写在前面：实验指导书和参考代码中cck-0003为主节点，同学们请自行修改为主节点名字。

1. **安装Kafka**

1、解压kafka

tar -zxvf kafka\_2.11-0.10.2.2.tgz

2、将解压得到的文件夹移到/home/modules目录下

mv kafka\_2.11-0.10.2.2 /home/modules/

3、编辑config/server.properties文件，修改delete.topic.enable和zookeeper.connect

vim /home/modules/kafka\_2.11-0.10.2.2/config/server.properties





4、将kafka文件夹通过scp发送到其余结点对应目录下

scp -r /home/modules/kafka\_2.11-0.10.2.2 root@cck-0001:/home/modules/

scp -r /home/modules/kafka\_2.11-0.10.2.2 root@cck-0002:/home/modules/

scp -r /home/modules/kafka\_2.11-0.10.2.2 root@cck-0004:/home/modules/

5、编辑config/server.properties文件，修改broker.id分别为1、2、3、4（各节点不一样）

vim /home/modules/kafka\_2.11-0.10.2.2/config/server.properties



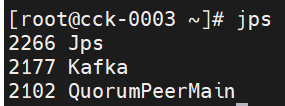
6、各节点启动zookeeper

/usr/local/zookeeper/bin/zkServer.sh start

7、各节点启动kafka（下面这是一条命令，只是word一行写不下了）

/home/modules/kafka\_2.11-0.10.2.2/bin/kafka-server-start.sh /home/modules/kafka\_2.11-0.10.2.2/config/server.properties

8、jps确认kafka启动成功（10分）



1. **安装Redis（单机部署）**
2. 升级gcc

①yum -y install centos-release-scl

②yum -y install devtoolset-9-gcc devtoolset-9-gcc-c++ devtoolset-9-binutils devtoolset-9-libatomic-devel

③scl enable devtoolset-9 bash

1. 解压redis

tar -zxvf redis-6.0.6.tar.gz

1. 进入解压得到的文件夹，编译，安装

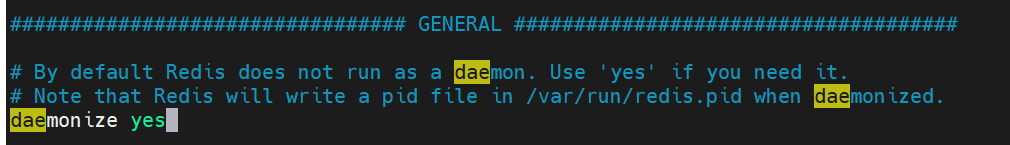
cd redis-6.0.6

make

make install

1. 修改redis.conf，设置redis为守护进程，并允许远程连接，关闭保护模式

vim /root/redis-6.0.6/redis.conf







1. 启动redis

redis-server redis-6.0.6/redis.conf

1. ps确认redis运行成功（10分）



1. **安装python3**

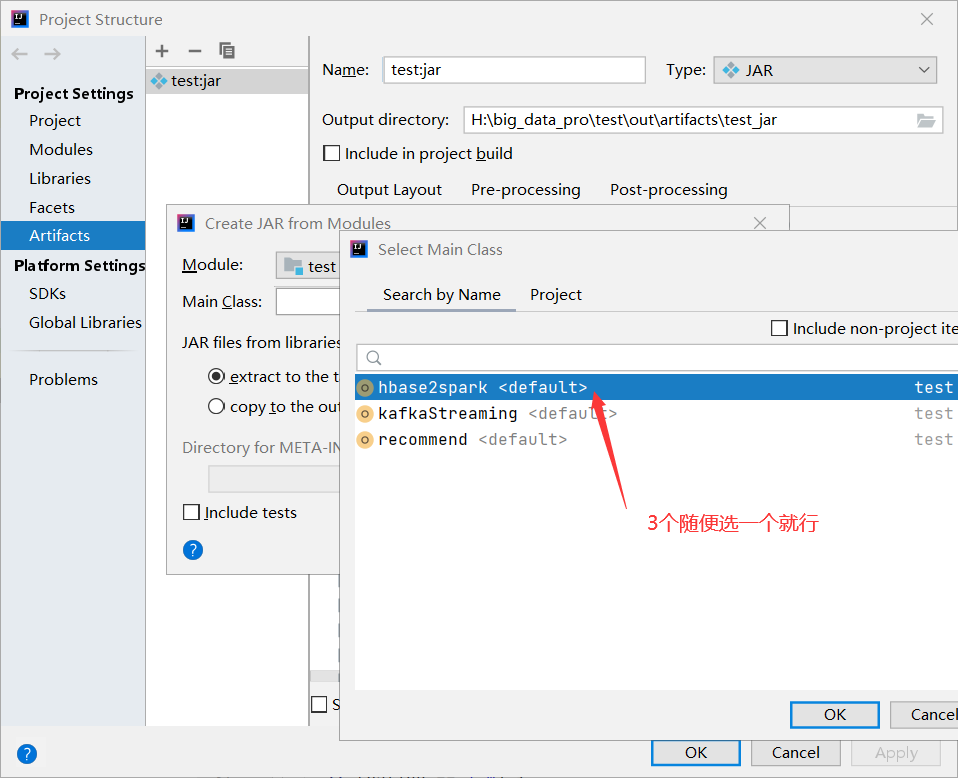
yum –y install python3

**注：后续使用python3解释器需要在命令行使用python3而非python，安装一些库需要用pip3而不是pip**

1. **打jar包**

pom.xml和3个scala代码文件均已提供

打jar包的过程和之前的实验类似，需要注意的是，Main Class随便选一个就可以了



1. **运行简易推荐系统**

**1、代码说明：**

（1）load\_train\_ratings\_hbase.py：将json\_train\_ratings.json作为历史数据直接存储到HBase中

（2）load\_movie\_redis.py：将movies.csv中存储的movieId与movie title的映射关系写入Redis中，用于后续推荐时从movieId转换成movie title

（3）generatorRecord.py：将json\_test\_ratings.json作为实时数据发送到Kafka中

（4）recommend\_server.py与recommend\_client.py：recommend\_server等待recommend\_client连接，连接后recommend\_client可以向recommend\_server发送一个userId，recommend\_server根据推荐算法从Redis中获取召回列表、特征参数与模型参数后，向recommend\_client返回推荐列表。向recommend\_client是一个交互式程序，输入用户id，会输出推荐结果。

（5）hbase2spark.scala：批式计算程序，计算后将结果写入 Redis

（6）kafkaStreaming.scala：流式计算程序，计算后将结果写入 Redis。（同学们也可以自行编写代码，使用Flink实现）

（7）recommend.scala：推荐模型程序，计算后将结果写入 Redis

**2、运行流程：**

**注：所有步骤涉及的命令已在“一些命令.txt”中提供**

1. 启动HDFS
2. 启动zookeeper
3. 启动HBase
4. 配置HBase Thrift连接，以便python中的happybase库能够连接Hbase (详见 一些命令.txt）
5. 在HBase中创建对应的表
6. 启动load\_train\_ratings\_hbase.py（需要运行完）
7. 启动redis
8. 启动load\_movie\_redis.py（需要运行完）
9. 启动Kafka 并创建 Kafka Topic
10. 启动generatorRecord.py（这个程序会一直运行，不需要等待停止）
11. 启动hbase2spark、kafkaStreaming、recommend
12. 启动recommend\_server.py
13. 启动recommend\_client.py

在客户端输入userid，返回结果截图，返回结果可能与图片不一致（40分）

图形用户界面, 文本, 应用程序, 聊天或短信

描述已自动生成