0、作业信息

how your work is done and what problem you have faced. Also the code for wordcount.

Please create a 5 nodes cluster on Netease with (1 nodenode/jobtracker, 4 datanodes/tasktracker).

Download any version of Hadoop and deploy on the cluster.

Write a simple wordcount program to calculate the word frequency for the files about spamming emails (check the course content) and run in the cluster.

For deployment details, please check the attachment.

1、网易云服务器配置

1.1、创建服务器

首先在网易云平台创建四台规格为n1，1核CPU，2GB内存，20GBSSD的云服务器。之后为hadoop1绑定公网ip59.111.99.128，用我们的电脑登陆这台公网的hadoop1主机。

1.2、配置免密登陆hadoop主机

为了使我们的电脑能够SSH免密登陆hadoop1主机，我们电脑需要创建自己的私钥和公钥，私钥保存在自己的电脑中，公钥传给hadoop1主机，同理，想要hadoop1访问hadoop2，3，4，同样需要生成hadoop1的公钥和私钥，并把hadoop1的公钥和私钥传给hadoop2，3，4。

配置本机免密登陆hadoop1的步骤如下：在终端输入

```
ssh-keygen -t rsa
```

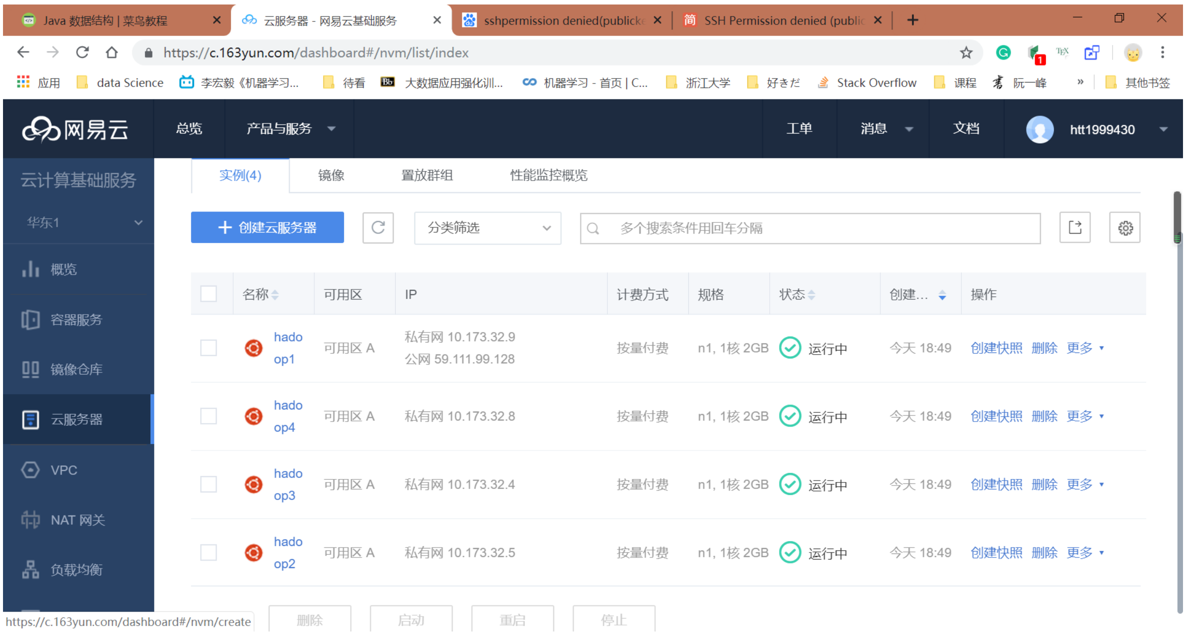
生成公钥，私钥，此时使用ls命令查看保存路径下的文件，可以发现多了id_rsa（私钥）和id_rsa.pub（公钥）。将公钥中的内容复制到网易云密钥管理模块的公钥内容中，再回到云服务器界面，确定已保存的密钥，便可以本机免密登陆hadoop1了。

配置hadoop1免密登陆hadoop2，3，4的步骤如下：同理，生成hadoop1的公钥和私钥，复制hadoop1的公钥内容到SSH密钥管理中，对hadoop2，3，4分别更改SSH密钥，添加hadoop1的公钥，设置完成后，hadoop1就可以免密登陆hadoop2，3，4了。

1.3、配置通过主机名登陆

为了便于通过hadoop1主机直接登陆其他主机，我们需要配置成直接ssh主机名登陆，配置步骤如下：修改hosts，编辑hadoop1中的/etc/hosts，在其中添加四台主机的内网ip和名称，建立主机名到ip的映射，此时hadoop1便可以直接用ssh 主机名的命令来访问其他主机了。之后使用scp命令键hosts文件发送到其他节点即可完成配置。

配置完成之后的服务器如下图所示：



名称	可用区	IP	计费方式	规格	状态	创建...	操作
hadoop1	可用区 A	私有网 10.173.32.9 公网 59.111.99.128	按量付费	n1, 1核 2GB	运行中	今天 18:49	创建快照 删除 更多
hadoop4	可用区 A	私有网 10.173.32.8	按量付费	n1, 1核 2GB	运行中	今天 18:49	创建快照 删除 更多
hadoop3	可用区 A	私有网 10.173.32.4	按量付费	n1, 1核 2GB	运行中	今天 18:49	创建快照 删除 更多
hadoop2	可用区 A	私有网 10.173.32.5	按量付费	n1, 1核 2GB	运行中	今天 18:49	创建快照 删除 更多

2、网易云hadoop集群搭建

2.1、安装Java

对四台主机分别进行Java环境的配置，步骤如下：在终端中输入

```
sudo apt-get install default-jre default-jdk
```

安装Java环境，该过程中需要保持联网状态。安装结束后，需要配置JAVA_HOME环境变量，在Linux终端中打开当前登陆用户的环境变量配置文件.bashrc，在文件最前面添加如下单独一行，然后保存退出：

```
export JAVA_HOME=/usr/lib/jvm/default-java
```

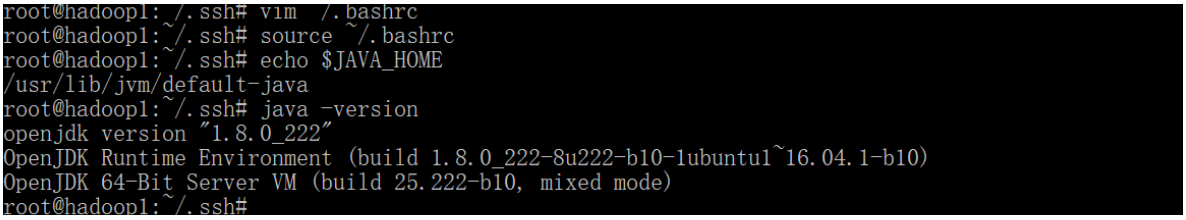
接下来，执行如下命令使得环境变量立即生效：

```
source ~/.bashrc       # 使环境变量生效
```

之后，通过执行如下命令检验设置是否正确：

```
echo $JAVA_HOME        # 检验变量值
```

检验的结果如下图所示：



```
root@hadoop1:~/.ssh# vim ~/.bashrc
root@hadoop1:~/.ssh# source ~/.bashrc
root@hadoop1:~/.ssh# echo $JAVA_HOME
/usr/lib/jvm/default-java
root@hadoop1:~/.ssh# java -version
openjdk version "1.8.0_222"
OpenJDK Runtime Environment (build 1.8.0_222-8u222-b10-1ubuntu1~16.04.1-b10)
OpenJDK 64-Bit Server VM (build 25.222-b10, mixed mode)
root@hadoop1:~/.ssh#
```

2.2、安装hadoop

首先将hadoop下载到本机，再通过如下命令将hadoop文件传到hadoop1主机：

```
scp -r Downloads/hadoop-3.2.0.tar.gz root@59.111.99.128:~
```

本机进入hadoop1主机，将hadoop安装到/usr/local/中：

```
sudo tar -zxvf hadoop-3.2.0.tar.gz -C /usr/local     # 解压到/usr/local中
cd /usr/local/
sudo mv hadoop-3.2.0/ hadoop                         # 将文件夹名改为hadoop
```

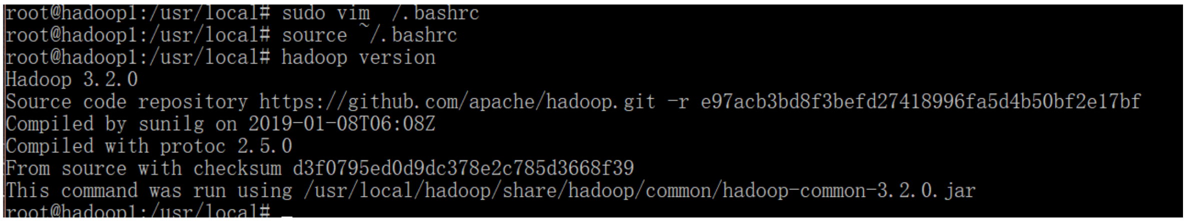
之后进行hadoop环境变量配置，将hadoop的安装路径加入到PATH变量中，这样就可以在任意目录中使用hadoop、hdfs等指令了。在终端中打开编辑./bashrc文件，加入一行：

```
export PATH=$PATH:/usr/local/hadoop/bin:/usr/local/hadoop/sbin
```

保存退出后，执行source命令使得该环境变量立即生效。配置完成后，在终端输入如下命令查看hadoop版本信息：

```
hadoop version
```

查看hadoop版本信息如下：



```
root@hadoop1:/usr/local# sudo vim  /.bashrc
root@hadoop1:/usr/local# source ~/.bashrc
root@hadoop1:/usr/local# hadoop version
Hadoop 3.2.0
Source code repository https://github.com/apache/hadoop.git -r e97acb3bd8f3befd27418996fa5d4b50bf2e17bf
Compiled by sunilg on 2019-01-08T06:08Z
Compiled with protoc 2.5.0
From source with checksum d3f0795ed0d9dc378e2c785d3668f39
This command was run using /usr/local/hadoop/share/hadoop/common/hadoop-common-3.2.0.jar
root@hadoop1:/usr/local# _
```

之后进行集群/分布式环境的配置，需要修改/usr/local/hadoop/etc/hadoop中的5个配置文件，这里设置了正常启动必须的设置项：slaves、core-site.xml、hdfs-site.xml、mapred-site.xml、yarn-site.xml。

文件slaves，将作为DataNode的主机名写入该文件，每行一个，默认为localhost，所以在伪分布 式配置时，节点即作为 NameNode 也作为DataNode。分布式配置可以保留 localhost，也可以删掉， 让Master节点仅作为NameNode使用。这里将hadoop1节点作为NameNode使用，让hadoop2，3，4作为DataNode使用，因此将文件中原来的localhost删除，只添加其他三台主机的名称。

之后将剩下的四个文件都更改为教程中的环境配置。

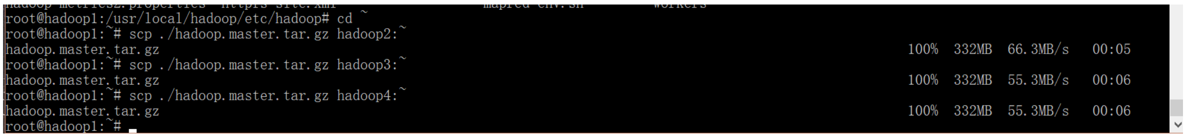
配置好后，将hadoop1上的/usr/local/hadoop文件复制到各个节点hadoop2，3，4上，首先对该文件进行压缩：

```
sudo tar -zcvf ~/hadoop.master.tar.gz /usr/local/haoop
```

随后进行复制操作：

```
scp ./hadoop.master.tar.gz hadoop2:~
scp ./hadoop.maste
r.tar.gz hadoop3:~
scp ./hadoop.master.tar.gz hadoop4:~
```

复制过程截图如下：



```
root@hadoop1:/usr/local/hadoop/etc/hadoop# cd ~
root@hadoop1:~# scp ./hadoop.master.tar.gz hadoop2:~
hadoop.master.tar.gz                                                    100%  332MB  66.3MB/s   00:05
root@hadoop1:~# scp ./hadoop.master.tar.gz hadoop3:~
hadoop.master.tar.gz                                                    100%  332MB  55.3MB/s   00:06
root@hadoop1:~# scp ./hadoop.master.tar.gz hadoop4:~
hadoop.master.tar.gz                                                    100%  332MB  55.3MB/s   00:06
root@hadoop1:~#
```

在hadoop2，3，4节点上执行：

```
sudo tar-zvxf ~/hadoop.master.tar.gz -C /usr/local
```

2.3、hadoop1主机启动hadoop

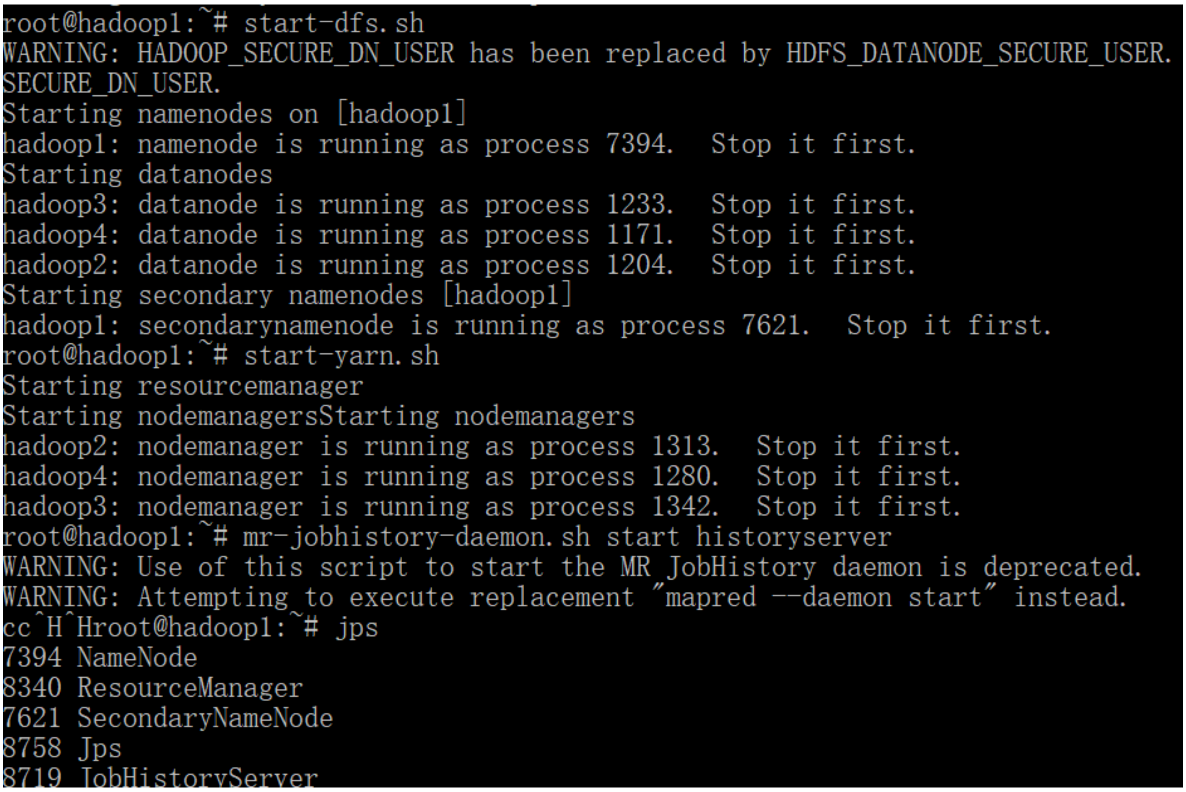
首次启动需要在hadoop1节点执行NameNode的格式化：

```
hdfs  namenode -format                   #首次运行需要初始化，之后不需要
```

接着在hadoop1节点上运行如下三个指令：

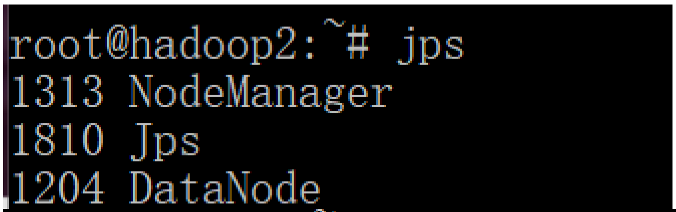
```
start-dfs.sh
start-yarn.sh
mr-jobhistory-daemon.sh start historyserver
```

之后通过命令jps查看各个节点启动的进程，上述指令运行结果如下图所示：

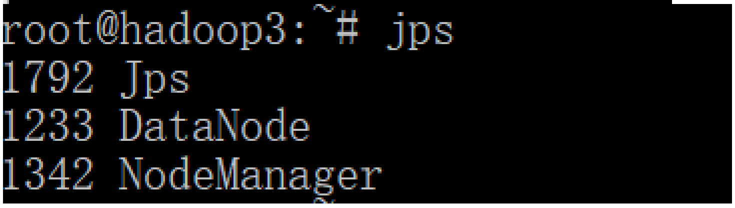


```
root@hadoop1:~# start-dfs.sh
WARNING: HADOOP_SECURE_DN_USER has been replaced by HDFS_DATANODE_SECURE_USER.
SECURE_DN_USER.
Starting namenodes on [hadoop1]
hadoop1: namenode is running as process 7394.  Stop it first.
Starting datanodes
hadoop3: datanode is running as process 1233.  Stop it first.
hadoop4: datanode is running as process 1171.  Stop it first.
hadoop2: datanode is running as process 1204.  Stop it first.
Starting secondary namenodes [hadoop1]
hadoop1: secondarynamenode is running as process 7621.  Stop it first.
root@hadoop1:~# start-yarn.sh
Starting resourcemanager
Starting nodemanagersStarting nodemanagers
hadoop2: nodemanager is running as process 1313.  Stop it first.
hadoop4: nodemanager is running as process 1280.  Stop it first.
hadoop3: nodemanager is running as process 1342.  Stop it first.
root@hadoop1:~# mr-jobhistory-daemon.sh start historyserver
WARNING: Use of this script to start the MR JobHistory daemon is deprecated.
WARNING: Attempting to execute replacement "mapred --daemon start" instead.
cc^H^Hroot@hadoop1:~# jps
7394 NameNode
8340 ResourceManager
7621 SecondaryNameNode
8758 Jps
8719 JobHistoryServer
```

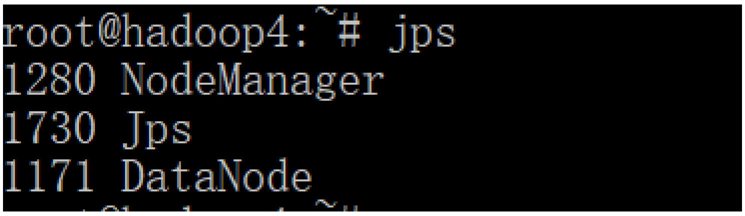
在hadoop2，3，4节点可以看到DataNode和NodeManager进程，如下图所示：



```
root@hadoop2:~# jps
1313 NodeManager
1810 Jps
1204 DataNode
```



```
root@hadoop3:~# jps
1792 Jps
1233 DataNode
1342 NodeManager
```

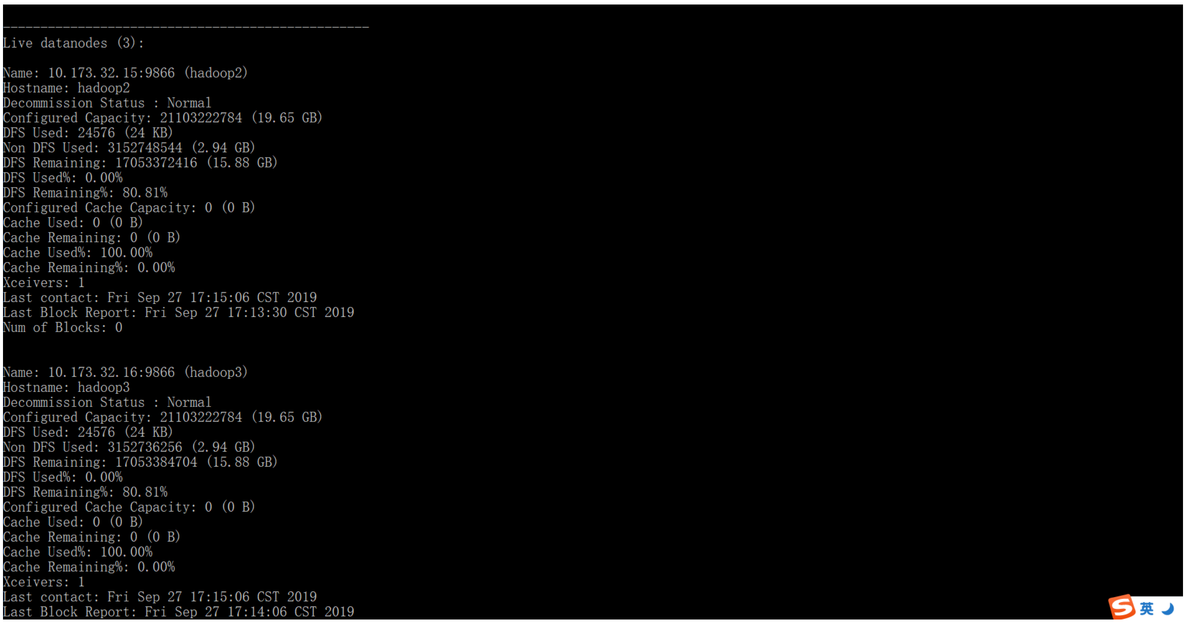


```
root@hadoop4:~# jps
1280 NodeManager
1730 Jps
1171 DataNode
```

另外需要在hadoop1节点上通过命令

```
hdfs dfsadmin -report
```

查看DataNode是否正常启动，如果Live datanodes不为0，则说明集群启动成功，命令运行结果如下图所示：



```
-------------------------------------------------
Live datanodes (3):

Name: 10.173.32.15:9866 (hadoop2)
Hostname: hadoop2
Decommission Status : Normal
Configured Capacity: 21103222784 (19.65 GB)
DFS Used: 24576 (24 KB)
Non DFS Used: 3152748544 (2.94 GB)
DFS Remaining: 17053372416 (15.88 GB)
DFS Used%: 0.00%
DFS Remaining%: 80.81%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Fri Sep 27 17:15:06 CST 2019
Last Block Report: Fri Sep 27 17:13:30 CST 2019
Num of Blocks: 0


Name: 10.173.32.16:9866 (hadoop3)
Hostname: hadoop3
Decommission Status : Normal
Configured Capacity: 21103222784 (19.65 GB)
DFS Used: 24576 (24 KB)
Non DFS Used: 3152736256 (2.94 GB)
DFS Remaining: 17053384704 (15.88 GB)
DFS Used%: 0.00%
DFS Remaining%: 80.81%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Fri Sep 27 17:15:06 CST 2019
Last Block Report: Fri Sep 27 17:14:06 CST 2019
```

2.4、执行Hadoop自带样例

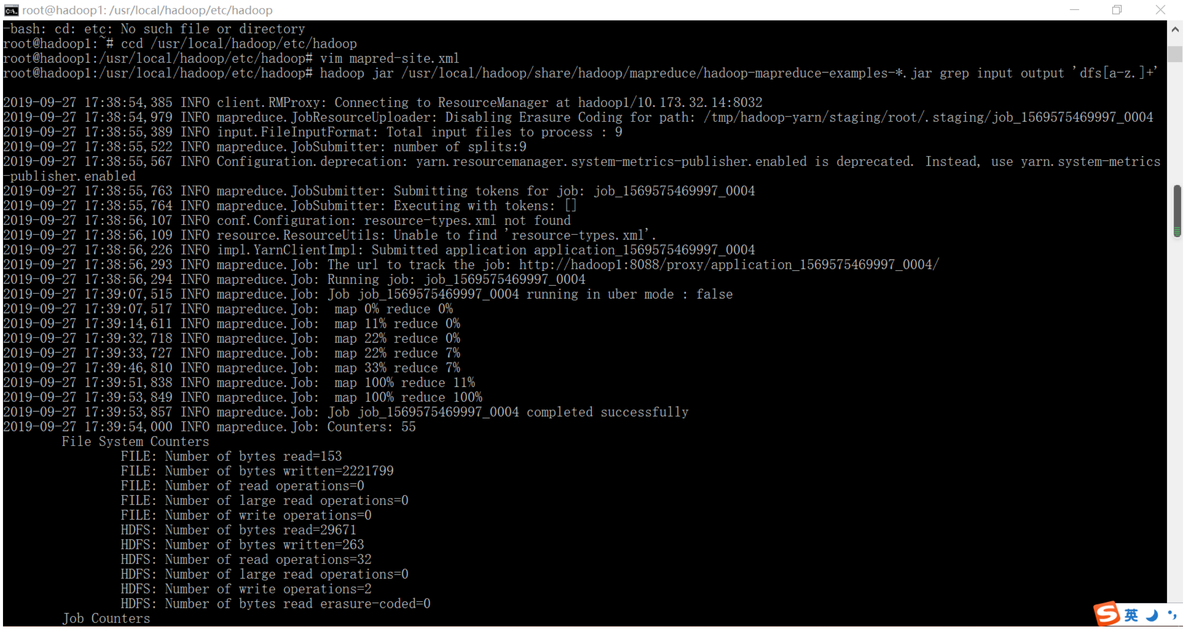
首先创建HDFS上的用户目录：

```
hdfs dfs -mkdir -p /user/root/input
```

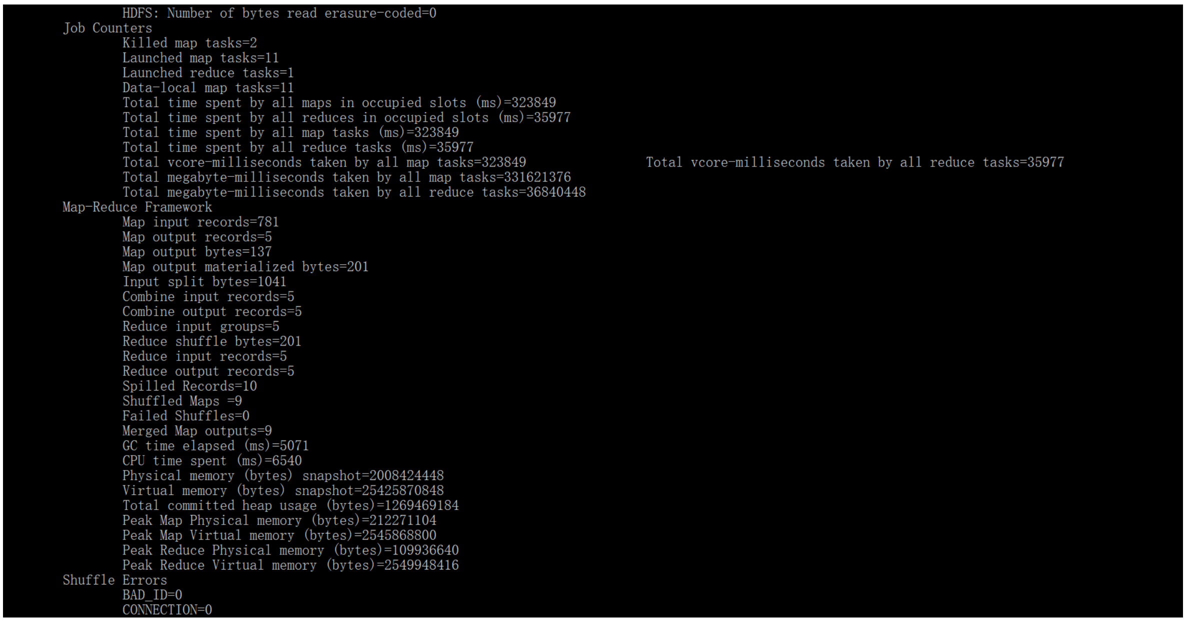
将/usr/local/hadoop/etc/hadoop中的配置文件作为输入文件复制到分布式文件系统中：

```
hdfs dfs -put /usr/local/hadoop/etc/hadoop/*.xml /user/root/input
```

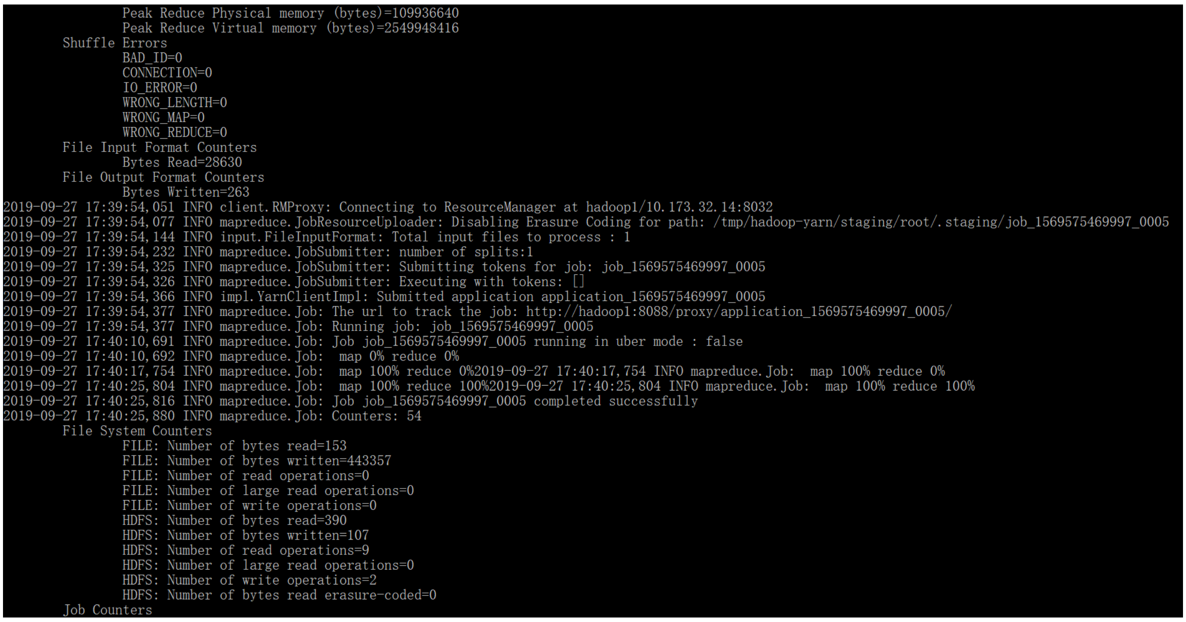
运行测试程序后的结果如下图所示：



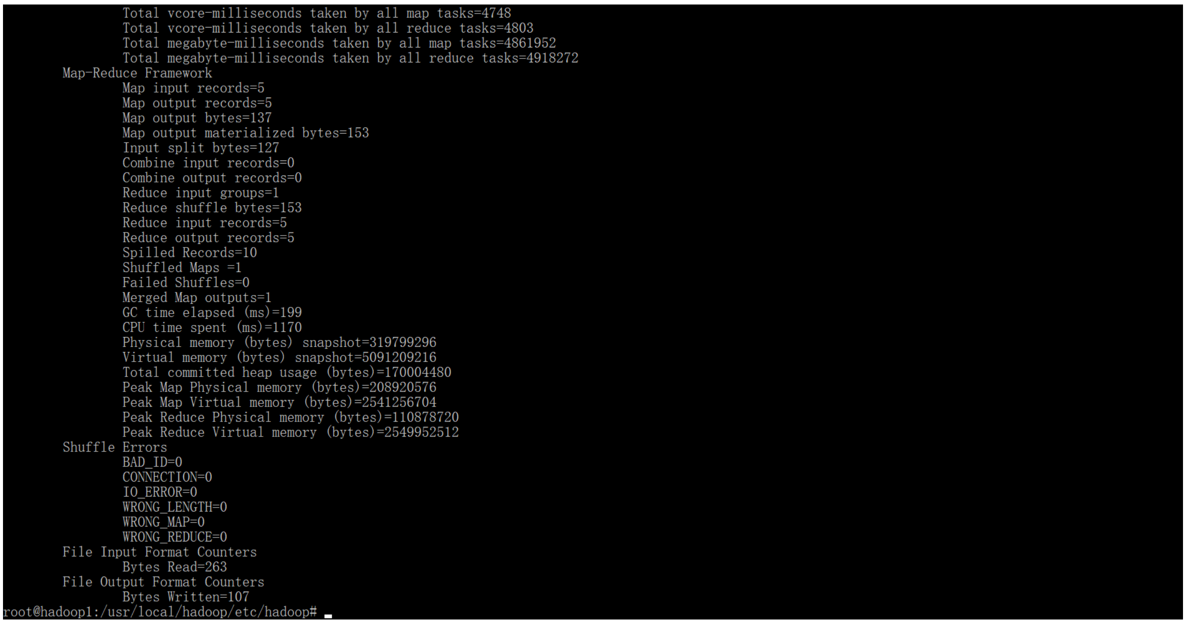
```
-bash: cd: etc: No such file or directory
root@hadoop1:~# ccd /usr/local/hadoop/etc/hadoop
root@hadoop1:/usr/local/hadoop/etc/hadoop# vim mapred-site.xml
root@hadoop1:/usr/local/hadoop/etc/hadoop# hadoop jar /usr/local/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-*.jar grep input output 'dfs[a-z.]+'

2019-09-27 17:38:54,385 INFO client.RMProxy: Connecting to ResourceManager at hadoop1/10.173.32.14:8032
2019-09-27 17:38:54,979 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1569575469997_0004
2019-09-27 17:38:55,389 INFO input.FileInputFormat: Total input files to process : 9
2019-09-27 17:38:55,522 INFO mapreduce.JobSubmitter: number of splits:9
2019-09-27 17:38:55,567 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics
-publisher.enabled
2019-09-27 17:38:55,763 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1569575469997_0004
2019-09-27 17:38:55,764 INFO mapreduce.JobSubmitter: Executing with tokens: []
2019-09-27 17:38:56,107 INFO conf.Configuration: resource-types.xml not found
2019-09-27 17:38:56,109 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2019-09-27 17:38:56,226 INFO impl.YarnClientImpl: Submitted application application_1569575469997_0004
2019-09-27 17:38:56,293 INFO mapreduce.Job: The url to track the job: http://hadoop1:8088/proxy/application_1569575469997_0004/
2019-09-27 17:38:56,294 INFO mapreduce.Job: Running job: job_1569575469997_0004
2019-09-27 17:39:07,515 INFO mapreduce.Job: Job job_1569575469997_0004 running in uber mode : false
2019-09-27 17:39:07,517 INFO mapreduce.Job:  map 0% reduce 0%
2019-09-27 17:39:14,611 INFO mapreduce.Job:  map 11% reduce 0%
2019-09-27 17:39:32,718 INFO mapreduce.Job:  map 22% reduce 0%
2019-09-27 17:39:33,727 INFO mapreduce.Job:  map 22% reduce 7%
2019-09-27 17:39:46,810 INFO mapreduce.Job:  map 33% reduce 7%
2019-09-27 17:39:51,838 INFO mapreduce.Job:  map 100% reduce 11%
2019-09-27 17:39:53,849 INFO mapreduce.Job:  map 100% reduce 100%
2019-09-27 17:39:53,857 INFO mapreduce.Job: Job job_1569575469997_0004 completed successfully
2019-09-27 17:39:54,000 INFO mapreduce.Job: Counters: 55
        File System Counters
                FILE: Number of bytes read=153
                FILE: Number of bytes written=2221799
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=29671
                HDFS: Number of bytes written=263
                HDFS: Number of read operations=32
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
```



```
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Killed map tasks=2
                Launched map tasks=11
                Launched reduce tasks=1
                Data-local map tasks=11
                Total time spent by all maps in occupied slots (ms)=323849
                Total time spent by all reduces in occupied slots (ms)=35977
                Total time spent by all map tasks (ms)=323849
                Total time spent by all reduce tasks (ms)=35977
                Total vcore-milliseconds taken by all map tasks=323849                Total vcore-milliseconds taken by all reduce tasks=35977
                Total megabyte-milliseconds taken by all map tasks=331621376
                Total megabyte-milliseconds taken by all reduce tasks=36840448
        Map-Reduce Framework
                Map input records=781
                Map output records=5
                Map output bytes=137
                Map output materialized bytes=201
                Input split bytes=1041
                Combine input records=5
                Combine output records=5
                Reduce input groups=5
                Reduce shuffle bytes=201
                Reduce input records=5
                Reduce output records=5
                Spilled Records=10
                Shuffled Maps =9
                Failed Shuffles=0
                Merged Map outputs=9
                GC time elapsed (ms)=5071
                CPU time spent (ms)=6540
                Physical memory (bytes) snapshot=2008424448
                Virtual memory (bytes) snapshot=25425870848
                Total committed heap usage (bytes)=1269469184
                Peak Map Physical memory (bytes)=212271104
                Peak Map Virtual memory (bytes)=2545868800
                Peak Reduce Physical memory (bytes)=109936640
                Peak Reduce Virtual memory (bytes)=2549948416
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
```



```
                Peak Reduce Physical memory (bytes)=109936640
                Peak Reduce Virtual memory (bytes)=2549948416
        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=28630
        File Output Format Counters
                Bytes Written=263
2019-09-27 17:39:54,051 INFO client.RMProxy: Connecting to ResourceManager at hadoop1/10.173.32.14:8032
2019-09-27 17:39:54,077 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1569575469997_0005
2019-09-27 17:39:54,144 INFO input.FileInputFormat: Total input files to process : 1
2019-09-27 17:39:54,232 INFO mapreduce.JobSubmitter: number of splits:1
2019-09-27 17:39:54,325 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1569575469997_0005
2019-09-27 17:39:54,326 INFO mapreduce.JobSubmitter: Executing with tokens: []
2019-09-27 17:39:54,366 INFO impl.YarnClientImpl: Submitted application application_1569575469997_0005
2019-09-27 17:39:54,377 INFO mapreduce.Job: The url to track the job: http://hadoop1:8088/proxy/application_1569575469997_0005/
2019-09-27 17:39:54,377 INFO mapreduce.Job: Running job: job_1569575469997_0005
2019-09-27 17:40:10,691 INFO mapreduce.Job: Job job_1569575469997_0005 running in uber mode : false
2019-09-27 17:40:10,692 INFO mapreduce.Job:  map 0% reduce 0%
2019-09-27 17:40:17,754 INFO mapreduce.Job:  map 100% reduce 0%2019-09-27 17:40:17,754 INFO mapreduce.Job:  map 100% reduce 0%
2019-09-27 17:40:25,804 INFO mapreduce.Job:  map 100% reduce 100%2019-09-27 17:40:25,804 INFO mapreduce.Job:  map 100% reduce 100%
2019-09-27 17:40:25,816 INFO mapreduce.Job: Job job_1569575469997_0005 completed successfully
2019-09-27 17:40:25,880 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=153
                FILE: Number of bytes written=443357
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=390
                HDFS: Number of bytes written=107
                HDFS: Number of read operations=9
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
```



```
                Total vcore-milliseconds taken by all map tasks=4748
                Total vcore-milliseconds taken by all reduce tasks=4803
                Total megabyte-milliseconds taken by all map tasks=4861952
                Total megabyte-milliseconds taken by all reduce tasks=4918272
        Map-Reduce Framework
                Map input records=5
                Map output records=5
                Map output bytes=137
                Map output materialized bytes=153
                Input split bytes=127
                Combine input records=0
                Combine output records=0
                Reduce input groups=1
                Reduce shuffle bytes=153
                Reduce input records=5
                Reduce output records=5
                Spilled Records=10
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=199
                CPU time spent (ms)=1170
                Physical memory (bytes) snapshot=319799296
                Virtual memory (bytes) snapshot=5091209216
                Total committed heap usage (bytes)=170004480
                Peak Map Physical memory (bytes)=208920576
                Peak Map Virtual memory (bytes)=2541256704
                Peak Reduce Physical memory (bytes)=110878720
                Peak Reduce Virtual memory (bytes)=2549952512
        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=263
        File Output Format Counters
                Bytes Written=107
root@hadoop1:/usr/local/hadoop/etc/hadoop# _
```

2.5、执行WordCount样例（file0、1、2）

首先配置hadoop包环境配置，将hadoop的classpath信息添加到CLASSPATH变量中，在./bahrc文件中添加如下几行：

```
export HADOOP_HOME=/usr/local/hadoop export CLASSPATH=$($HADOOP_HOME/bin/hadoop classpath):$CLASSPATH
```

执行source指令使变量生效。

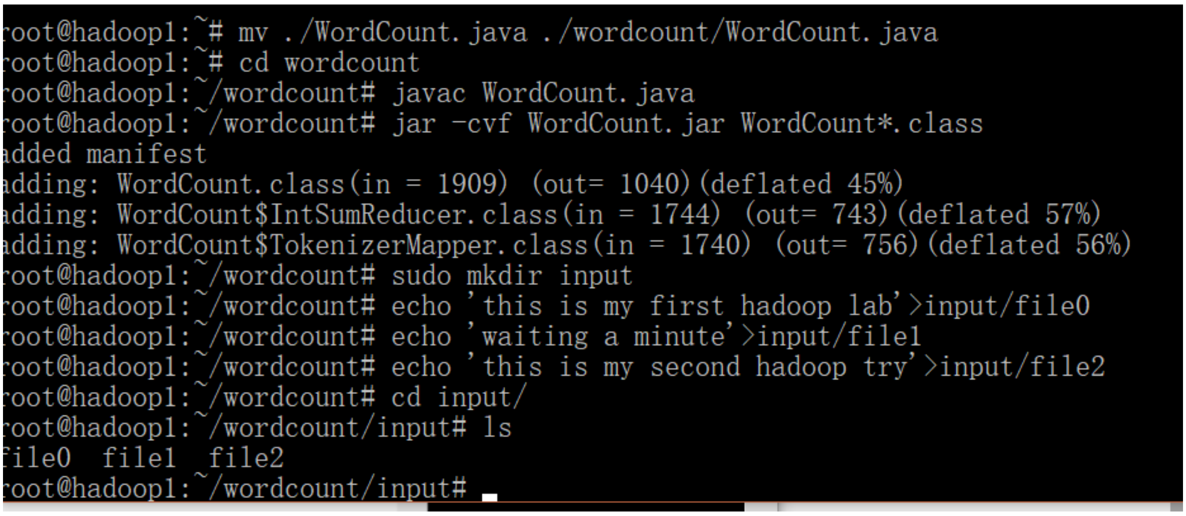
之后编写wordcount程序，在hadoop1当前目录~下新建目录wordcount，将代码复制进WordCount.java，编译，打包jar：

```
Sudo mkdir wordcount
cd wordcount/ sudo vi WordCount.java
javac WordCount.java jar -cvf WordCount.jar WordCount*.class
```

打包完成后，创建输入文件，并测试运行：

```
sudo mkdir input echo 'this is my first hadoop lab' > input/file0 echo 'waiting a minute' > input/file1 echo 'this is my second hadoop try' > input/file2
```

运行结果如下图所示：



```
oot@hadoop1:~# mv ./WordCount.java ./wordcount/WordCount.java
oot@hadoop1:~# cd wordcount
oot@hadoop1:~/wordcount# javac WordCount.java
oot@hadoop1:~/wordcount# jar -cvf WordCount.jar WordCount*.class
dded manifest
dding: WordCount.class(in = 1909) (out= 1040)(deflated 45%)
dding: WordCount$IntSumReducer.class(in = 1744) (out= 743)(deflated 57%)
dding: WordCount$TokenizerMapper.class(in = 1740) (out= 756)(deflated 56%)
oot@hadoop1:~/wordcount# sudo mkdir input
oot@hadoop1:~/wordcount# echo 'this is my first hadoop lab'>input/file0
oot@hadoop1:~/wordcount# echo 'waiting a minute'>input/file1
oot@hadoop1:~/wordcount# echo 'this is my second hadoop try'>input/file2
oot@hadoop1:~/wordcount# cd input/
oot@hadoop1:~/wordcount/input# ls
ile0  file1  file2
oot@hadoop1:~/wordcount/input# _
```

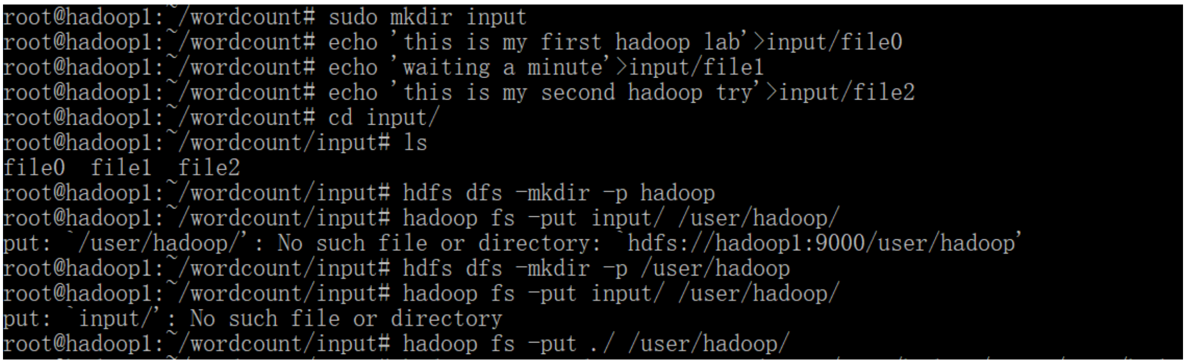
重新创建hdfs用户目录：

```
hdfs dfs -mkdir -p /user/hadoop
```

把本地文件上传到分布式HDFS上：

```
hadoop fs -put input/ /user/hadoop/
```

执行结果如下图所示：

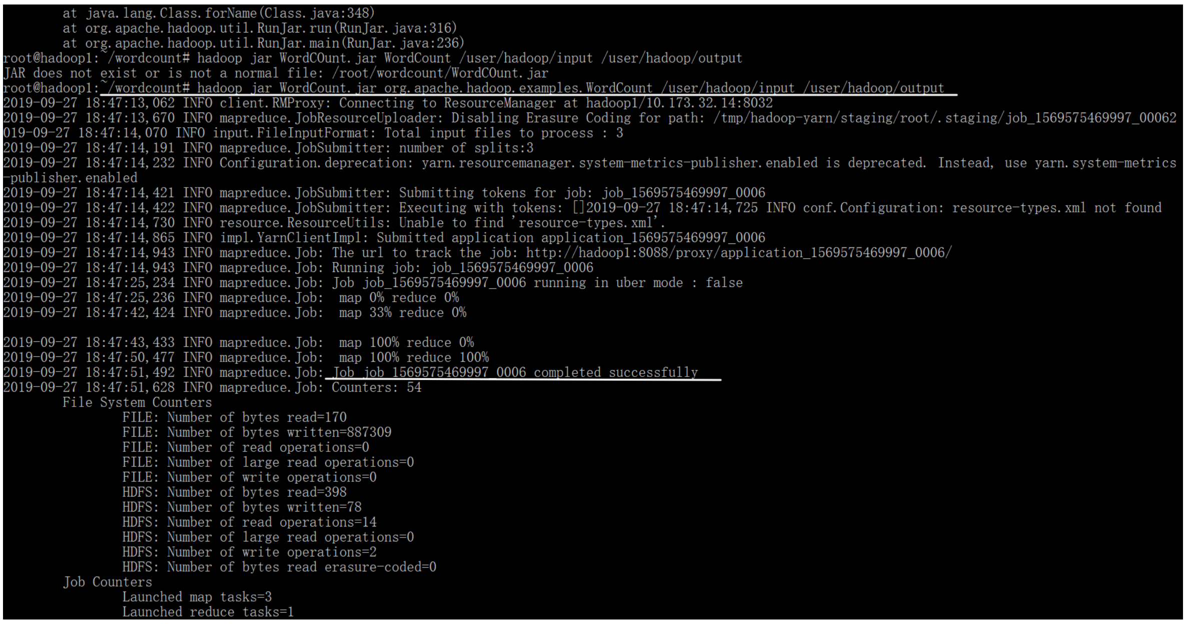


```
root@hadoop1:~/wordcount# sudo mkdir input
root@hadoop1:~/wordcount# echo 'this is my first hadoop lab'>input/file0
root@hadoop1:~/wordcount# echo 'waiting a minute'>input/file1
root@hadoop1:~/wordcount# echo 'this is my second hadoop try'>input/file2
root@hadoop1:~/wordcount# cd input/
root@hadoop1:~/wordcount/input# ls
file0  file1  file2
root@hadoop1:~/wordcount/input# hdfs dfs -mkdir -p hadoop
root@hadoop1:~/wordcount/input# hadoop fs -put input/ /user/hadoop/
put: `/user/hadoop/': No such file or directory: `hdfs://hadoop1:9000/user/hadoop'
root@hadoop1:~/wordcount/input# hdfs dfs -mkdir -p /user/hadoop
root@hadoop1:~/wordcount/input# hadoop fs -put input/ /user/hadoop/
put: `input/': No such file or directory
root@hadoop1:~/wordcount/input# hadoop fs -put ./ /user/hadoop/
```

开始运行：

```
hadoop jar WordCount.jar WordCount /user/hadoop/input /user/hadoop/output
```

运行结果如下图所示：

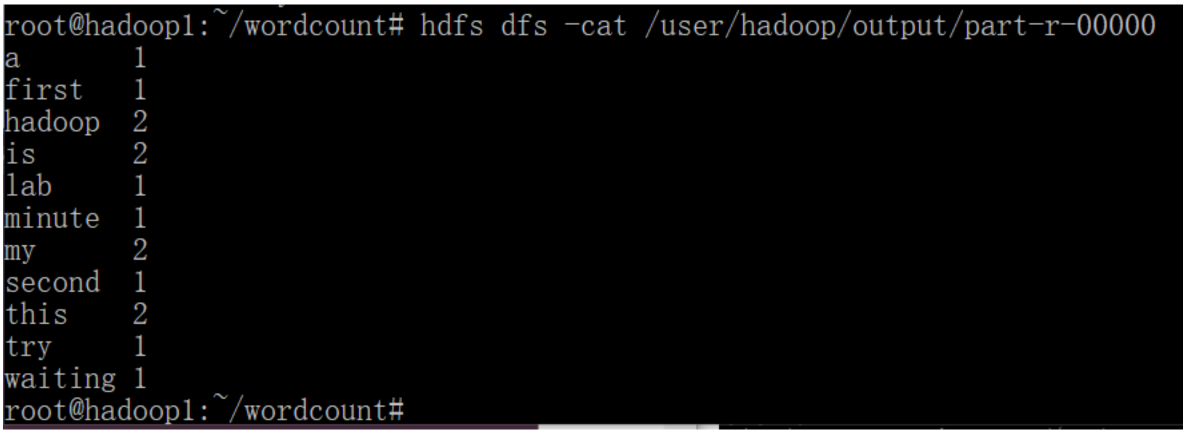


```
        at java.lang.Class.forName(Class.java:348)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:316)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
root@hadoop1:~/wordcount# hadoop jar WordCOunt.jar WordCount /user/hadoop/input /user/hadoop/output
JAR does not exist or is not a normal file: /root/wordcount/WordCOunt.jar
root@hadoop1:~/wordcount# hadoop jar WordCount.jar org.apache.hadoop.examples.WordCount /user/hadoop/input /user/hadoop/output
2019-09-27 18:47:13,062 INFO client.RMProxy: Connecting to ResourceManager at hadoop1/10.173.32.14:8032
2019-09-27 18:47:13,670 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/root/.staging/job_1569575469997_00062
019-09-27 18:47:14,070 INFO input.FileInputFormat: Total input files to process : 3
2019-09-27 18:47:14,191 INFO mapreduce.JobSubmitter: number of splits:3
2019-09-27 18:47:14,232 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics
-publisher.enabled
2019-09-27 18:47:14,421 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1569575469997_0006
2019-09-27 18:47:14,422 INFO mapreduce.JobSubmitter: Executing with tokens: []2019-09-27 18:47:14,725 INFO conf.Configuration: resource-types.xml not found
2019-09-27 18:47:14,730 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2019-09-27 18:47:14,865 INFO impl.YarnClientImpl: Submitted application application_1569575469997_0006
2019-09-27 18:47:14,943 INFO mapreduce.Job: The url to track the job: http://hadoop1:8088/proxy/application_1569575469997_0006/
2019-09-27 18:47:14,943 INFO mapreduce.Job: Running job: job_1569575469997_0006
2019-09-27 18:47:25,234 INFO mapreduce.Job: Job job_1569575469997_0006 running in uber mode : false
2019-09-27 18:47:25,236 INFO mapreduce.Job:  map 0% reduce 0%
2019-09-27 18:47:42,424 INFO mapreduce.Job:  map 33% reduce 0%

2019-09-27 18:47:43,433 INFO mapreduce.Job:  map 100% reduce 0%
2019-09-27 18:47:50,477 INFO mapreduce.Job:  map 100% reduce 100%
2019-09-27 18:47:51,492 INFO mapreduce.Job: Job job_1569575469997_0006 completed successfully
2019-09-27 18:47:51,628 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=170
                FILE: Number of bytes written=887309
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=398
                HDFS: Number of bytes written=78
                HDFS: Number of read operations=14
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=3
                Launched reduce tasks=1
```

用

```
hdfs dfs -cat /user/hadoop/output/part-r-00000
```

查看结果，结果如下图所示，实验成功。



```
root@hadoop1:~/wordcount# hdfs dfs -cat /user/hadoop/output/part-r-00000
a       1
first   1
hadoop  2
is      2
lab     1
minute  1
my      2
second  1
this    2
try     1
waiting 1
root@hadoop1:~/wordcount#
```

3、WordCount处理垃圾邮件数据

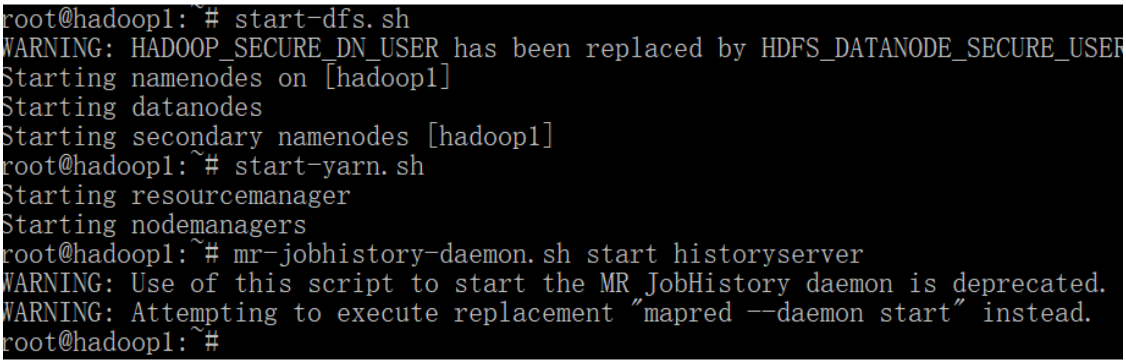
实验要求通过WordCount处理垃圾邮件数据，所以根据上述步骤在邮件数据上进行了同样的操作，在此一一列举：

1. 通过`scp`命令将dataset传输到hadoop1中。

```
C:\Users\x1c>scp -r ./Downloads/dataset.zip root@59.111.99.235:~
C:\Users\x1c>ssh root@59.111.99.235
```

2. 启动hadoop，运行jps观察是否启动成功。

```
root@haoop1:~# start-dfs.sh
root@haoop1:~# start-yarn.sh
root@haoop1:~# mr-jobhistory-daemn.sh start historyserver
```

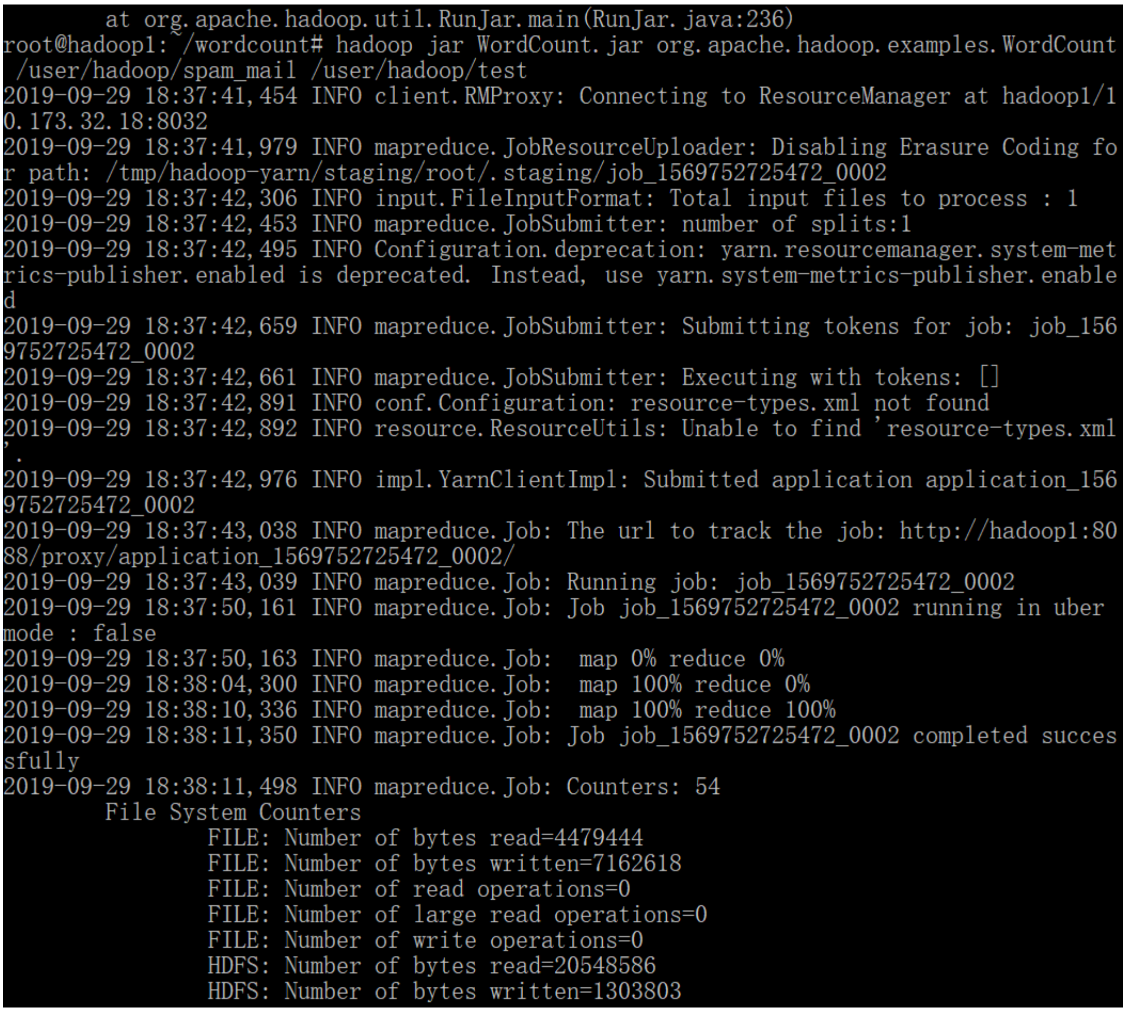


```
root@hadoop1:~# start-dfs.sh
WARNING: HADOOP_SECURE_DN_USER has been replaced by HDFS_DATANODE_SECURE_USER
Starting namenodes on [hadoop1]
Starting datanodes
Starting secondary namenodes [hadoop1]
root@hadoop1:~# start-yarn.sh
Starting resourcemanager
Starting nodemanagers
root@hadoop1:~# mr-jobhistory-daemon.sh start historyserver
WARNING: Use of this script to start the MR JobHistory daemon is deprecated.
WARNING: Attempting to execute replacement "mapred --daemon start" instead.
root@hadoop1:~#
```

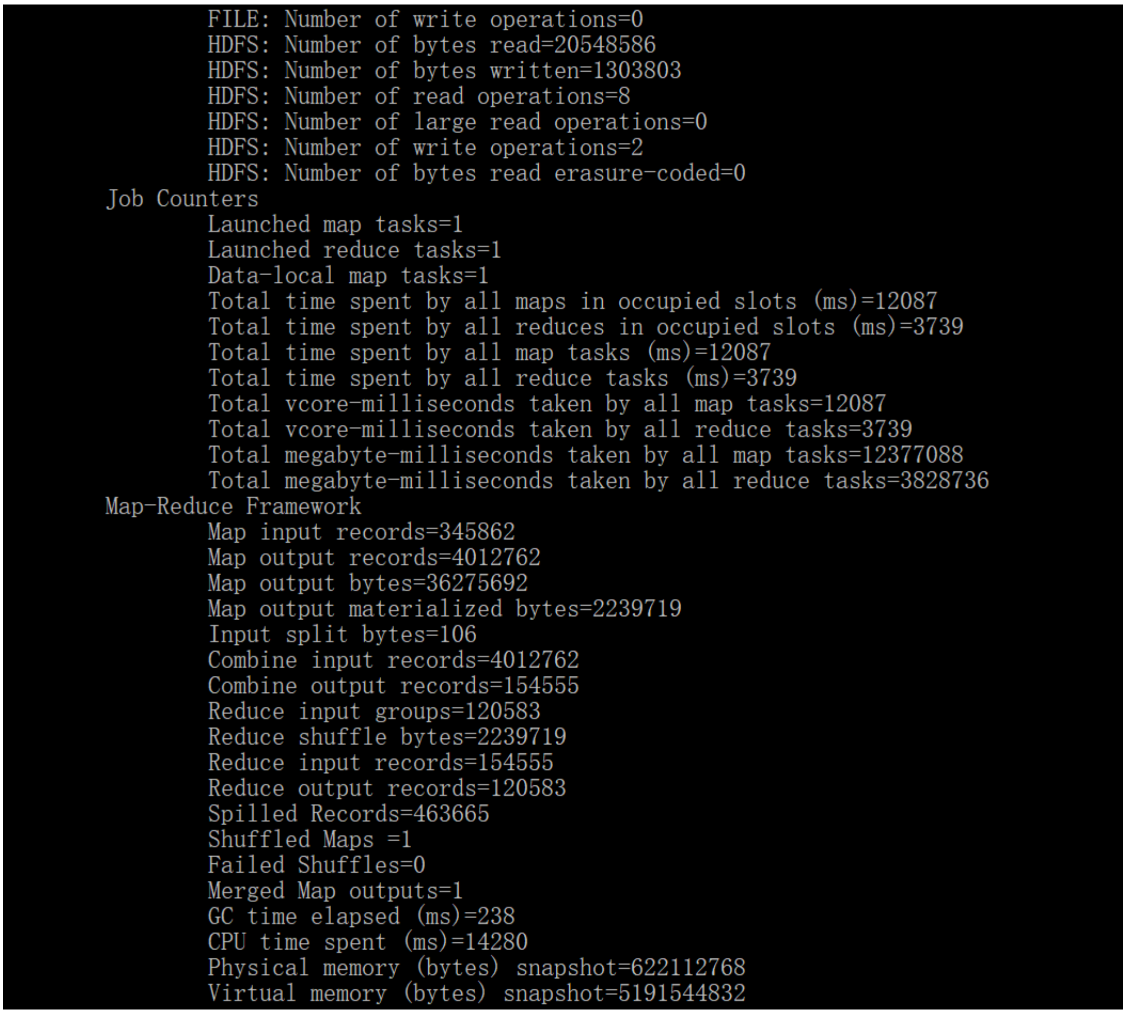
3. 将spam数据传送到hadoop分布式文件系统`hdfs`的文件夹`/user/hadoop/`中，随后运行`WordCount.jar`包启动WordCount程序。

```
root@hadoop1:~/wordcount# hdfs dfs -mkdir -p /user/hadoop
root@hadoop1:~/wordcount# hadoop fs -put ./spam/ /user/hadoop/
root@hadoop1:~/wordcount# hadoop jar WordCount.jar
org.apache.hadoop.examples.WordCount /user/hadoop/spam /user/hadoop/output1
```

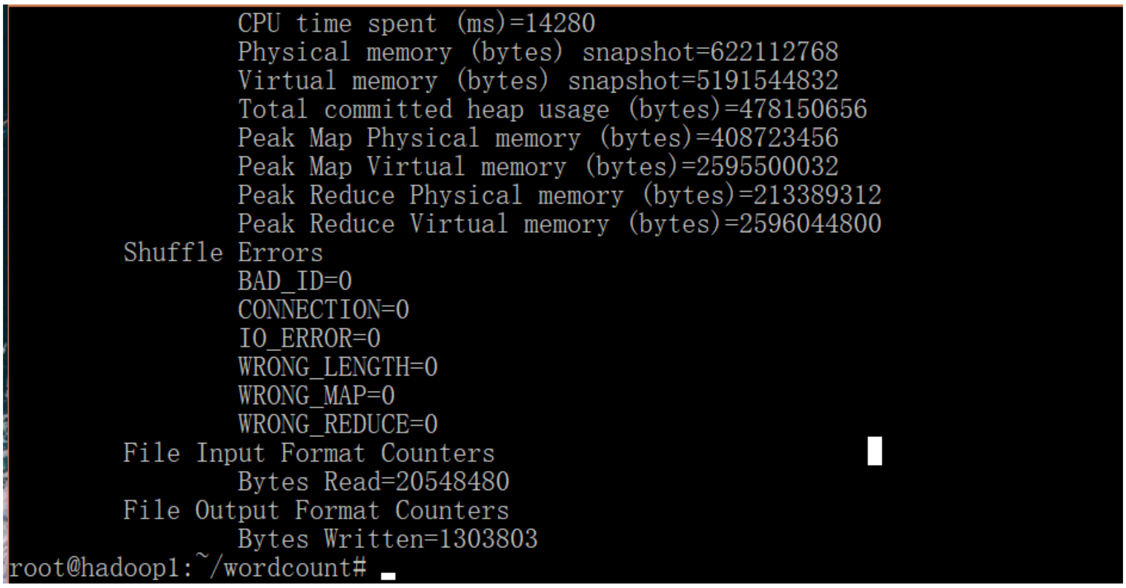
4. 运行结果：



```
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
root@hadoop1:~/wordcount# hadoop jar WordCount.jar org.apache.hadoop.examples.WordCount
 /user/hadoop/spam_mail /user/hadoop/test
2019-09-29 18:37:41,454 INFO client.RMProxy: Connecting to ResourceManager at hadoop1/1
0.173.32.18:8032
2019-09-29 18:37:41,979 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding fo
r path: /tmp/hadoop-yarn/staging/root/.staging/job_1569752725472_0002
2019-09-29 18:37:42,306 INFO input.FileInputFormat: Total input files to process : 1
2019-09-29 18:37:42,453 INFO mapreduce.JobSubmitter: number of splits:1
2019-09-29 18:37:42,495 INFO Configuration.deprecation: yarn.resourcemanager.system-met
rics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enable
d
2019-09-29 18:37:42,659 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_156
9752725472_0002
2019-09-29 18:37:42,661 INFO mapreduce.JobSubmitter: Executing with tokens: []
2019-09-29 18:37:42,891 INFO conf.Configuration: resource-types.xml not found
2019-09-29 18:37:42,892 INFO resource.ResourceUtils: Unable to find 'resource-types.xml
'.
2019-09-29 18:37:42,976 INFO impl.YarnClientImpl: Submitted application application_156
9752725472_0002
2019-09-29 18:37:43,038 INFO mapreduce.Job: The url to track the job: http://hadoop1:80
88/proxy/application_1569752725472_0002/
2019-09-29 18:37:43,039 INFO mapreduce.Job: Running job: job_1569752725472_0002
2019-09-29 18:37:50,161 INFO mapreduce.Job: Job job_1569752725472_0002 running in uber
mode : false
2019-09-29 18:37:50,163 INFO mapreduce.Job:  map 0% reduce 0%
2019-09-29 18:38:04,300 INFO mapreduce.Job:  map 100% reduce 0%
2019-09-29 18:38:10,336 INFO mapreduce.Job:  map 100% reduce 100%
2019-09-29 18:38:11,350 INFO mapreduce.Job: Job job_1569752725472_0002 completed succes
sfully
2019-09-29 18:38:11,498 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=4479444
                FILE: Number of bytes written=7162618
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=20548586
                HDFS: Number of bytes written=1303803
```



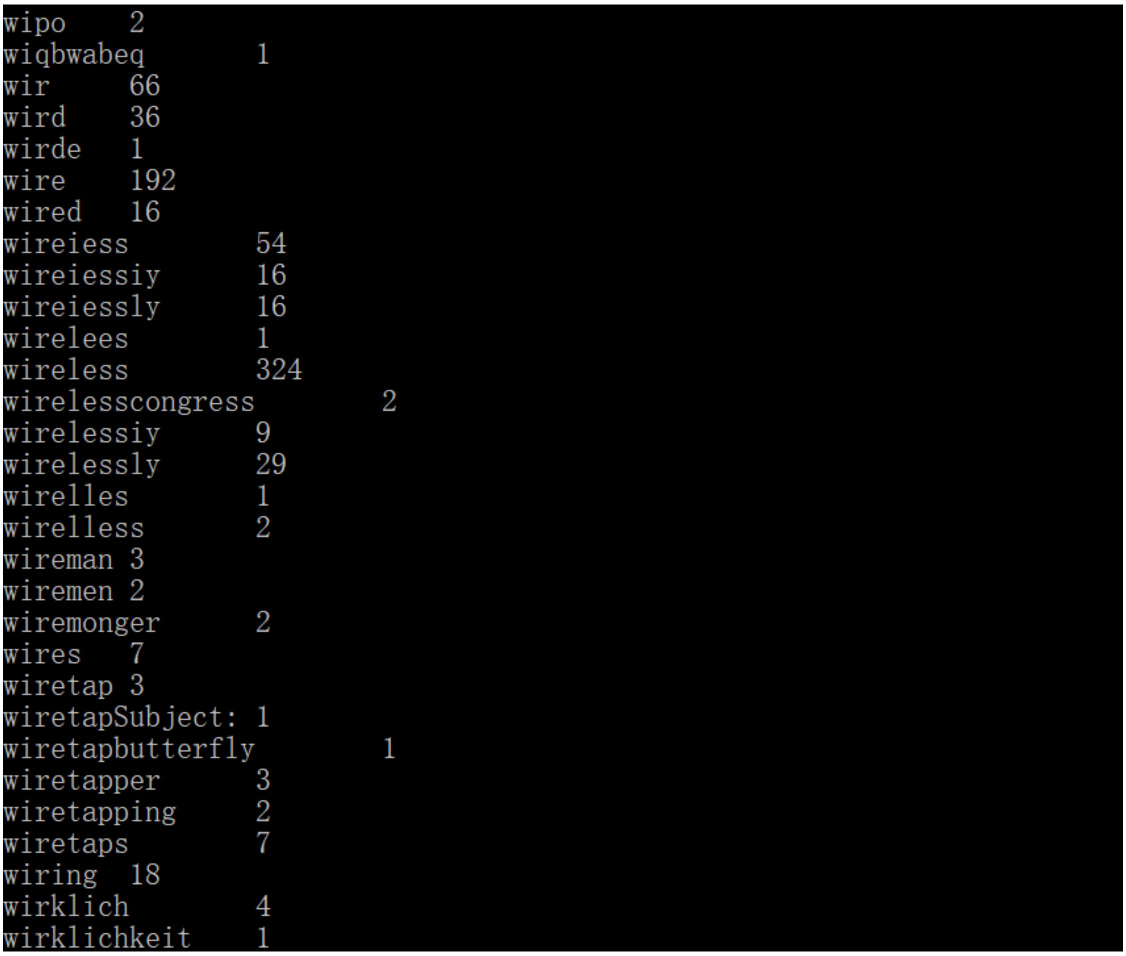
```
                FILE: Number of write operations=0
                HDFS: Number of bytes read=20548586
                HDFS: Number of bytes written=1303803
                HDFS: Number of read operations=8
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        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)=12087
                Total time spent by all reduces in occupied slots (ms)=3739
                Total time spent by all map tasks (ms)=12087
                Total time spent by all reduce tasks (ms)=3739
                Total vcore-milliseconds taken by all map tasks=12087
                Total vcore-milliseconds taken by all reduce tasks=3739
                Total megabyte-milliseconds taken by all map tasks=12377088
                Total megabyte-milliseconds taken by all reduce tasks=3828736
        Map-Reduce Framework
                Map input records=345862
                Map output records=4012762
                Map output bytes=36275692
                Map output materialized bytes=2239719
                Input split bytes=106
                Combine input records=4012762
                Combine output records=154555
                Reduce input groups=120583
                Reduce shuffle bytes=2239719
                Reduce input records=154555
                Reduce output records=120583
                Spilled Records=463665
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=238
                CPU time spent (ms)=14280
                Physical memory (bytes) snapshot=622112768
                Virtual memory (bytes) snapshot=5191544832
```



```
                CPU time spent (ms)=14280
                Physical memory (bytes) snapshot=622112768
                Virtual memory (bytes) snapshot=5191544832
                Total committed heap usage (bytes)=478150656
                Peak Map Physical memory (bytes)=408723456
                Peak Map Virtual memory (bytes)=2595500032
                Peak Reduce Physical memory (bytes)=213389312
                Peak Reduce Virtual memory (bytes)=2596044800
        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=20548480
        File Output Format Counters
                Bytes Written=1303803
root@hadoop1:~/wordcount# _
```

5. 输出统计结果，实验成功。

```
root@hadoop1:~/wordcount# hdfs dfs -cat /user/hadoop/test/part-r-00000
```



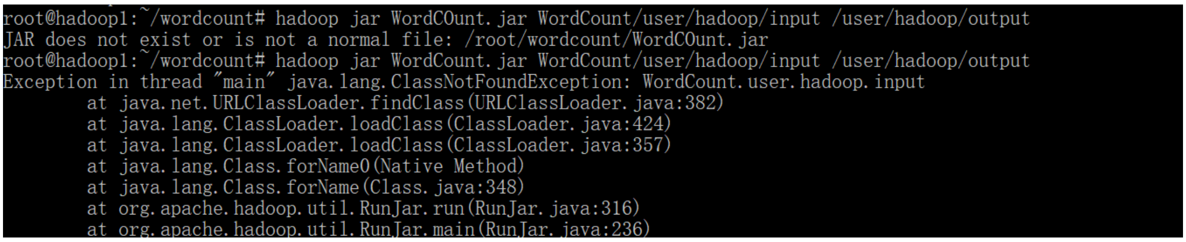
```
wipo    2
wiqbwabeq       1
wir     66
wird    36
wirde   1
wire    192
wired   16
wireiess        54
wireiessiy      16
wireiessly      16
wirelees        1
wireless        324
wirelesscongress        2
wirelessiy      9
wirelessly      29
wirelles        1
wirelless       2
wireman 3
wiremen 2
wiremonger      2
wires   7
wiretap 3
wiretapSubject: 1
wiretapbutterfly        1
wiretapper      3
wiretapping     2
wiretaps        7
wiring  18
wirklich        4
wirklichkeit    1
```

> 注：因为文件太多了，实在是太慢了，即使改了配置也要2分钟才能进行1%的Map，上次运行到50%服务器突然断开了连接白干了，因为钱不经花所以在助教的提示下，将所有spam文件通过`cat * > spam_mail`合并到一个文件，然后一下子就出结果了。

4、遇到的问题

4.1、命名空间

问题：



```
root@hadoop1:~/wordcount# hadoop jar WordCOunt.jar WordCount/user/hadoop/input /user/hadoop/output
JAR does not exist or is not a normal file: /root/wordcount/WordCOunt.jar
root@hadoop1:~/wordcount# hadoop jar WordCount.jar WordCount/user/hadoop/input /user/hadoop/output
Exception in thread "main" java.lang.ClassNotFoundException: WordCount.user.hadoop.input
        at java.net.URLClassLoader.findClass(URLClassLoader.java:382)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:424)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:357)
        at java.lang.Class.forName0(Native Method)
        at java.lang.Class.forName(Class.java:348)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:316)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
```

解决方案：

原来的`hadoop jar WordCOunt.jar WordCount /user/hadoop/input /user/hadoop/output`被修改为root@hadoop1:~/wordcount# hadoop jar WordCount.jar **org.apache.hadoop.examples.WordCount** /user/hadoop/input /user/hadoop/output

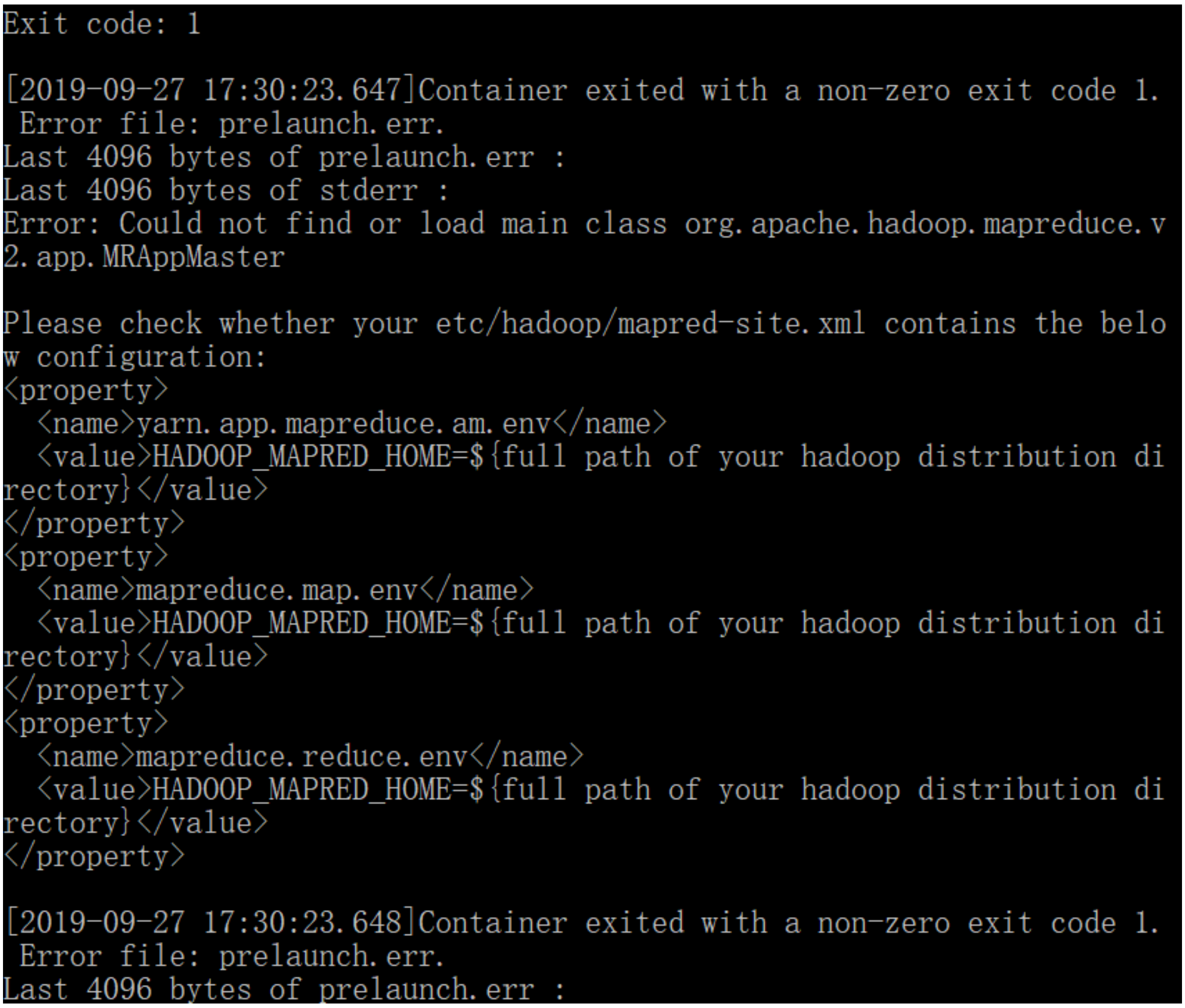
4.2、打错字

问题：user打成usr死活找不到，hadoop1的地址映射是14打成了13又折腾了半天，在此提及以警戒自己以后小心点。

解决方案：改回去。

4.3、无法加载类MPAPPMaster

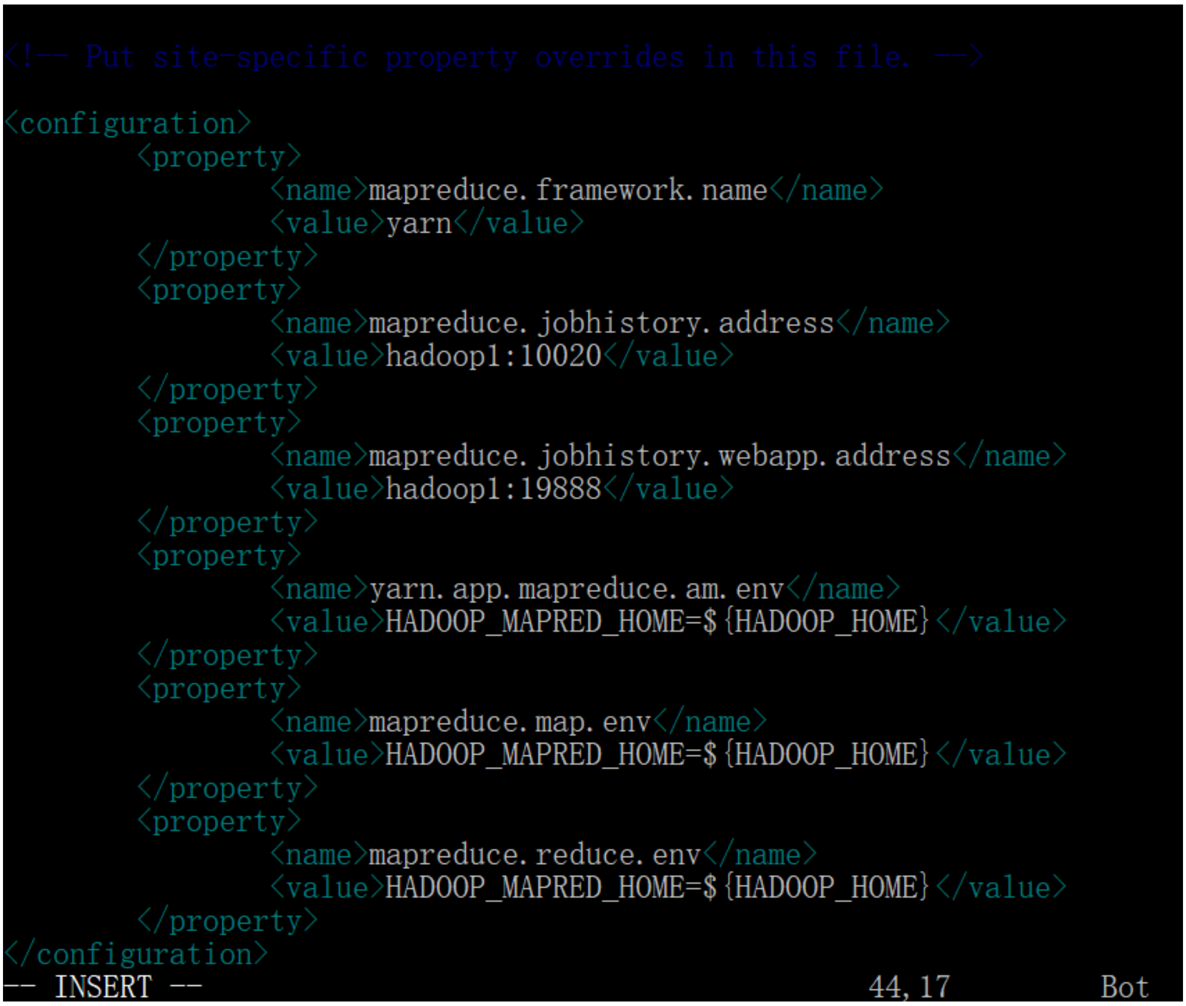
问题：环境未配置完全



```
Exit code: 1

[2019-09-27 17:30:23.647]Container exited with a non-zero exit code 1.
 Error file: prelaunch.err.
Last 4096 bytes of prelaunch.err :
Last 4096 bytes of stderr :
Error: Could not find or load main class org.apache.hadoop.mapreduce.v
2.app.MRAppMaster

Please check whether your etc/hadoop/mapred-site.xml contains the belo
w configuration:
<property>
  <name>yarn.app.mapreduce.am.env</name>
  <value>HADOOP_MAPRED_HOME=${full path of your hadoop distribution di
rectory}</value>
</property>
<property>
  <name>mapreduce.map.env</name>
  <value>HADOOP_MAPRED_HOME=${full path of your hadoop distribution di
rectory}</value>
</property>
<property>
  <name>mapreduce.reduce.env</name>
  <value>HADOOP_MAPRED_HOME=${full path of your hadoop distribution di
rectory}</value>
</property>

[2019-09-27 17:30:23.648]Container exited with a non-zero exit code 1.
 Error file: prelaunch.err.
Last 4096 bytes of prelaunch.err :
```

解决方案：

根据提示更新`mapred.xml`文件，新增如下内容：



```
<!-- Put site-specific property overrides in this file. -->

<configuration>
        <property>
                <name>mapreduce. framework. name</name>
                <value>yarn</value>
        </property>
        <property>
                <name>mapreduce. jobhistory. address</name>
                <value>hadoop1:10020</value>
        </property>
        <property>
                <name>mapreduce. jobhistory. webapp. address</name>
                <value>hadoop1:19888</value>
        </property>
        <property>
                <name>yarn. app. mapreduce. am. env</name>
                <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
        </property>
        <property>
                <name>mapreduce. map. env</name>
                <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
        </property>
        <property>
                <name>mapreduce. reduce. env</name>
                <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
        </property>
</configuration>
-- INSERT --                                                  44,17         Bot
```

4.4、配置ssh

问题：只能通过`hadoop1`访问2、3、4，无法进行通过`hadoop2`访问`hadoop3`等操作

解决方案：

利用`ssh-keygen -t rsa`命令，将生成的公钥文件保存在自己的主机上，并通过网易云服务器ssh密钥管理在其他主机上添加公钥，成功互相访问。

5、附录

在此记录每次断开后重新恢复主机时需要干的事情：

从镜像一个一个恢复服务器，申请公钥再登录hadoop1，修改hosts文件中主机名和新IP地址的映射，然后将hosts文件传送到其他三台主机中更新信息，此时需要先经过一个`ssh-keygen`命令。

```
root@hadoop1:~# sudo vi /etc/hosts
>hadoop1 新IP
>hadoop2 新IP
>hadoop3 新IP
>hadoop4 新IP

root@hadoop1:~# ssh-keygen -f "/root/.ssh/known_hosts" -R hadoop2
root@hadoop1:~# ssh-keygen -f "/root/.ssh/known_hosts" -R hadoop3
root@hadoop1:~# ssh-keygen -f "/root/.ssh/known_hosts" -R hadoop4
root@hadoop1:~# scp -r /etc/hosts root@hadoop2:/etc/
root@hadoop1:~# scp -r /etc/hosts root@hadoop3:/etc/
root@hadoop1:~# scp -r /etc/hosts root@hadoop4:/etc/
```

此外，统计结果中有部分乱码、奇怪符号的情况出现，观察原文本可见是邮件内容中出现的问题，在下一个实验中将对邮件原始数据进行预处理。

通过本次实验，配置了Hadoop环境并运行了Word Count程序，走出了我们大数据之路的第一步！