把他们做成一个程序，并且还可以通过定制参数对功能进行选择（比如只进行词频统计），简化用户操作。

2. 该实验由于使用了开源的分词包 HanLP，并且使用的词典没有对金融专业的词语，符号进行特别设计，所以分词过程中还是有一些专业的词语或者符号被拆开来了，后续可以考虑通过应用自己的词典，或者将这些词语添加进默认词典。

3. 我在实验中尝试过使用自己的 Stop-Word 表和分词词典，但是由于 MapReduce 程序直接从 hdfs 上读取文件的效率太低，只能放弃，后续或许可以通过使用 DistributedCache 的方法加快文件读取的运行效率，从而实现使用自己的词表。