“大数据工程”课程实验报告

题目：Hive 编程实践	学号姓名：21377061 范春	日期：2024.5.9

实验环境：

（1）虚拟机软件：VMware

（2）Hadoop：3.1.3

（3）JDK：1.8

（4）Hive：3.1.3

（5）编程语言：HiveQL

实验内容与完成情况：

问题 1：

1、安装 Hive3.1.3

```
hadoop@u-virtual-machine:~$ cd /home/hadoop/下载
hadoop@u-virtual-machine:~/下载$ sudo tar -zxvf ./apache-hive-3.1.3-bin.ta
r.gz -C /usr/local
[sudo] hadoop 的密码：
apache-hive-3.1.3-bin/LICENSE
apache-hive-3.1.3-bin/RELEASE_NOTES.txt
apache-hive-3.1.3-bin/NOTICE
apache-hive-3.1.3-bin/binary-package-licenses/com.thoughtworks.paranamer-L
ICENSE
apache-hive-3.1.3-bin/hcatalog/share/webhcat/svr/lib/commons-exec-1.1.jar
apache-hive-3.1.3-bin/hcatalog/share/webhcat/java-client/hive-webhcat-java
-client-3.1.3.jar
hadoop@u-virtual-machine:~/下载$ cd /usr/local/
hadoop@u-virtual-machine:/usr/local$ sudo mv apache-hive-3.1.3-bin hive
hadoop@u-virtual-machine:/usr/local$ sudo chown -R dblab:dblab hive
chown: 无效的用户：“dblab:dblab”
hadoop@u-virtual-machine:/usr/local$ sudo chown -R hadoop:hadoop hive
hadoop@u-virtual-machine:/usr/local$ gedit ~/.bashrc
hadoop@u-virtual-machine:/usr/local$ source ~/.bashrc
hadoop@u-virtual-machine:/usr/local$ cd /usr/local/hive/conf
hadoop@u-virtual-machine:/usr/local/hive/conf$ mv hive-default.xml.templat
e hive-default.xml
hadoop@u-virtual-machine:/usr/local/hive/conf$ cd /usr/local/hive/conf
hadoop@u-virtual-machine:/usr/local/hive/conf$ gedit hive-site.xml
hadoop@u-virtual-machine:/usr/local/hive/conf$ gedit hive-site.xml
hadoop@u-virtual-machine:/usr/local/hive/conf$
```

打开(O)　*.bashrc　保存(S)

```
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
export HIVE_HOME=/usr/local/hive
export PATH=$PATH:$HIVE_HOME/bin
export HADOOP_HOME=/usr/local/hadoop
export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_162
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:/usr/local/hbase/bin:$PATH
case $- in
    *i*) ;;
      *) return;;
esac

# don't put duplicate lines or lines starting with space in the
```

打开(O) hive-site.xml /usr/local/hive/conf 保存(S)

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>javax.jdo.option.ConnectionURL</name>
    <value>jdbc:mysql://localhost:3306/hive?-
createDatabaseIfNotExist=true&amp;useSSL=false</value>
    <description>JDBC connect string for a JDBC metastore</description>
  </property>
  <property>
    <name>javax.jdo.option.ConnectionDriverName</name>
    <value>com.mysql.jdbc.Driver</value>
    <description>Driver class name for a JDBC metastore</description>
  </property>
  <property>
    <name>javax.jdo.option.ConnectionUserName</name>
    <value>hive</value>
    <description>username to use against metastore database</-
description>
  </property>
  <property>
    <name>javax.jdo.option.ConnectionPassword</name>
    <value>hive</value>
    <description>password to use against metastore database</-
description>
  </property>
</configuration>
```

2、安装并配制 MySQL

```
hadoop@u-virtual-machine:~$ cd /usr/local
hadoop@u-virtual-machine:/usr/local$ sudo apt-get update
[sudo] hadoop 的密码:
命中:1 http://mirrors.aliyun.com/ubuntu focal InRelease
命中:2 http://mirrors.aliyun.com/ubuntu focal-updates InRelease
命中:3 http://mirrors.aliyun.com/ubuntu focal-backports InRelease
命中:4 http://mirrors.aliyun.com/ubuntu focal-security InRelease
正在读取软件包列表... 完成
hadoop@u-virtual-machine:/usr/local$ sudo apt-get install mysql-server
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
将会同时安装下列软件:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7
  libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2
  mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
```

```
hadoop@u-virtual-machine:/usr/local$ service mysql start
hadoop@u-virtual-machine:/usr/local$ sudo netstat -tap | grep mysql
tcp        0      0 localhost:mysql         0.0.0.0:*               LISTEN
      3108/mysqld
tcp        0      0 localhost:33060         0.0.0.0:*               LISTEN
      3108/mysqld
hadoop@u-virtual-machine:/usr/local$
```

```
hadoop@u-virtual-machine:/usr/local$ service mysql stop
hadoop@u-virtual-machine:/usr/local$ cd /home/hadoop/桌面
hadoop@u-virtual-machine:~/桌面$ sudo tar -zxvf mysql-connector-java-5.1.4
0.tar.gz
mysql-connector-java-5.1.40/
mysql-connector-java-5.1.40/docs/
```

```
hadoop@u-virtual-machine:~/桌面$ cp mysql-connector-java-5.1.40/mysql-conn
ector-java-5.1.40-bin.jar  /usr/local/hive/lib
```

```
mysql> create database hive;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on *.* to hive@localhost identified by 'hive';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual
 that corresponds to your MySQL server version for the right syntax to use
 near 'identified by 'hive'' at line 1
mysql>
mysql> CREATE USER 'hive'@'localhost' IDENTIFIED BY 'hive';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'hive'@'localhost' WITH GRANT OPTION
;
Query OK, 0 rows affected (0.00 sec)

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

mysql>
```

```
hadoop@u-virtual-machine:/usr/local$ cd /usr/local/hive
hadoop@u-virtual-machine:/usr/local/hive$ ./bin/schematool -initSchema -db
Type mysql
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/log4j-slf4j-impl-2.1
7.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/li
b/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
```

```
hive> create database if not exists hive;
OK
Time taken: 0.515 seconds
hive> show databases;
OK
default
hive
Time taken: 0.132 seconds, Fetched: 2 row(s)
hive>
```

问题 2：

1、将数据上传到 HDFS

```
hadoop@u-virtual-machine:/usr/local/hadoop$ ./bin/hdfs dfs -ls week11
Found 2 items
-rw-r--r--   1 hadoop supergroup 1801526075 2024-05-09 23:47 week11/googlebooks-eng-all-1gram-20120701-a
-rw-r--r--   1 hadoop supergroup 1268392934 2024-05-09 23:43 week11/googlebooks-eng-all-1gram-20120701-b
hadoop@u-virtual-machine:/usr/local/hadoop$ ./bin/hdfs dfs -cat week11/googlebooks-eng-all-1gram-20120701-a | head -n 10
2024-05-09 23:58:05,290 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
A'Aang_NOUN     1879    45      5
A'Aang_NOUN     1882    5       4
A'Aang_NOUN     1885    1       1
A'Aang_NOUN     1891    1       1
A'Aang_NOUN     1899    20      4
A'Aang_NOUN     1927    3       1
A'Aang_NOUN     1959    5       2
A'Aang_NOUN     1962    2       2
A'Aang_NOUN     1963    1       1
A'Aang_NOUN     1966    45      13
cat: Unable to write to output stream.
```

2、在 hive 数据库中创建表 word_counts，并将 HDFS 中的两个数据文件合并存储到该表中。

```
hive> use hive;
OK
Time taken: 0.028 seconds
hive> CREATE TABLE word_counts (
    >     word STRING,
    >     year INT,
    >     occurrence_count INT,
    >     book_count INT
    > )
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t'
    > STORED AS TEXTFILE;
OK
Time taken: 0.723 seconds
hive> LOAD DATA INPATH '/user/hadoop/week11/googlebooks-eng-all-1gram-20120701-a' INTO TABLE word_counts;
Loading data to table hive.word_counts
OK
Time taken: 0.594 seconds
hive> LOAD DATA INPATH '/user/hadoop/week11/googlebooks-eng-all-1gram-20120701-b' INTO TABLE word_counts;
Loading data to table hive.word_counts
OK
Time taken: 0.243 seconds
hive> SELECT * FROM word_counts LIMIT 10;
OK
A'Aang_NOUN     1879    45      5
A'Aang_NOUN     1882    5       4
A'Aang_NOUN     1885    1       1
A'Aang_NOUN     1891    1       1
A'Aang_NOUN     1899    20      4
A'Aang_NOUN     1927    3       1
A'Aang_NOUN     1959    5       2
A'Aang_NOUN     1962    2       2
A'Aang_NOUN     1963    1       1
A'Aang_NOUN     1966    45      13
Time taken: 1.457 seconds, Fetched: 10 row(s)
```

问题 3：对于每个独特的 bigram，计算其每年出现的平均次数，并将结果保存到表 word_averages 中。

```
hive> CREATE TABLE word_averages AS
    > SELECT
    >     word,
    >     SUM(occurrence_count) AS total_occurrence_count,
    >     COUNT(*) AS count_of_occurrences,
    >     SUM(occurrence_count) / CAST(COUNT(*) AS DOUBLE) AS avg_occurrence_per_record
    > FROM
    >     word_counts
    > GROUP BY
    >     word;
Query ID = hadoop_20240510095125_f784d5be-d096-412a-bcc3-db7885f8b003
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Job running in-process (local Hadoop)
2024-05-10 09:51:28,407 Stage-1 map = 0%,  reduce = 0%
2024-05-10 09:51:29,420 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local573197381_0001
Moving data to directory hdfs://localhost:9000/user/hive/warehouse/hive.db/word_averages
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 48942 HDFS Write: 864 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 4.007 seconds
hive>
```

问题 4：将每年平均出现次数最高的 20 个 bigram（按平均值降序排列）保存在表 top_20_word_averages 中并输出。

```
hive> CREATE TABLE top_20_word_averages AS
    > select
    > word,
    > avg_occurrence_per_record
    > from word_averages
    > order by avg_occurrence_per_record
    > desc limit 20;
Query ID = hadoop_20240510095622_29a41c1d-773c-4301-99dc-b89784ef576c
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Job running in-process (local Hadoop)
2024-05-10 09:56:23,465 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1559666484_0002
Moving data to directory hdfs://localhost:9000/user/hive/warehouse/hive.db/top_20_word_averages
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 50670 HDFS Write: 2318 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 1.578 seconds
```

```
hive> select * from top_20_word_averages
    > ;
OK
A.N._NOUN       1426.659217877095
A.J.B._NOUN     79.03157894736842
A'Aang_NOUN     9.5625
A.Phoenix_NOUN  7.5
A.Briggs_NOUN   6.512820512820513
A.D.A.A.        5.787878787878788
A.E.U._DET      5.72
A.K.K.  5.327586206896552
A.C.M.S._NOUN   5.2926829268292686
A.5.3_DET       5.225806451612903
A.L.I.V.E.      4.55
A.N.Kolmogorov_NOUN     4.033333333333333
A.M.Inst.N.A    4.0
A.M.C._VERB     3.933333333333333
A.IR._NOUN      3.9318181818181817
A.R.R   3.8372093023255816
A.165   3.076923076923077
A.J.U.  2.911764705882353
A'que_ADJ       2.891304347826087
A.C.I.I_NOUN    2.75
Time taken: 0.107 seconds, Fetched: 20 row(s)
```

出现的问题：

最初尝试对所有数据进行计算，但出现了内存相关的错误导致无法实现，最后利用如下代码实现了随机选取一小部分数据进行运算。

```
CREATE TEMPORARY TABLE word_counts_sample
AS
SELECT *
FROM word_counts TABLESAMPLE(BUCKET 1 OUT OF 10000 ON RAND())
s;
```