3. 分布式集群实现实验

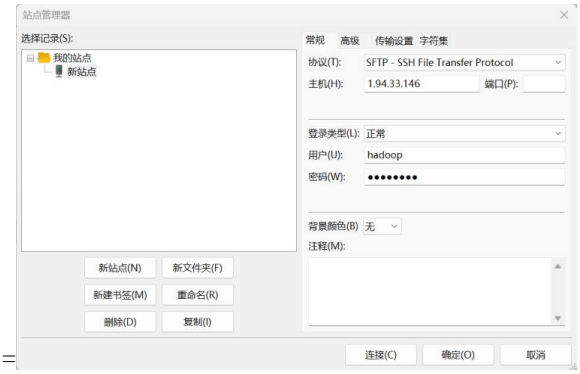
使用 hadoop 和 spark 集群，修改代码，解决相关遇到的集群问题，得到和单机版一样的结果和图片。

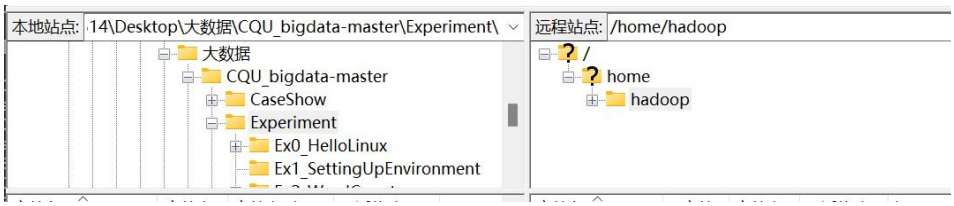
三、实验过程或算法（源程序）

1. 实验准备

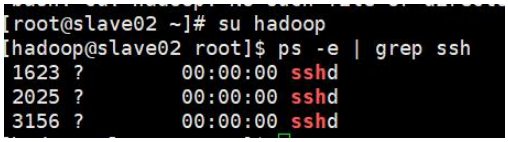
1.1 下载 FTP 工具和配置 SSH 远程开发

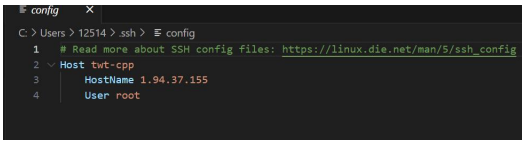
1.1.1 以下是连接 master 服务器和上传文件结果：



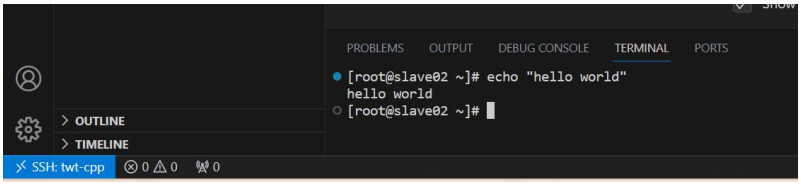


1.1.2 检查 SSH 和在 VSCode 中配置 SSH 环境：

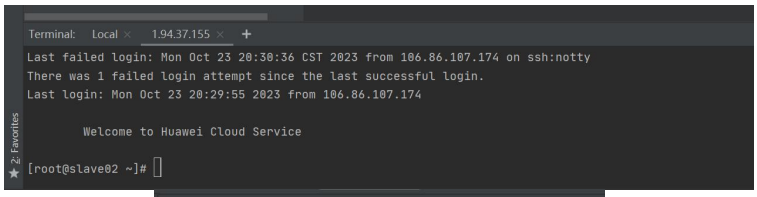


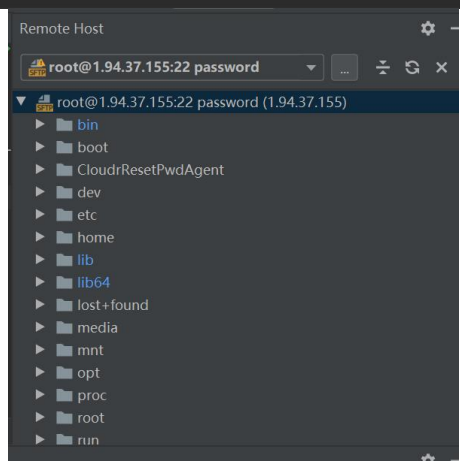


1.1.3 VSCode 测试连接：



1.1.4 Pycharm 测试连接：





1.2 安装 python 库和下载 chrome 及驱动器

1.2.1 安装字体文件

```
[hadoop@slave02 usr]$ sudo yum install wqy-microhei-fonts
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.aliyun.com
 * extras: mirrors.aliyun.com
 * updates: mirrors.aliyun.com
base
epel
extras
updates
(1/2): epel/x86_64/updateinfo
(2/2): epel/x86_64/primary_db
Package wqy-microhei-fonts-0.2.0-0.12.beta.el7.noarch already installed and latest version
Nothing to do
```

1.2.2 安装 jieba 库

```
[hadoop@slave02 usr]$ sudo pip3 install jieba -i https://pypi.tuna.tsinghua.edu.cn/simple
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Collecting jieba
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/c6/cb/18eeb235f833b726522d7ebed54f2278ce28ba9438e3135ab0278d9792a2/
     100% |                                | 19.2MB 93kB/s
Installing collected packages: jieba
  Running setup.py install for jieba ... done
Successfully installed jieba-0.42.1
[hadoop@slave02 usr]$
```

1.2.3 安装 wordcloud 库

```
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Collecting wordcloud
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/e2/e3/f2ed031cd1b1914383f822931e348f1ffe623c76b96d7a81a83113cb9aa1/wordcloud-1.9.2-cp36-cp36m-
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (438 kB)
     |                                | 438 kB 1.4 MB/s
Collecting pillow
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/ea/0f/2fa195c2d8c6fe0b3dc2df5fc6ac6b8dbd005ea30aaa0fa43eca88b8c664/Pillow-8.4.0-cp36-cp36m-man
linux_2_17_x86_64.manylinux2014_x86_64.whl (3.1 MB)
     |                                | 3.1 MB 25.0 MB/s
Collecting matplotlib
  Using cached https://pypi.tuna.tsinghua.edu.cn/packages/09/03/b7b30fa81cb687d1178e085d0f01111ceaaa3bf81f9330c937fb6f6c8ca0/matplotlib-3.3.4-cp36-cp36
-manylinux1_x86_64.whl (11.5 MB)
```

1.2.4 安装 pyecharts 库

```
[hadoop@slave02 usr]$ sudo pip3 install -i https://pypi.tuna.tsinghua.edu.cn/simple pyecharts==1.7.0
```

```
[hadoop@slave02 usr]$ sudo pip3 install snapshot-selenium
WARNING: pip is being invoked by an old script wrapper. This will fail in a future version of pip.
Please see https://github.com/pypa/pip/issues/5599 for advice on fixing the underlying issue
```

1.2.5 安装 chrome 浏览器

(1) 安装 google-chrome

```
[hadoop@slave02 usr]$ wget -O /etc/yum.repos.d/CentOS-Base.repo   http://mirrors.aliyun.com/repo/Centos-7.repo
shot https://www.baidu.com//etc/yum.repos.d/CentOS-Base.repo: Permission denied
[hadoop@slave02 usr]$
[hadoop@slave02 usr]$ curl https://intoli.com/install-google-chrome.sh | bash
```

```
[hadoop@slave02 ~]$ google-chrome --no-sandbix --headless --disable-gpu --screen
shot https://www.baidu.com/
[1023/234158.619970:WARNING:sandbox_linux.cc(393)] InitializeSandbox() called wi
th multiple threads in process gpu-process.
[1023/234158.657251:WARNING:crashpad_client_linux.cc(376)] prctl: Invalid argume
nt (22)
[1023/234158.658458:WARNING:bluez_dbus_manager.cc(247)] Floss manager not presen
t, cannot set Floss enable/disable.
[1023/234158.690770:WARNING:crashpad_client_linux.cc(376)] prctl: Invalid argume
nt (22)
[1023/234158.717341:WARNING:crashpad_client_linux.cc(376)] prctl: Invalid argume
nt (22)
3249 bytes written to file screenshot.png
```

(2) 安装 chromedriver

查看 google-chrome 版本号：

```
[hadoop@slave02 ~]$ google-chrome-stable --version
Google Chrome 118.0.5993.88
```

wget 下载：

```
[hadoop@slave02 ~]$ sudo wget https://edgedl.me.gvtl.com/edgedl/chrome/chrome-fo
r-testing/118.0.5993.88/linux64/chromedriver-linux64.zip --no-check-certificate
```

```
[hadoop@slave02 ~]$ sudo rm -f /usr/bin/chromedriver
[hadoop@slave02 ~]$ sudo rm -f /usr/local/bin/chromedriver
[hadoop@slave02 ~]$ sudo rm -f /usr/local/share/chromedriver
```

```
[hadoop@slave02 chromedriver-linux64]$ chmod 777 chromedriver
[hadoop@slave02 chromedriver-linux64]$ mv  chromedriver /usr/bin/
mv: cannot move 'chromedriver' to '/usr/bin/chromedriver': Permis
sion denied
[hadoop@slave02 chromedriver-linux64]$ sudo mv chromedriver /usr
/bin/
```

1.3 设置日志级别

```
[hadoop@slave02 conf]$ cp log4j.properties.template log4j.propert
ies
cp: cannot stat 'log4j.properties.template': No such file or dire
ctory
[hadoop@slave02 conf]$ ls
fairscheduler.xml.template   spark-defaults.conf.template
log4j2.properties.template   spark-env.sh
metrics.properties.template  spark-env.sh.template
slaves                       workers.template
[hadoop@slave02 conf]$ cp log4j2.properties.template log4j2.prope
rties
[hadoop@slave02 conf]$ vim log4j2.properties
```

修改 log4j.rootCategory=WARN,console

```
# Set everything to be logged to the console
rootLogger.level = WARN
rootLogger.appenderRef.stdout.ref = console
```

2. 实验流程

2.1 jiebaCut 函数编写

jiebacut()函数能够将一个字符串分成词语，并返回一个列表，这里需要做的是完善 reduce 函数，reduce 函数的参数是一个函数，这里使用 python 的 lambda 表达式，代码如下：

```
def jiebaCut(answers_filePath):
    """
    结巴分词
    :param answers_filePath: answers.txt路径
    :return:
    """
    # 读取answers.txt
    answersRdd = sc.textFile(answers_filePath) # answersRdd每一个元素对应answers.txt每-

    # 利用SpardRDD reduce()函数,合并所有回答
    # 【现在你应该完成下面函数编码】

    str = answersRdd.reduce(lambda a,b: a + b )

    # jieba分词
    words_list = jieba.lcut(str)
    return words_list
```

2.2 wordcount 函数编写

wordcount 函数的作用就是将 RDD 中的词语进行过滤、键值映射、键合并。所用的 fitlter 函数参数为函数（这里是 lambda 表达式），返回值是一个 bool 变量；map 函数用于建立键值关系，reduceByKey 函数用于合并键，sortBy 函数能够排序 RDD；参考官方文档后补全相关代码，关键代码块块如下:

```
    # wordcount：去除停用词等同时对最后结果按词频进行排序
    # 完成SparkRDD操作进行词频统计
    # 提示：你应该依次使用
    #       1.filter函数分别进行停用词过滤、去除长度<=1的词汇
    #       2.map进行映射, 如['a','b','a'] --> [('a',1),('b',1),('a',1)]
    #       3.reduceByKey相同key进行合并 [('a',2),('b',1)]
    #       4.sortBy进行排序, 注意应该是降序排序
    #【现在你应该完成下面函数编码】
    resRdd = wordsRdd.filter(lambda word: word not in stopwords) \
                     .filter(lambda word:len(word) > 1 )\
                     .map(lambda word:(word,1)) \
                     .reduceByKey(lambda a, b: a+b) \
                     .sortBy(ascending=False, numPartitions=None, keyfunc=lambda x: x[1
```

2.3 visualize.py 函数编写

rdd2dic 将 rdd 转化为 python 列表，这里主要用到了 rdd 的 collectAsMap 函数，相关代码块如下：

```
将RDD转换为Dic，并截取指定长度topK
:param resRdd: 词频统计降序排序结果RDD
:param topK: 截取的指定长度
:return:
"""

# 提示：SparkRdd有函数可直接转换
# 【现在你应该完成下面函数编码】
resDic =  resRdd.collectAsMap()
# resDic =
# 截取字典前K个
K = 0
wordDicK = {}
for key, value in resDic.items():
    # 完成循环截取字典
    wordDick[key] = value
    K += 1
    if K == topK:
        break
return wordDicK
```

2.4 提交代码

```
if __name__ == '__main__':

    # 进行词频统计并可视化
    resRdd = wordcount(isvisualize=True)
    print(resRdd.take(10))  # 查看前10个
```

```
[hadoop@slave02 spark]$ bin/spark-submit /home/hadoop/Experiment/Ex2_WordCount/WordCount.py
23/10/24 15:09:08 INFO spark.SparkContext: Running Spark version 3.4.1
```

3. 拓展分布式集群实现

3.1 上传数据到 hadoop 文件集群（此处必须在启动 hadoop 集群启动下完成！）

```
[hadoop@master hadoop]$ cd /usr/local/hadoop
[hadoop@master hadoop]$ ./bin/hadoop fs -mkdir -p /ex/ex2dataset
```

```
[hadoop@master hadoop]$ ./bin/hadoop fs -put /home/hadoop/Experiment/Ex2_WordCount/src/answers.txt /ex/ex2dataset
```

```
[hadoop@master hadoop]$ ./bin/hadoop fs -ls -R /
drwxr-xr-x   - hadoop supergroup          0 2023-10-30 11:17 /ex
drwxr-xr-x   - hadoop supergroup          0 2023-10-30 11:32 /ex/ex
2dataset
-rw-r--r--   3 hadoop supergroup    6178784 2023-10-30 11:20 /ex/ex
2dataset/answers.txt
-rw-r--r--   3 hadoop supergroup      14112 2023-10-30 11:32 /ex/ex
2dataset/stop_words.txt
```

3.2 修改文件读取路径

```
#SRCPATH = '/home/hadoop/Experiment/Ex2_WordCount/src/'
SRCPATH = 'hdfs://master:9000/ex/ex2dataset/'
```

集群下使用 sc.textFile 进行读取。

```python
def getStopWords(stopWords_filePath):
    print("开始读stopword")
    stopwords = sc.textFile(stopWords_filePath).collect()
    print("stopword 读取 成功！ ")
    return stopwords
```

3.3 修改 spark 环境设置，改为集群启动

```python
conf = SparkConf().setAppName("ex2").setMaster("spark://master:7077")
# conf = SparkConf().setAppName("ex2").set("spark.task.maxFailures", "3")  # 设置最大重试次数为10次
# conf = SparkConf().setAppName("ex2").setMaster("local")
sc = SparkContext(conf=conf)
```

3.4 修改算法核心代码（解决 spark 集群下的对数据处理慢导致报错的问题，原因将在后续分析）

```python
def jiebaCut(answers_filePath):
    """
    结巴分词
    :param answers_filePath: answers.txt路径
    :return:
    """
    # 读取answers.txt
    answersRdd = sc.textFile(answers_filePath)  # answersRdd每一个元素对应answers.txt每一行

    # 获取RDD的行数
    lines = answersRdd.zipWithIndex()
    print("rdd to str")
    print(lines.count())
    words_list = []
    start_line = 0
    batch_size = 30
    # 逐批处理数据
    while start_line < lines.count():
        # 读取指定批次的数据
        print(start_line)
        tempRdd = lines.filter(lambda x: start_line <= x[1] < start_line + batch_size).map(lambda x: x[0])

        # 使用treeReduce函数合并数据
        combined_str = tempRdd.treeReduce(lambda a, b: a + b)

        # jieba分词
        words_list.extend(jieba.lcut(combined_str))
        start_line += batch_size

    return words_list
```

3.5 代码提交，提交格式以及语法

```
[hadoop@master spark]$ bin/spark-submit --master spark://master:7077 --executor-memory 5G /home/hadoop/Experiment/Ex2_WordCount/WordCount.py
log4j:WARN No appenders could be found for logger (org.apache.spark.util.ShutdownHookManager).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
```

启动语法如下：

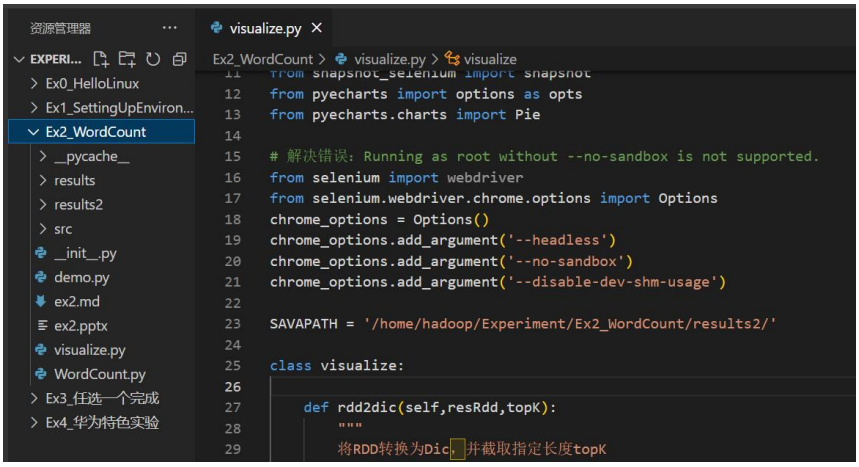
bin/spark-submit --master spark://master:7077 --executor-memory 5G /home/hadoop/Experiment/Ex2_WordCount/WordCount.py

其中 spark://master:7077，7077 是 spark 的端口，--executor-memory 5G 是给每个 slave 规定的运行的内存。

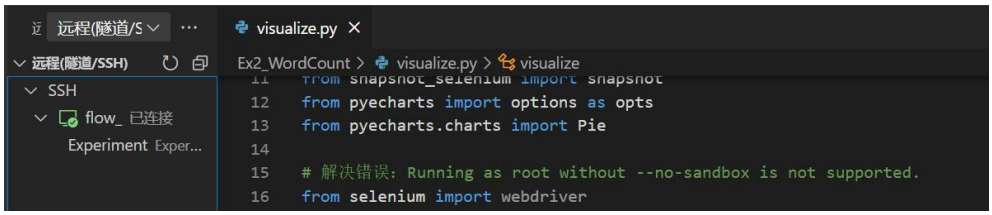
四、实验结果及分析和（或）源程序调试过程

1. 实验准备

在远程和本地均安装 Remote-SSH，并连接到服务器，选择上传的文件夹并打开，能够正确打开，并在此基础上进行远程开发：



查看左侧导航栏，显示已连接，成功登陆：



2. 安装相关库

2.1 安装字体文件

可以看到字体文件夹下有本实验所需的字体文件：

```
[hadoop@master fonts]$ ls
```

```
wqy-microhei
```

2.2 安装 jieba

pip show 命令查看是否安装成功 jieba，正确输出了版本号和相关信息：

```
[hadoop@master /]$ pip show jieba
Name: jieba
Version: 0.42.1
Summary: Chinese Words Segmentation Utilities
Home-page: https://github.com/fxsjy/jieba
```

2.3 安装 wordcloud

pip show 命令查看是否安装成功 wordcloud，正确输出了版本号和相关信息：

```
[hadoop@master /]$ pip show wordcloud
Name: wordcloud
Version: 1.8.0
Summary: A little word cloud generator
Home-page: https://github.com/amueller/word_cloud
```

2.4 安装 pyecharts

pip show 命令查看是否安装成功 pyecharts，正确输出了版本号和相关信息：

```
[hadoop@master /]$ pip show pyecharts
Name: pyecharts
Version: 1.7.0
Summary: Python options, make charting easier
Home-page: https://github.com/pyecharts/pyecharts
```

2.5 安装驱动

2.5.1 安装 google-chrome

查看 google-chrome 版本，能够正确显示版本号，根据对应的版本号安装 driver：

```
[hadoop@master root]$ google-chrome-stable --version
Google Chrome 118.0.5993.88
```

2.5.2 安装 chromedriver

将 chromedriver 移动到/usr/bin 路径下，ls 查看当前目录，已经成功安装 chromedriver 并移动到相应路径：

```
[hadoop@master ~]$ cd /usr/bin/
[hadoop@master bin]$ ls
```

```
chromedriver
```

3. 完善并提交代码

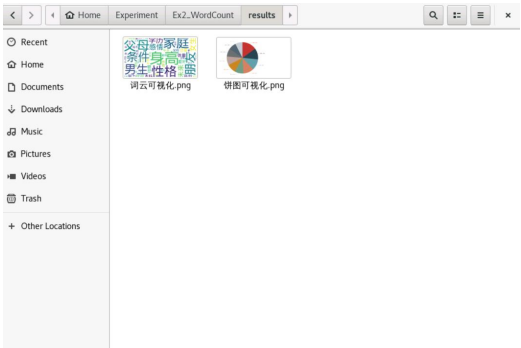
3.1 词频统计

完善代码中内容之后并提交，此时可视化有效位为 FALSE，可以看到成功建立了前缀字典，resRdd 包含了词频统计结果，且按照降序排列，并正确打印了 resRdd 前十个元素：

```
Prefix dict has been built successfully.
[('身高', 2627), ('家庭', 2018), ('父母', 2002), ('性格', 1882), ('
男生', 1640), ('朋友', 1618), ('条件', 1568), ('学历', 1445), ('女
生', 1380), ('感情', 1301)]
```

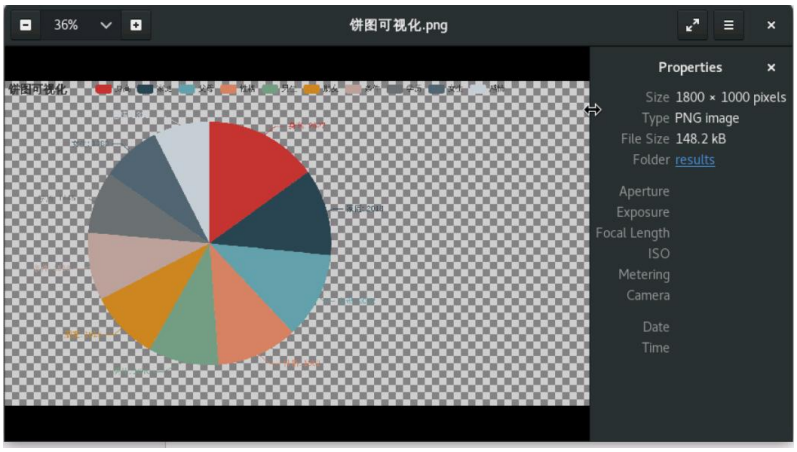
3.2 结果可视化

完善可视化代码并提交，将可视化有效位置位，在指定的保存路径下可以看到正确生成的词云图和饼状图：



可以看到生成的词云图中正确显示了 resRdd 包含的词频统计结果：

可以看到生成的饼状图中也中正确显示了 resRdd 包含的词频统计结果：



4. 分布式集群下的结果

4.1 上传文件到 hdfs

```
[hadoop@master hadoop]$ ./bin/hadoop fs -ls -R /
drwxr-xr-x   - hadoop supergroup          0 2023-10-30 11:17 /ex
drwxr-xr-x   - hadoop supergroup          0 2023-10-30 11:32 /ex/ex
2dataset
-rw-r--r--   3 hadoop supergroup    6178784 2023-10-30 11:20 /ex/ex
2dataset/answers.txt
-rw-r--r--   3 hadoop supergroup      14112 2023-10-30 11:32 /ex/ex
2dataset/stop_words.txt
```

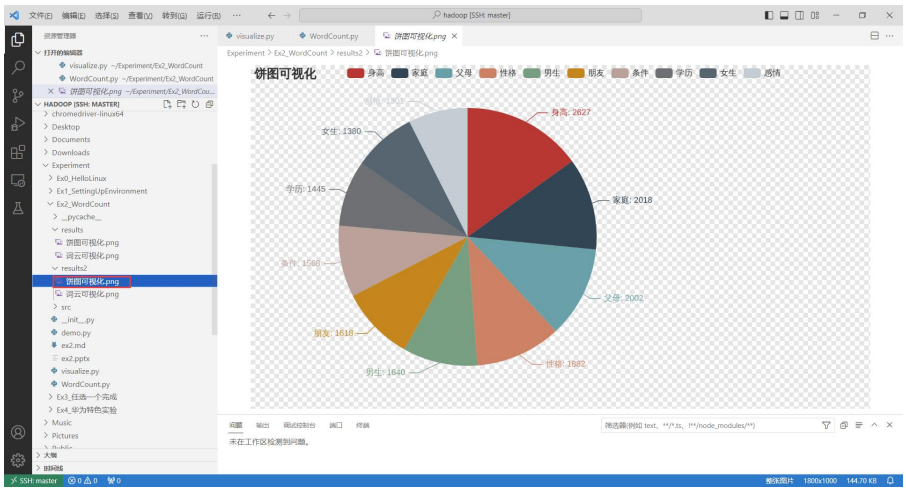
4.2 终端输出（附带 debug 时的相关输出信息）：

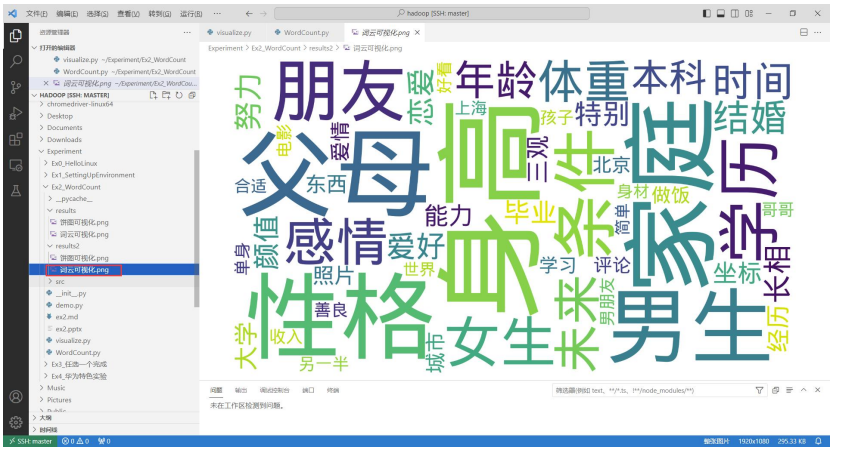
```
Building prefix dict from the default dictionary ...
Loading model from cache /tmp/jieba.cache
Loading model cost 0.566 seconds.
Prefix dict has been built successfully.
30
60
90
120
150
180
210
240


2760
2790
2820
2850
2880
2910
2940
2970
3000
3030
3060
3090
55结巴分词完成
57行,词统计完成
70行resRdd排序完成
饼图-**-**-*-*-*-*-
词云可视化-**-**-*-*-*-*-
程序结束----------
[('身高', 2627), ('家庭', 2018), ('父母', 2002), ('性格', 1882), ('男生', 1640), ('朋友', 1618), ('条件', 1568), ('学历', 1445), ('女生', 1380),
 ('感情', 1301)]
[hadoop@master spark]$
```

可以看到，符合之前的处理逻辑，每三十行一个 batch 处理一次 str，从而可以解决 Spark 集群下的对 str 进行大量合并导致的报错。

4.3 图片结果：





上面分别为集群代码中存入本地 result2 中的两个图片。

4.4 前往 spark 网站查看集群运行成功的截图：

app-20231030230247-0007	ex2	4	5.0 GiB		2023/10/30 23:02:47	hadoop	FINISHED	17 min

可以看到，运行的时间还是很久（主要的原因就是分布式下的存储方式，导致的 spark 对数据格式的转变或者的较长字符串的合并操作的时间需求非常的高），长达 17min。

五、遇到的问题及解决方案

1. 单机版下遇到的问题和解决方案

1.1 安装 google-chrome 和 chrome-driver

如果按照实验指导书以下步骤进行，会下载官方正处于**测试版的版本而非稳定版**

- 安装google-chrome

```
wget -O /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.aliyun.com/repo/Centos-7.repo

curl https://intoli.com/install-google-chrome.sh | bash
ldd /opt/google/chrome/chrome | grep "not found"

google-chrome --no-sandbox --headless --disable-gpu --screenshot
https://www.baidu.com/
```

- 安装chromedrive

  chromedrive版本要和google-chrome对应，所以我们先查看google-chrome版本号：

```
google-chrome-stable --version
```

  记录版本号后，去https://npm.taobao.org/mirrors/chromedriver/ 下载对应的驱动。下载方式：

可能会产生以下问题：

• 在淘宝镜像站上无法找到最新测试版的驱动器

• 在官网上找到的驱动器可能处于 404 无法下载或不可用问题(目前官网暂时正常)

Upcoming version: 118.0.5993.117 (r1192594)

Binary	Platform	URL	HTTP status
chrome	linux64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/linux64/chrome-linux64.zip	404
chrome	mac-arm64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-arm64/chrome-mac-arm64.zip	404
chrome	mac-x64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-x64/chrome-mac-x64.zip	404
chrome	win32	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/win32/chrome-win32.zip	404
chrome	win64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/win64/chrome-win64.zip	404
chromedriver	linux64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/linux64/chromedriver-linux64.zip	404
chromedriver	mac-arm64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-arm64/chromedriver-mac-arm64.zip	404
chromedriver	mac-x64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-x64/chromedriver-mac-x64.zip	404
chromedriver	win32	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/win32/chromedriver-win32.zip	404
chromedriver	win64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/win64/chromedriver-win64.zip	404

• 在后续执行 python 时可能会产生以下问题：

```
File "/usr/local/lib/python3.6/site-packages/selenium/webdriver/remote/errorhandler.py", line 242, in check_response
  raise exception_class(message, screen, stacktrace)
elenium.common.exceptions.SessionNotCreatedException: Message: session not created: Chrome failed to start: exited normally.
(session not created: DevToolsActivePort file doesn't exist)
(The process started from chrome location /opt/google/chrome/chrome is no longer running, so ChromeDriver is assuming that Chro
has crashed.)
```

解决方案：chrome 和 chrome-driver 均采用 zip 方式安装(Chrome for Testing availability (googlechromelabs.github.io)，且下载稳定版本.

以下是建议修改的安装步骤：

```
# 在hadoop下运行
sudo yum remove google-chrome-stable  # 删除原有版本chrome
cd /usr/bin
rm -f chromedriver  # 删除原有chromedriver
cd /opt/google
rm -rf chrome       # 执行完后google文件夹应为空,删不掉问题应该也不大
```

下载特定版本的chrome：参考网址Chrome for Testing availability (googlechromelabs.github.io)**(注意以下操作初始文件夹仍在/opt/google)**

```
wget https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/linux64/chrome-linux64.zip
unzip chrome-linux64.zip
mv chrome-linux64 chrome   # 前提是上面已经删除了chrome文件夹，如果没有删掉，就把chrome-linux64里面的所有文件移动到/opt/google/chrome/下即可
```

总的来说，就是解压后放在/opt/google/chrome/目录下

设置环境变量

```
vim ~/.bashrc
```

修改：注意前后的冒号(在hadoop下)

export PATH=$HADOOP_HOME/bin:$SPARK_HOME/bin**:/opt/google/chrome/:**$PATH

如果你本来就没有PATH，那么建议你把/usr/bin:/usr/local/bin也加上

export PATH=$HADOOP_HOME/bin:$SPARK_HOME/bin:**/usr/bin:/usr/local/bin:/opt/google/chrome/:**$PATH

保存退出

```
source ~/.bashrc
```

输入：chrome --version　有版本号则正确

-------chromedriver同实验指导书

1.2 **设置日志级别：在 conf 文件中 log4j2 已经替换掉了 log4j,这是因为前者比后者更安全，并且 log4j2 中的文件内容与 log4j 有一定差异。**

1. 切换到`conf` 目录

```
cd /usr/local/spark/conf
```

2. 设置配置文件

```
cp log4j.properties.template log4j.properties
vim log4j.properties
```

bash

修改 `log4j.rootCategory=WARN,console`

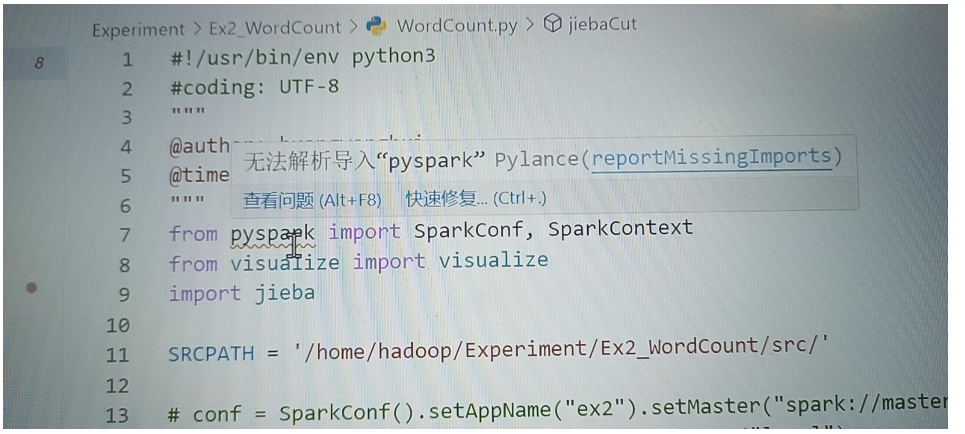
解决方案：将设置配置文件内容修改为：

cp log4j2.properties.template log4j.properties

vim log4j.properties

在 log4j.properties 中所有的“info”字段修改为“warn”即可.

1.3 Python 远程开发无法解析 pyspark：



解决方案：

方案 1：pip3 install pyspark（下载很慢，并且后续运行可能出现版本不兼容的

问题，因此建议使用方案 2)

方案 2：将已下载的 spark 中的 pyspark 包移动到 site-packages 中：

cd /usr/local/spark/python/

sudo cp pyspark -r /usr/local/lib/python3.6/site-packages/

1.4 运行时错误：在自己按照 zip 安装后仍然产生以下错误：

```
 File "/usr/local/lib/python3.6/site-packages/selenium/webdriver/remote/errorhandler.py", line 242, in check_response
   raise exception_class(message, screen, stacktrace)
elenium.common.exceptions.SessionNotCreatedException: Message: session not created: Chrome failed to start: exited normally.
 (session not created: DevToolsActivePort file doesn't exist)
 (The process started from chrome location /opt/google/chrome/chrome is no longer running, so ChromeDriver is assuming that Chro
has crashed.)
```

解决方案：

找到 site-packages/snapshot-selenium/snapshot.py 文件，修改 get_chrome_driver 如下：

```
def get_chrome_driver():
    options = webdriver.ChromeOptions()
    options.add_argument("headless")
    options.add_argument('--no-sandbox')
    options.add_argument('--disable-gpu')
    options.add_argument('--disable-dev-shm-usage')
     return webdriver.Chrome(options=options)
```

*如果 snapshot.py 中没有 get_chrome_driver()而只有 get_chrome()，可以进行重装：

pip3 uninstall snapshot-selenium

pip3 install snapshot-selenium

1.5 运行时错误

```
Traceback (most recent call last):
  File "/home/hadoop/Ex2_WordCount/WordCount.py", line 84, in <module>
    resRdd = wordcount(isvisualize=True)
  File "/home/hadoop/Ex2_WordCount/WordCount.py", line 75, in wordcount
    v.drawPie(pieDic)
  File "/home/hadoop/Ex2_WordCount/visualize.py", line 101, in drawPie
    make_snapshot(snapshot, pie_position().render(), SAVAPATH + '饼图可视化.png')
  File "/usr/local/lib/python3.6/site-packages/pyecharts/render/snapshot.py", line 37, in make_snapshot
    **kwargs,
  File "/usr/local/lib/python3.6/site-packages/snapshot_selenium/snapshot.py", line 52, in make_snapshot
    return driver.execute_script(snapshot_js)
  File "/usr/local/lib/python3.6/site-packages/selenium/webdriver/remote/webdriver.py", line 636, in execute_script
    'args': converted_args})['value']
  File "/usr/local/lib/python3.6/site-packages/selenium/webdriver/remote/webdriver.py", line 321, in execute
    self.error_handler.check_response(response)
  File "/usr/local/lib/python3.6/site-packages/selenium/webdriver/remote/errorhandler.py", line 242, in check_response
    raise exception_class(message, screen, stacktrace)
selenium.common.exceptions.JavascriptException: Message: javascript error: echarts is not defined
  (Session info: headless chrome=118.0.5993.70)
```

一种可能的解决方案也是：

pip3 uninstall snapshot-selenium

pip3 install snapshot-selenium

1.6 权限不足或文件路径没找到：

```
  File "/usr/local/lib/python3.6/site-packages/pyecharts/render/snapshot.py", line 52, in make_snapshot
    save_as_png(image_data, output_name)
  File "/usr/local/lib/python3.6/site-packages/pyecharts/render/snapshot.py", line 77, in save_as_png
    with open(output_name, "wb") as f:
FileNotFoundError: [Errno 2] No such file or directory: '/home/hadoop/Experiment/Ex2_WordCount/results/饼图可视化.png'
23/10/29 23:38:04 INFO SparkContext: Invoking stop() from shutdown hook
23/10/29 23:38:04 INFO SparkContext: SparkContext is stopping with exitCode 0.
23/10/29 23:38:04 INFO SparkUI: Stopped Spark web UI at http://slave02:4040
```

解决方案：自己新建一个 results 文件夹并修改权限：

mkdir results

sudo chmod 777 -R results

1.7 权限生成词云这块遇到以下报错：

```
Traceback (most recent call last):
  File "/home/hadoop/Ex2_WordCount/WordCount.py", line 84, in <module>
    resRdd = wordcount(isvisualize=True)
  File "/home/hadoop/Ex2_WordCount/WordCount.py", line 78, in wordcount
    v.drawWorcCloud(wwDic)
  File "/home/hadoop/Ex2_WordCount/visualize.py", line 68, in drawWorcCloud
    wc.generate_from_frequencies(wordDic)
  File "/usr/local/lib64/python3.6/site-packages/wordcloud/wordcloud.py", line 454, in generate_from_frequ
    max_font_size=self.height)
  File "/usr/local/lib64/python3.6/site-packages/wordcloud/wordcloud.py", line 508, in generate_from_frequ
    box_size = draw.textbbox((0, 0), word, font=transposed_font, anchor="lt")
  File "/usr/local/lib64/python3.6/site-packages/PIL/ImageDraw.py", line 651, in textbbox
    raise ValueError("Only supported for TrueType fonts")
ValueError: Only supported for TrueType fonts
```

解决方案：

只要下载 1.8.0 版本的 wordcount 就可以解决问题：

sudo pip3 install wordcloud==1.8.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

2. Hadoop 集群下遇到的问题和解决方案

2.1 在使用分布式集群时，master 上传文件失败：

```
[hadoop@master root]$ cd /usr/local/hadoop
[hadoop@master hadoop]$ ./bin/hadoop fs -mkdir -p /ex/ex3dataset
mkdir: Call From master/192.168.0.251 to master:9000 failed on conn
ection exception: java.net.ConnectException: Connection refused; Fo
r more details see:  http://wiki.apache.org/hadoop/ConnectionRefuse
d
```

解决方案：

实验报告中未写清楚，在使用 master 上传文件到 hadoop 文件系统上时，需要先启动 hadoop 集群，最好 spark 集群也启动。

2.2 问题三解决中，我们发现出现防火墙也会导致无法上传文件，使用 nmap 指令无法 ping 通其他主机.

正常应该如下：

```
[hadoop@master hadoop]$ nmap -p 9000 1.94.33.146

Starting Nmap 6.40 ( http://nmap.org ) at 2023-10-29 22:35 CST
Nmap scan report for ecs-1-94-33-146.compute.hwclouds-dns.com (1.94
.33.146)
Host is up (0.0015s latency).
PORT     STATE SERVICE
9000/tcp open  cslistener
```

报错如下：

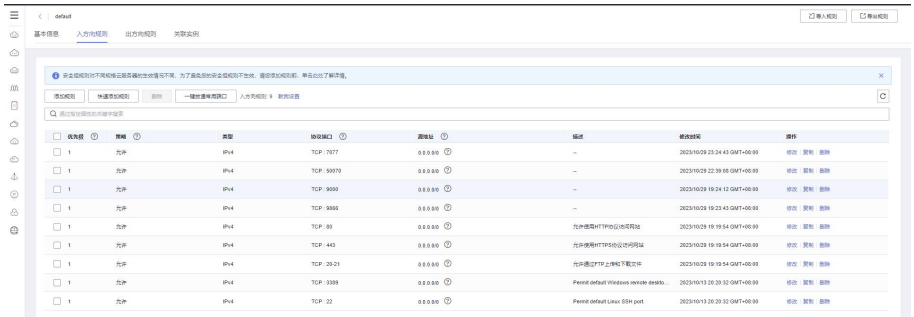
```
[hadoop@slave01 root]$ nmap -p 7077 1.94.11.79

Starting Nmap 6.40 ( http://nmap.org ) at 2023-10-29 23:43 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.02 seconds
```

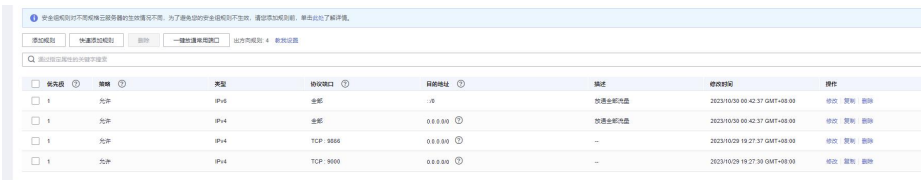
解决方法：关闭防火墙

如果关闭后仍然无法 ping 通，去华为云端口放开节点，此处建议安全组入协议和出协议全部放开端口（但是为了安全性考虑，定位问题后建议如下图放开端口）：

入端口：



出端口：



如果还是无法 ping 通，报错仍然为 host down，可以尝试多开启关闭防火墙几次，如果还是出现该报错，需要关闭服务器等待一段时间后自然会恢复（此种情况在其他小组极为少见，但是本小组同学实践中华为云弹性服务器就出现了此种情况，等待 1~2 小时候自然就恢复了，原因不详，较为玄学）。

2.3 在使用分布式集群时，出现了文件读取报错，报错位置是下面这行代码：

```
def getStopWords(stopWords_filePath):
    stopwords = [line.strip() for line in open(stopWords_filePath, 'r', encoding='utf-8').readlines()]
    return stopwords
```

报错信息为：无法读取到文件（但是我们可以用 cat 的方法直接拉去到该文件信息，说明路径和文件都是正确的）：

```
'r', encoding='utf-8').readlines()]
FileNotFoundError: [Errno 2] No such file or directory: 'hdfs://1.9
4.33.146:9000/ex/ex2dataset/stop_words.txt'
```

解决方法：

在集群下，似乎不支持 with open 这种语法打开集群的文件，所以我们采用以下方式进行：

```
def getStopWords(stopWords_filePath):
    print("开始读stopword")
    stopwords = sc.textFile(stopWords_filePath).collect()
    print("stopword 读取 成功！")
    return stopwords
```

2.4 一切路径和环境问题解决完毕后，仍然运行中出现了报错：

```
: org.apache.spark.SparkException: Job aborted due to stage failure
: Task 1 in stage 0.0 failed 4 times, most recent failure: Lost tas
k 1.3 in stage 0.0 (TID 6) (192.168.0.204 executor 0): TaskResultLo
st (result lost from block manager)
Driver stacktrace:
        at org.apache.spark.scheduler.DAGScheduler.failJobAndIndepe
ndentStages(DAGScheduler.scala:2785)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSt
age$2(DAGScheduler.scala:2721)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSt
age$2$adapted(DAGScheduler.scala:2720)
        at scala.collection.mutable.ResizableArray.foreach(Resizabl
eArray.scala:62)
```

报错的理由是：在运行到某一步时，出现了某个任务长期等待，四次尝试联系全部失败，所以丢失并报错。

解决方法：

此处报错花费了本小组近一天时间进行 debug，总结下来可能的解决方案有以下几个：

1. 资源倾斜导致的运算速度不平衡从而导致的等待超时。此时可以考虑提升云服务器配置或者减少 slave 节点个数来实现。
2. Spark 集群中使用的存储方式较为特殊，Rdd 存储中对于数据格式的转换速度极慢，我们通过打印字符串的方式定位到了引起该报错的代码：

```
str = answersRdd.reduce(lambda a, b: a+b)
```

分析原因：

因为在 spark 集群下，他对于数据的存储是较为特殊的，所以对于上述的操作特别花费时间，极有可能出现长时间当代某一步执行完成的情况。

（发现的原因是，我们使用较小数据集的 answers.txt 文件，我们集群能够顺利且快速的执行成功（小文件是完全版文件大小的的 1/30），但是使用完全版文件进行时，这

一步永远超时报错，所以我们采用如下解决方法）

解决方法：

由于考虑到一次那行将所有的行全部加在一起过大，并且分布式存储的原因导致这一步处理更加困难，所以我们想到了分批次处理，我们设置了一个 batch，batch 的大小根据实际情况确定，此处为 30 行，我们每三十行处理一次 str，对其进行结巴分词，并且将结果记录到列表中，然后再处理下一个 batch，此种操作下可以完美的避开由于 spark 集群存储特点导致的合并大量的字符串 str 导致的超时错误。

```python
def jiebaCut(answers_filePath):
    """
    结巴分词
    :param answers_filePath: answers.txt路径
    :return:
    """
    # 读取answers.txt
    answersRdd = sc.textFile(answers_filePath)  # answersRdd每一个元素对应answers.txt每一行

    # 获取RDD的行数
    lines = answersRdd.zipWithIndex()
    print("rdd to str")
    print(lines.count())
    words_list = []
    start_line = 0
    batch_size = 30
    # 逐批处理数据
    while start_line < lines.count():
        # 读取指定批次的数据
        print(start_line)
        tempRdd = lines.filter(lambda x: start_line <= x[1] < start_line + batch_size).map(lambda x: x[0])

        # 使用treeReduce函数合并数据
        combined_str = tempRdd.treeReduce(lambda a, b: a + b)

        # jieba分词
        words_list.extend(jieba.lcut(combined_str))
        start_line += batch_size

    return words_list
```

结果如下：

```
2760
2790
2820
2850
2880
2910
2940
2970
3000
3030
3060
3090
55结巴分词完成
57行,词统计完成
70行resRdd排序完成
饼图-**-**-*-*-*-*-
词云可视化-**-**-*-*-*-*-
程序结束----------
[('身高', 2627), ('家庭', 2018), ('父母', 2002), ('性格', 1882), ('男生', 1640), ('朋友', 1618), ('条件', 1568), ('学历', 1445), ('女生', 1380),
 ('感情', 1301)]
[hadoop@master spark]$
```