

项目是两步

第一步是 **offline** 建立 数据库 我们用两个 **map reduce** 的 **data pipeline** 来实现。

第二部是 **online** 显示把数据里面数据读出来显示。

1. 获取本机参数，使其能与 MapReduce 进行交互

```
ifconfig | grep inet | grep broadcast #获取本机 IP
```

2. 点击 Open WebStartPage, 进入 XAMP

看 MySQL 运行的端口

进入 XAMP，查看 MySQL 的 root 用户的密码

MySQL

MySQL can be administered with phpMyAdmin.

To connect to the MySQL server from your own scripts use the following connection parameters:

Host	localhost
Port	8889
User	root
Password	root
Socket	/Applications/MAMP/tmp/mysql/mysql.sock

3. 配置 MySQL，（主要目的是为了创建 table，然后可以远程的 map reduce 写 data）

```
打开 Terminal  
cd /Applications/MAMP/Library/bin/
```

(mac) ubuntu 用户在 /usr/local/ammps/sql/bin 这个路径下面

```
sudo ./mysql -uroot -p (Win 的话 mysql -u root -p)
```

(先输入机器密码，然后接下来输入刚才得到的 password root, 两种密码) (ubuntu 密码是 mysql)

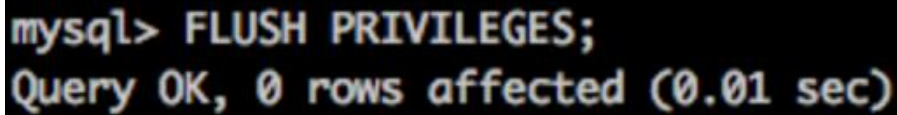
- 4.

```
$ create database test; # 创建 test database  
$ use test; # 切换 database (请理解这句话的意思)
```

```
$ create table output(starting_phrase VARCHAR(250), following_word  
VARCHAR(250), count INT);  
下面这一句 your-password 替换成刚才得到的密码  
$ GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' IDENTIFIED BY '**_your-password_**'  
WITH GRANT OPTION; //enable remote data transfer
```

5. Mysql 的 commands 必须加分号

```
FLUSH PRIVILEGES;
```



```
mysql> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.01 sec)
```

6. 配置 Hadoop

打开 docker

```
./start-container.sh # 启动 docker container
```

```
./start-hadoop.sh # start Hadoop
```

cd src # 进入 Hadoop 里面的文件夹里面

```
wget https://s3-us-west-2.amazonaws.com/jiuzhang-bigdata/mysql-connector-java-5.1.39-  
bin.jar # 下载 mysql-connector 为了远端链接 mysql
```

下一步在 HDFS 里面操作

```
$ hdfs dfs -mkdir /mysql # 在 hdfs 里面创建 mysql 的文件夹, 然后把一些 mysql 相关的东西进去  
  
$ hdfs dfs -put mysql-connector-java-*.jar /mysql/ #hdfs path to mysql-connector*
```

下载代码然后修改代码中的参数, 使其能够与本机 MySQL 通信

```
$ wget https://s3-us-west-2.amazonaws.com/jiuzhang-bigdata/NGram.tar  
$ tar -xvf NGram.tar  
$ cd NGram  
$ hdfs dfs -mkdir -p input
```

```
$ hdfs dfs -rm -r /output
$ hdfs dfs -put bookList/* input/
$ cd src
```

打开在 **Driver.java** 文件中修改以下 4 个参数 特别注意最后一行 **sql connector** 的 **path**:

```
local_ip_address : 192.168.1.5
MySQL_port : 3306
your_password: root
hdfs_path_to_mysql-connector: /mysql/mysql-connector-java-5.1.39-bin.jar

DBConfiguration.configureDB(conf2,
    "com.mysql.jdbc.Driver", // driver class
    "jdbc:mysql://local_ip_address:MySQL_port/test", // db url
    "root", // user name
    "your_password"); //password

job2.addArchiveToClassPath(new Path("hdfs_path_to_mysql-connector"));
```

下面是助教例子

```
DBConfiguration.configureDB(conf2,
    "com.mysql.jdbc.Driver", // driver class
    "jdbc:mysql://192.168.1.5:3306/test", // db url
    "root", // user name
    "root"); //password

job2.addArchiveToClassPath(new Path("/mysql/mysql-connector-java-5.1.39-bin.jar"));
```

7. 运行 Auto-Complete

```
hadoop com.sun.tools.javac.Main *.java
jar cf ngram.jar *.class
hadoop jar ngram.jar Driver input /output 2 3 4
```

请设置 3G 以上内存, 3 CPU 运行此代码(Windows 请至少配置 4G 内存喂 vagrant,

2 表示 ngram_size
3 表示 threshold_size , 当前单词出现的 count<threshold 就会被省略
4 表示 following_word_size

后面三个参数可以随便设置

这个地方会 run 两个 map reduce，如果你只看到一个成功了。第二个没成功，应该是你的 ip 没有配置正确。

8. 查看 MySQL 的 database 你会发现数据已经写到了 mysql 里面啦

用下面语句可以查看

```
select * from output limit 10;
```

看看是否有数据存在 如果没有数据，那就说明没有写到数据里面,检查一下 ip 地址配对没有

对应 Ip 的问题

```
000_0, Status : FAILED
Error: java.io.IOException: Access denied for user 'root'@'172.18.0.4' (password: YES)
        at org.apache.hadoop.mapreduce.lib.db.DBOutputFormat.write(DBOutputFormat.java:185)
```

如果遇到没有权限方位 172.18.0.4 那就要把对应 ip 的加到 root user 里

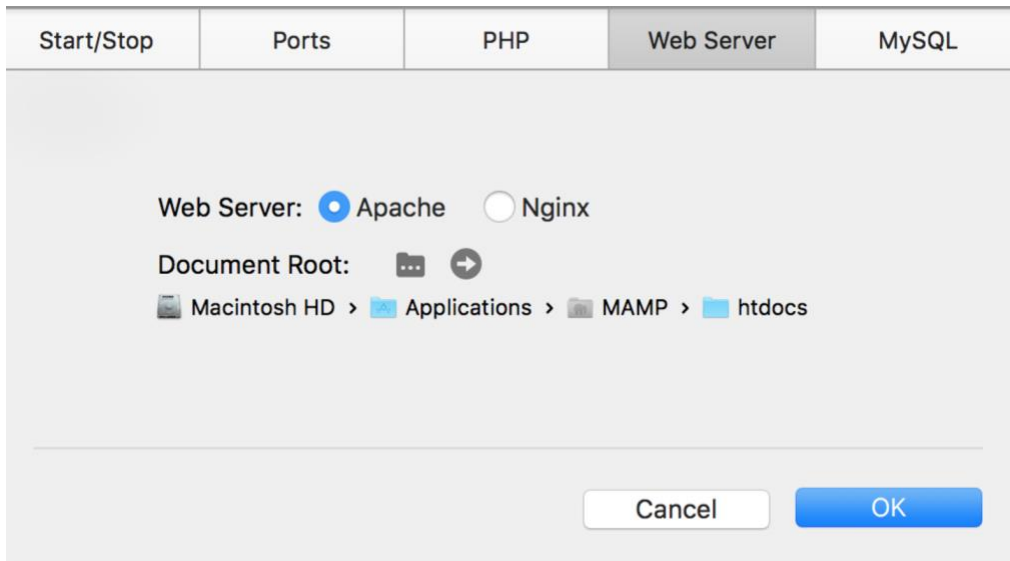
```
$ GRANT ALL ON *.* to 'root'@'192.168.0.103 ' IDENTIFIED BY 'root'; //enable remote data transfer
$ GRANT ALL PRIVILEGES ON *.* TO 'root'@'192.168.0.103 ' WITH GRANT OPTION;
```

9. 运行 Auto-Complete User Interface

```
wget https://s3-us-west-2.amazonaws.com/jiuzhang-bigdata/autocomplete.tar
```

下载相关前端代码包（win 直接点上面的链接用下载工具下载）

```
ubuntu 用户放在下面路径下面/usr/local/ammps/www
mac 用户 打开 XAMP-> preference -> web server
```



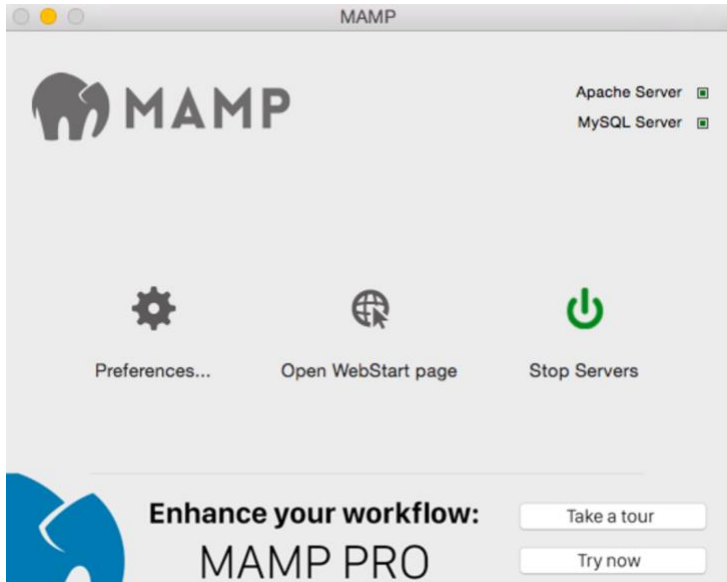
将 Autocomplete 代码包放在 web server 指定路径下 #如图为/htdocs 路径下

将 Autocomplete/ajax_refresh.php 文件一下参数改为本机参数(注意需要指定端口号):

```
<?php
// PDO connect *****
define ('DBUSER', 'root');
define ('DBPASS','root');
define ('DBNAME','test');

function connect() {
    return new PDO('mysql:host=localhost;dbname=test;port=8889', 'root', 'root', array(PDO::ATTR_ERRMODE => PDO::
        ERRMODE_EXCEPTION, PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8"));
}
```

分别改为本机用户名（默认为 root），本机 MySQL 密码
重启 server 然后就可以，点击 stop servers 然后 start servers



进入网页运行 localhost:8888/autocomplete(ubuntu 可以尝试 localhost:80, Windows 用户访问 localhost), 选择 autocomplete, 即可进入输入界面

如果 ubuntu 用户 访问 localhost 或者 ip

进入网页运行 http://localhost/autocomplete/