IERG 4330/ESTR 4316/IEMS 5730 Spring 2022 Homework 2

Release date: Feb 20, 2022

Due date: Mar 8, 2022 (Tuesday) 11:59:00 pm

We will discuss the solution soon after the deadline. No late homework will be accepted!

Every Student MUST include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on Elearning system is original except for source material explicitly acknowledged, and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website http://www.cuhk.edu.hk/policy/academichonesty/.

Signed (Student Jesse) Date: 8 - 3 - 22

Name Chan Kai Yin SID 1155124983

Submission notice:

- Submit your homework via the elearning system.
- All students are required to submit this assignment.

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

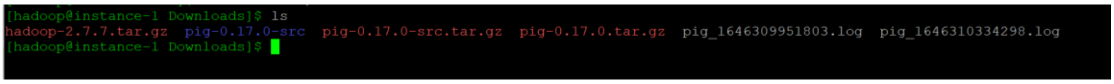
Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value, and justify any assumptions you make. You will be graded not only on whether your answer is correct, but also on whether you have done an intelligent analysis.

Q1:

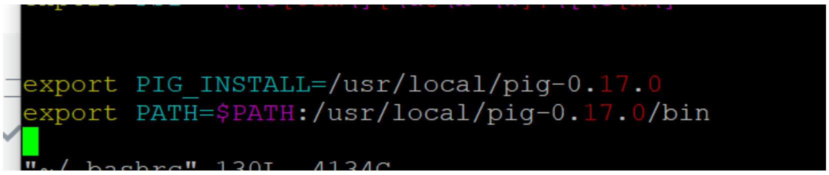
a.)

Download and install the Pig with version 0.17.0:



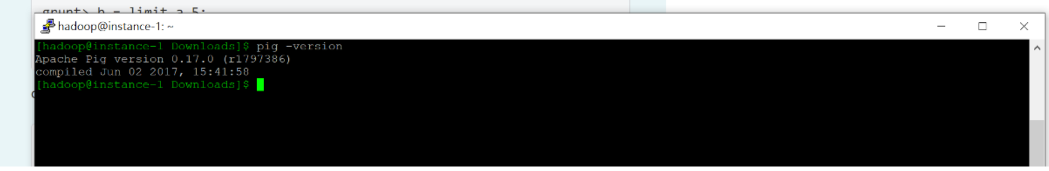
```
[hadoop@instance-1 Downloads]$ ls
hadoop-2.7.7.tar.gz  pig-0.17.0-src  pig-0.17.0-src.tar.gz  pig-0.17.0.tar.gz  pig_1646309951803.log  pig_1646310334298.log
[hadoop@instance-1 Downloads]$
```

modify the environment variable for Pig:



```
export PIG_INSTALL=/usr/local/pig-0.17.0
export PATH=$PATH:/usr/local/pig-0.17.0/bin

"~/.bashrc" 130L, 4134C
```

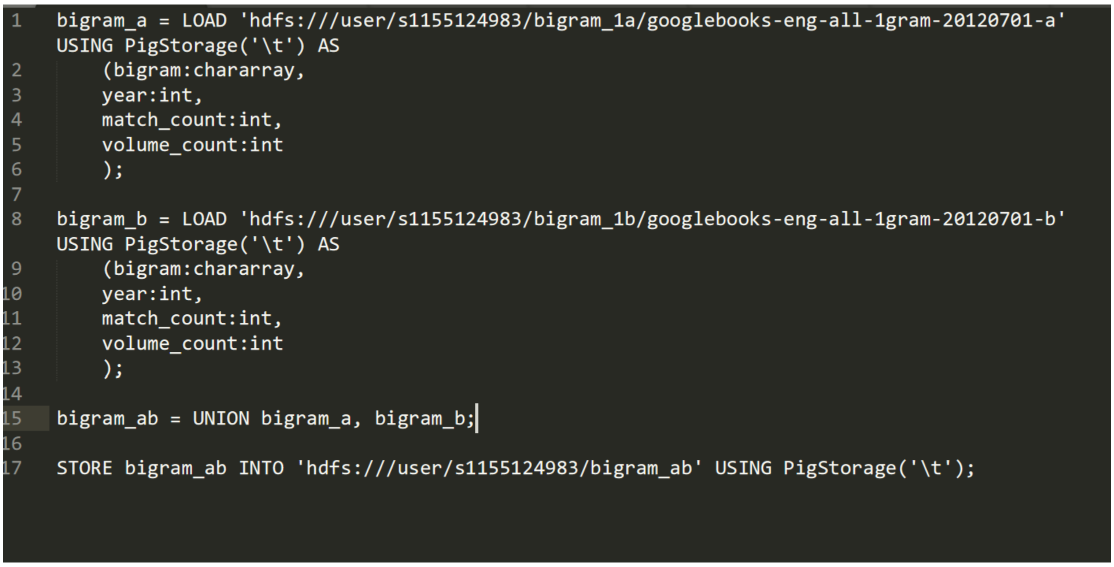
Check Pig version:



```
[hadoop@instance-1 Downloads]$ pig -version
Apache Pig version 0.17.0 (r1797386)
compiled Jun 02 2017, 15:41:58
[hadoop@instance-1 Downloads]$
```

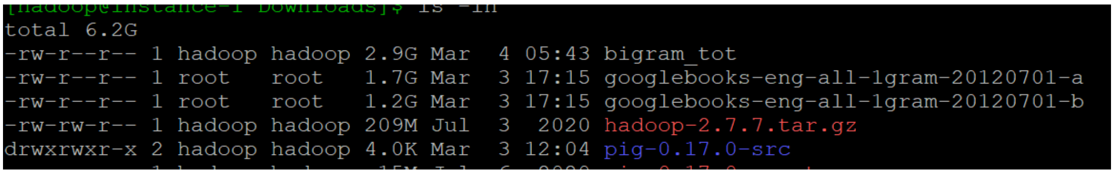
b.)

Code:



```
bigram_a = LOAD 'hdfs:///user/s1155124983/bigram_1a/googlebooks-eng-all-1gram-20120701-a' USING PigStorage('\t') AS
    (bigram:chararray,
    year:int,
    match_count:int,
    volume_count:int
    );

bigram_b = LOAD 'hdfs:///user/s1155124983/bigram_1b/googlebooks-eng-all-1gram-20120701-b' USING PigStorage('\t') AS
    (bigram:chararray,
    year:int,
    match_count:int,
    volume_count:int
    );

bigram_ab = UNION bigram_a, bigram_b;

STORE bigram_ab INTO 'hdfs:///user/s1155124983/bigram_ab' USING PigStorage('\t');
```

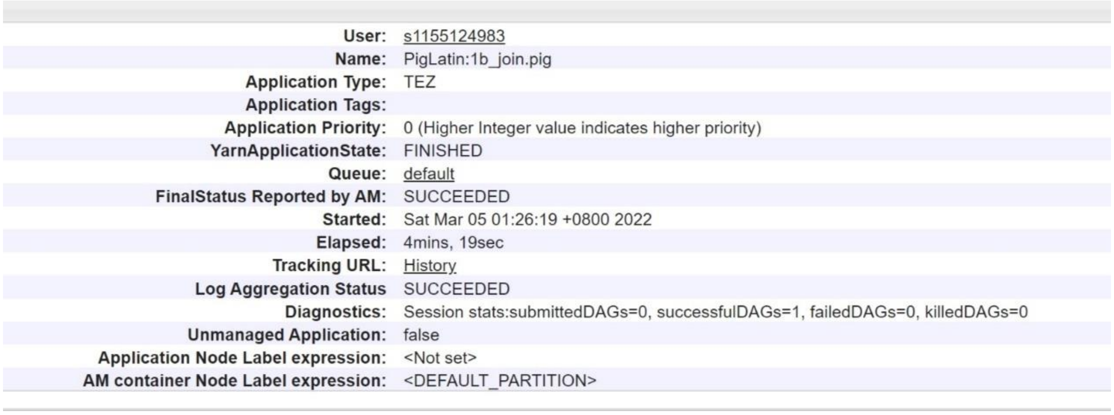
Output file:



```
[hadoop@instance-1 Downloads]$ ls -lh
total 6.2G
-rw-r--r-- 1 hadoop hadoop 2.9G Mar  4 05:43 bigram_tot
-rw-r--r-- 1 root   root   1.7G Mar  3 17:15 googlebooks-eng-all-1gram-20120701-a
-rw-r--r-- 1 root   root   1.2G Mar  3 17:15 googlebooks-eng-all-1gram-20120701-b
-rw-rw-r-- 1 hadoop hadoop 209M Jul  3  2020 hadoop-2.7.7.tar.gz
drwxrwxr-x 2 hadoop hadoop 4.0K Mar  3 12:04 pig-0.17.0-src
```

Merging parts of file and the final joined file named bigram_tot. From the size of a and b file, we can verify it success joined.

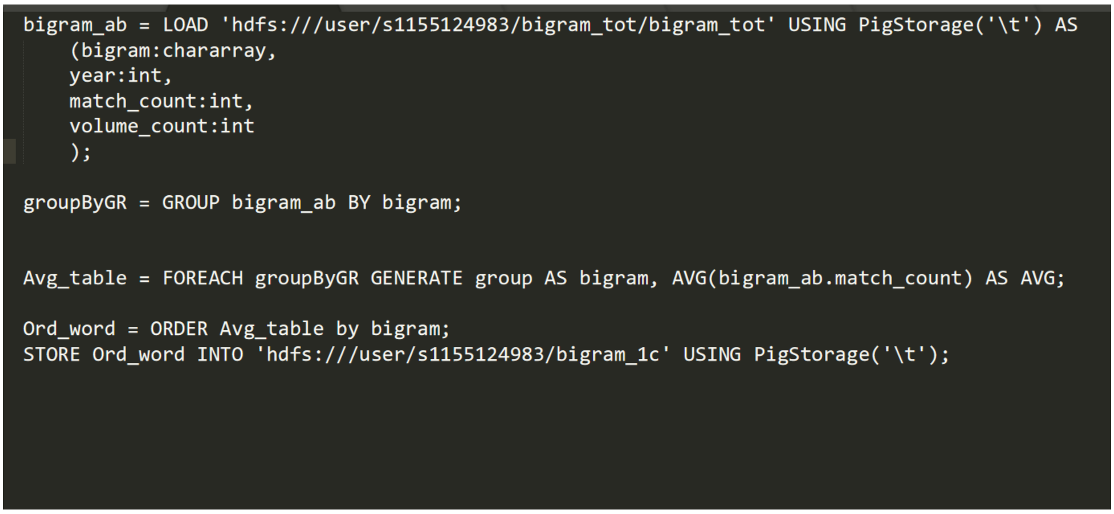
Time: Pig script completed in 4 minutes, 17 seconds and 432 milliseconds (257432 ms)



User:	s1155124983
Name:	PigLatin:1b_join.pig
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sat Mar 05 01:26:19 +0800 2022
Elapsed:	4mins, 19sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=1, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

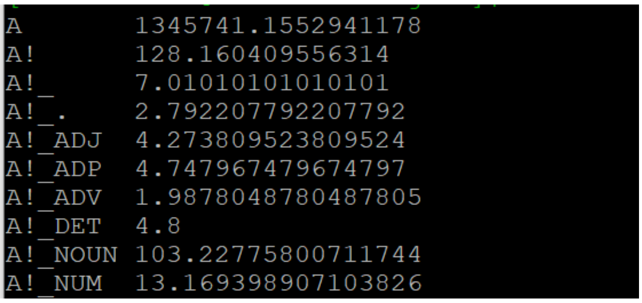
c.)

The code:



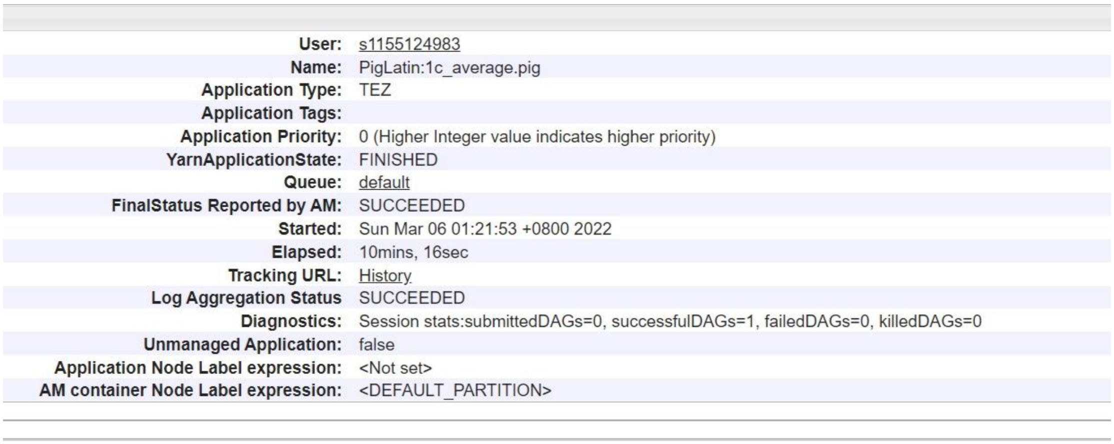
```
bigram_ab = LOAD 'hdfs:///user/s1155124983/bigram_tot/bigram_tot' USING PigStorage('\t') AS
    (bigram:chararray,
    year:int,
    match_count:int,
    volume_count:int
    );

groupByGR = GROUP bigram_ab BY bigram;


Avg_table = FOREACH groupByGR GENERATE group AS bigram, AVG(bigram_ab.match_count) AS AVG;

Ord_word = ORDER Avg_table by bigram;
STORE Ord_word INTO 'hdfs:///user/s1155124983/bigram_1c' USING PigStorage('\t');
```

Output:



```
A        1345741.1552941178
A!       128.160409556314
A!_      7.01010101010101
A!_.     2.792207792207792
A!_ADJ   4.273809523809524
A!_ADP   4.747967479674797
A!_ADV   1.9878048780487805
A!_DET   4.8
A!_NOUN  103.22775800711744
A!_NUM   13.169398907103826
```

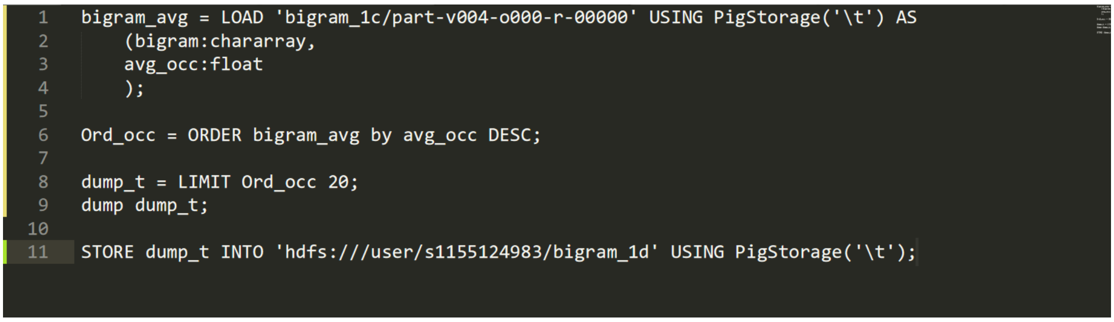
Time: 10mins



User:	s1155124983
Name:	PigLatin:1c_average.pig
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sun Mar 06 01:21:53 +0800 2022
Elapsed:	10mins, 16sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=1, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

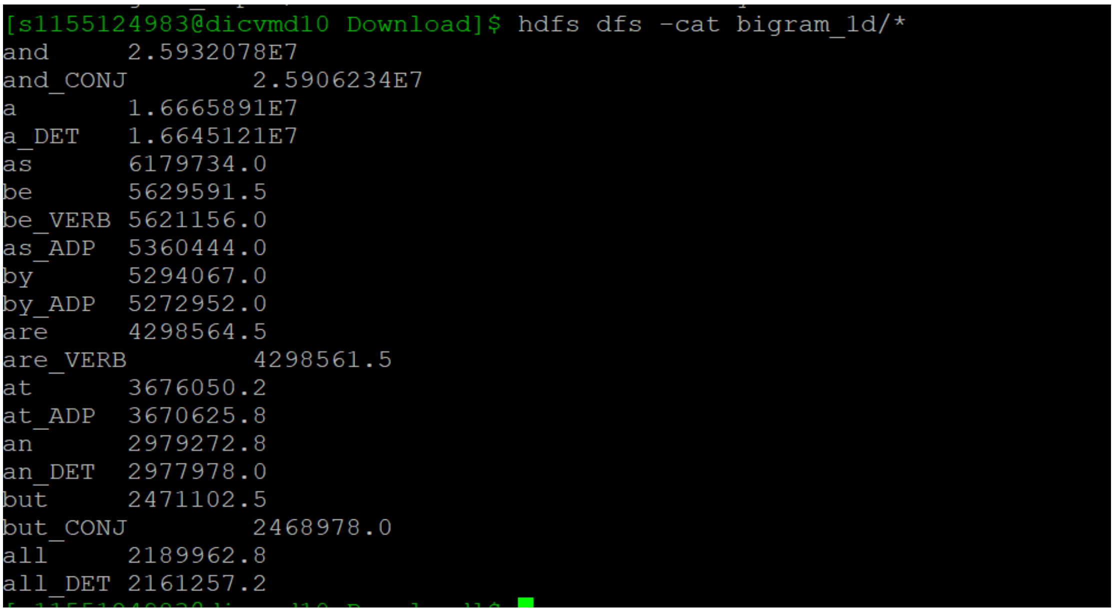
d.)

Code:



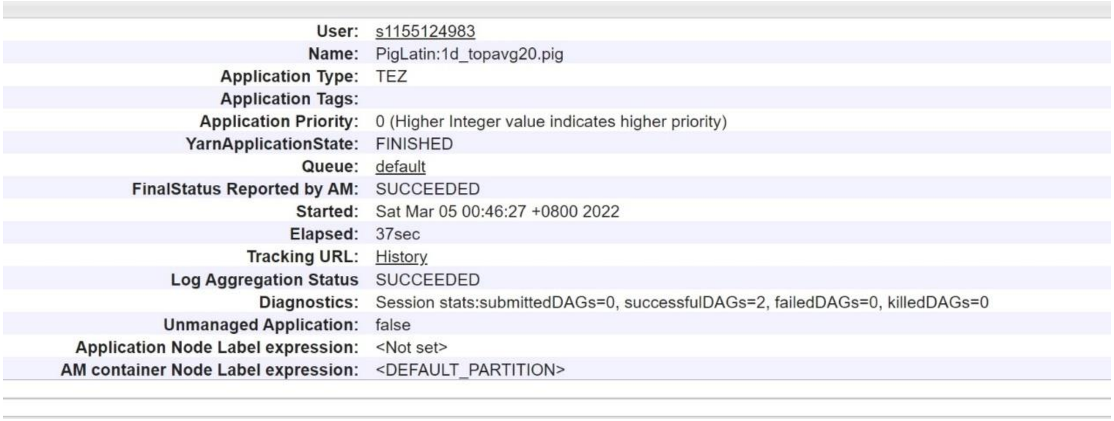
```
bigram_avg = LOAD 'bigram_1c/part-v004-o000-r-00000' USING PigStorage('\t') AS
    (bigram:chararray,
    avg_occ:float
    );

Ord_occ = ORDER bigram_avg by avg_occ DESC;

dump_t = LIMIT Ord_occ 20;
dump dump_t;

STORE dump_t INTO 'hdfs:///user/s1155124983/bigram_1d' USING PigStorage('\t');
```

Output:



```
[s1155124983@dicvmd10 Download]$ hdfs dfs -cat bigram_1d/*
and     2.5932078E7
and_CONJ        2.5906234E7
a       1.6665891E7
a_DET   1.6645121E7
as      6179734.0
be      5629591.5
be_VERB 5621156.0
as_ADP  5360444.0
by      5294067.0
by_ADP  5272952.0
are     4298564.5
are_VERB        4298561.5
at      3676050.2
at_ADP  3670625.8
an      2979272.8
an_DET  2977978.0
but     2471102.5
but_CONJ        2468978.0
all     2189962.8
all_DET 2161257.2
[s1155124983@dicvmd10 Download]$
```

Time: 37s

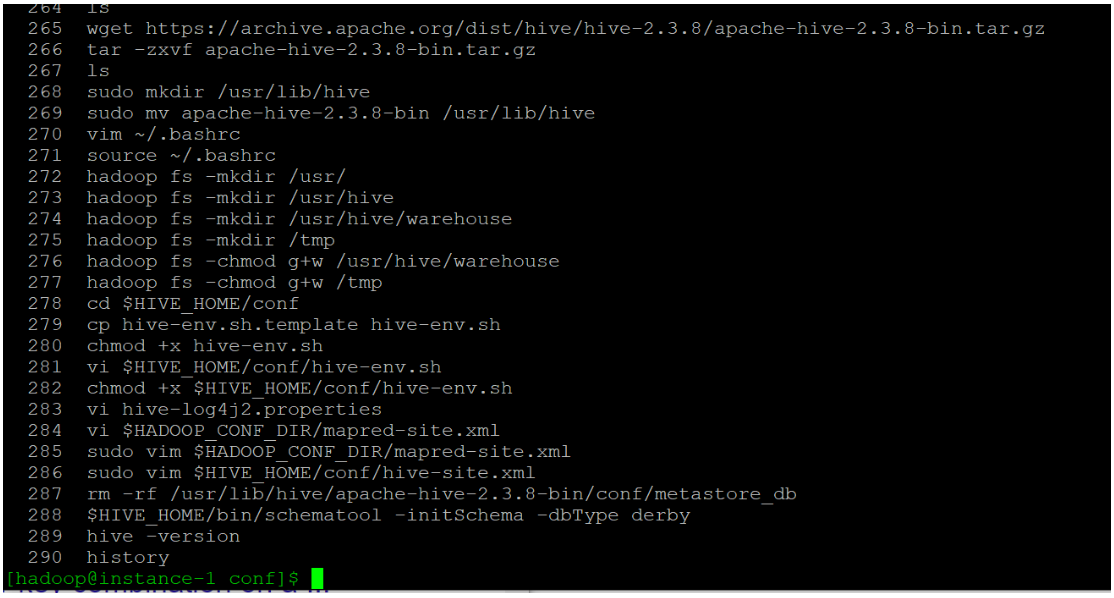


User:	s1155124983
Name:	PigLatin:1d_topavg20.pig
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sat Mar 05 00:46:27 +0800 2022
Elapsed:	37sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=2, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

Q2

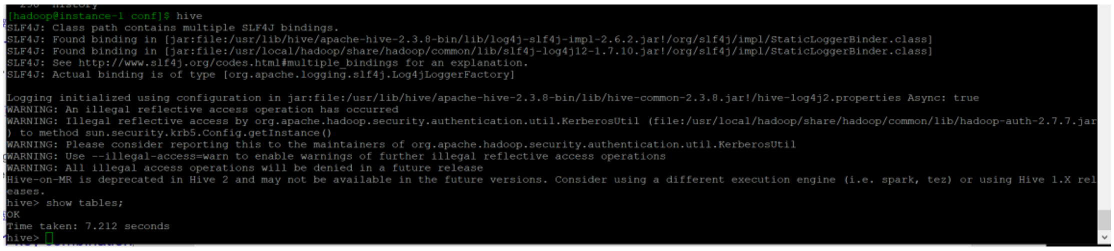
a.)

Installation commands:



```
  264  ls
  265  wget https://archive.apache.org/dist/hive/hive-2.3.8/apache-hive-2.3.8-bin.tar.gz
  266  tar -zxvf apache-hive-2.3.8-bin.tar.gz
  267  ls
  268  sudo mkdir /usr/lib/hive
  269  sudo mv apache-hive-2.3.8-bin /usr/lib/hive
  270  vim ~/.bashrc
  271  source ~/.bashrc
  272  hadoop fs -mkdir /usr/
  273  hadoop fs -mkdir /usr/hive
  274  hadoop fs -mkdir /usr/hive/warehouse
  275  hadoop fs -mkdir /tmp
  276  hadoop fs -chmod g+w /usr/hive/warehouse
  277  hadoop fs -chmod g+w /tmp
  278  cd $HIVE_HOME/conf
  279  cp hive-env.sh.template hive-env.sh
  280  chmod +x hive-env.sh
  281  vi $HIVE_HOME/conf/hive-env.sh
  282  chmod +x $HIVE_HOME/conf/hive-env.sh
  283  vi hive-log4j2.properties
  284  vi $HADOOP_CONF_DIR/mapred-site.xml
  285  sudo vim $HADOOP_CONF_DIR/mapred-site.xml
  286  sudo vim $HIVE_HOME/conf/hive-site.xml
  287  rm -rf /usr/lib/hive/apache-hive-2.3.8-bin/conf/metastore_db
  288  $HIVE_HOME/bin/schematool -initSchema -dbType derby
  289  hive -version
  290  history
[hadoop@instance-1 conf]$ 
```

Testing the hive:

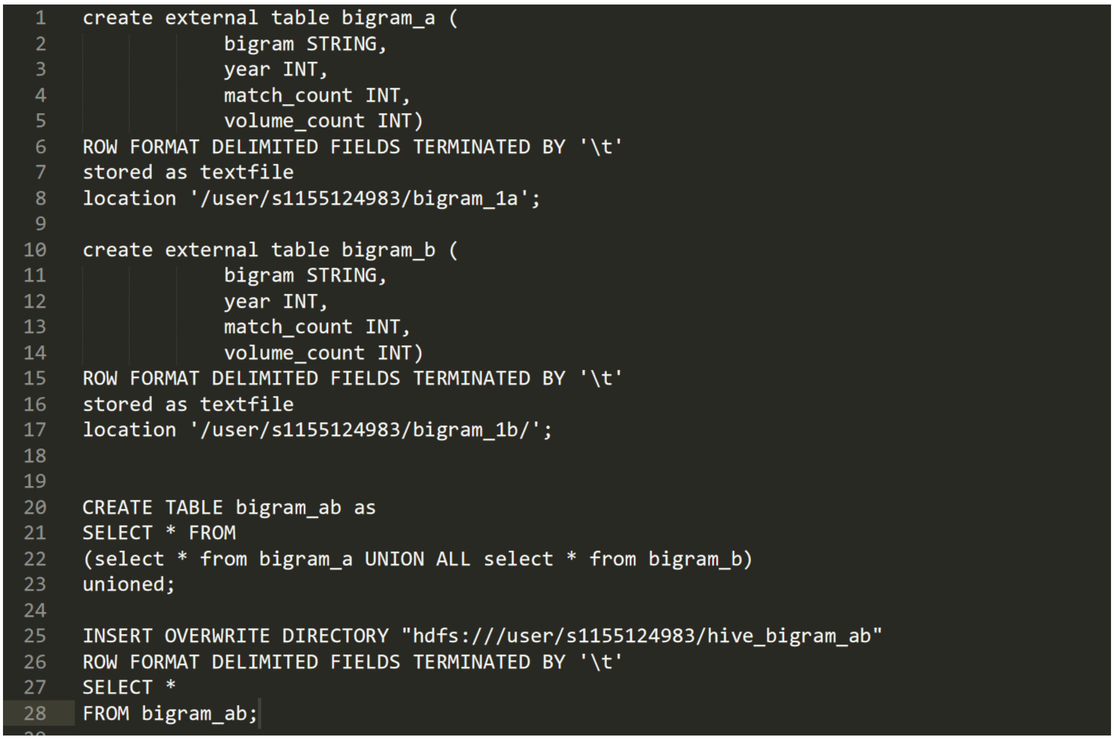


```
  290  history
[hadoop@instance-1 conf]$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hive/apache-hive-2.3.8-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/usr/lib/hive/apache-hive-2.3.8-bin/lib/hive-common-2.3.8.jar!/hive-log4j2.properties Async: true
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.7.7.jar
) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X rel
eases.
hive> show tables;
OK
Time taken: 7.212 seconds
hive> 
```

b.)

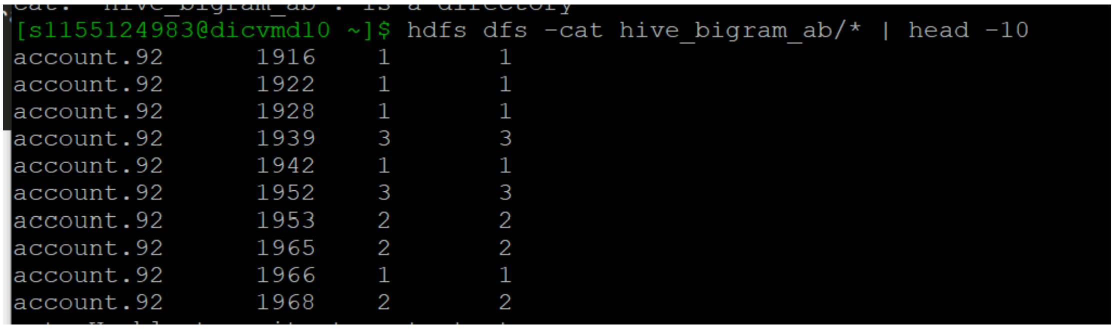
Redoing Q1b - join table:

Code:



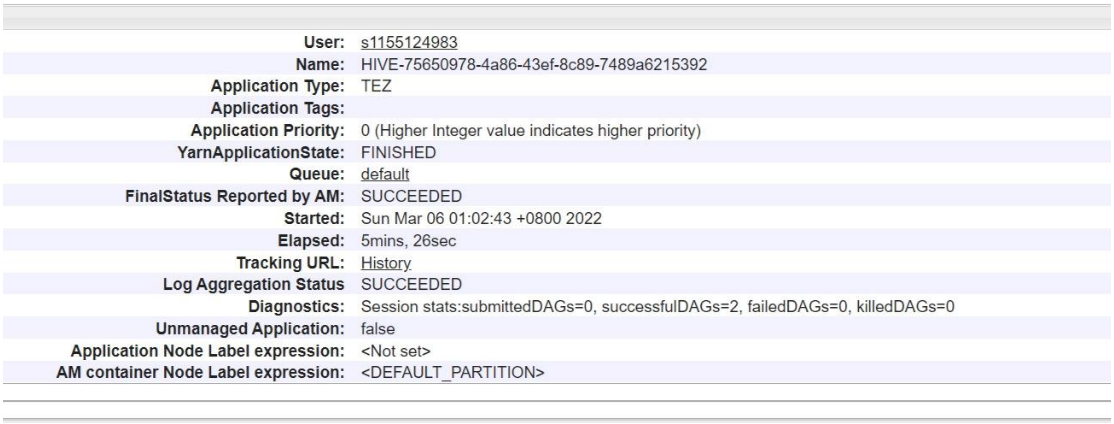
```
create external table bigram_a (
            bigram STRING,
            year INT,
            match_count INT,
            volume_count INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
stored as textfile
location '/user/s1155124983/bigram_1a';

create external table bigram_b (
            bigram STRING,
            year INT,
            match_count INT,
            volume_count INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
stored as textfile
location '/user/s1155124983/bigram_1b/';


CREATE TABLE bigram_ab as
SELECT * FROM
(select * from bigram_a UNION ALL select * from bigram_b)
unioned;

INSERT OVERWRITE DIRECTORY "hdfs:///user/s1155124983/hive_bigram_ab"
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
SELECT *
FROM bigram_ab;
```

Output:



```
cat: hive_bigram_ab: is a directory
[s1155124983@dicvmd10 ~]$ hdfs dfs -cat hive_bigram_ab/* | head -10
account.92      1916    1       1
account.92      1922    1       1
account.92      1928    1       1
account.92      1939    3       3
account.92      1942    1       1
account.92      1952    3       3
account.92      1953    2       2
account.92      1965    2       2
account.92      1966    1       1
account.92      1968    2       2
```

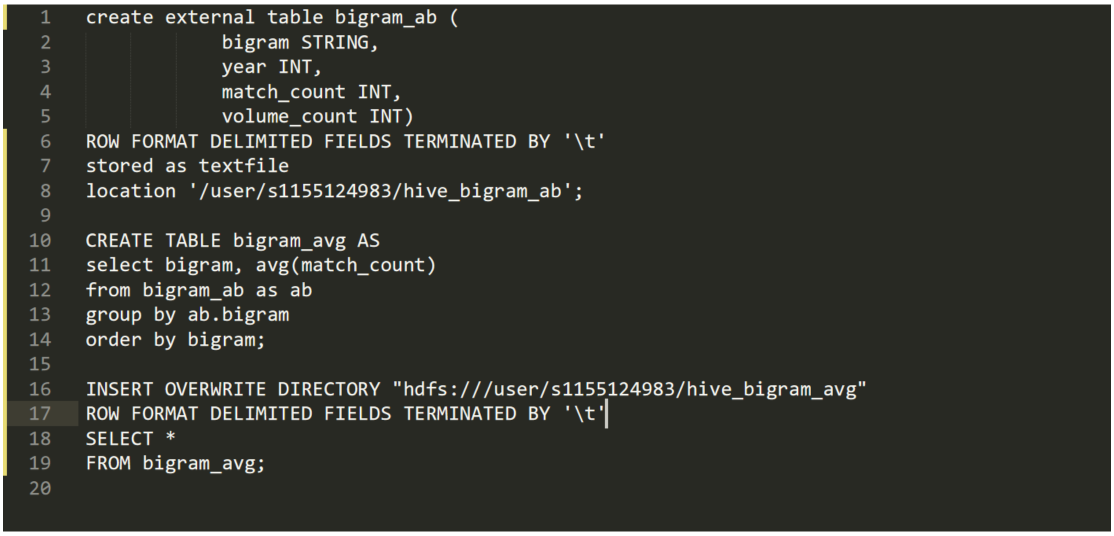
Time: 5 minutes



User:	s1155124983
Name:	HIVE-75650978-4a86-43ef-8c89-7489a6215392
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sun Mar 06 01:02:43 +0800 2022
Elapsed:	5mins, 26sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=2, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

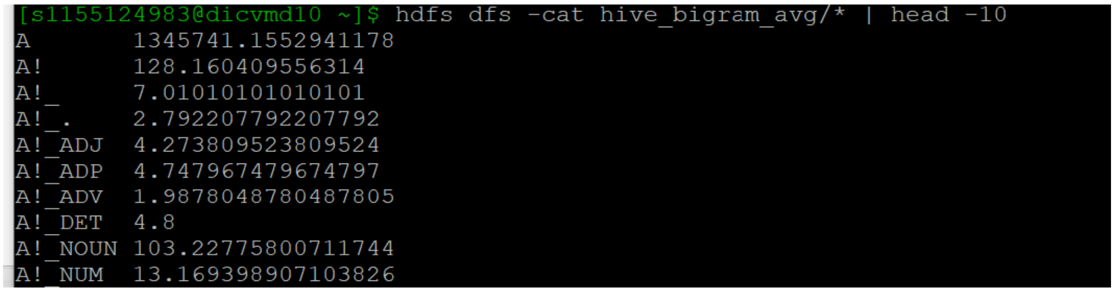
Redoing Q1c – calculate average:

Code:



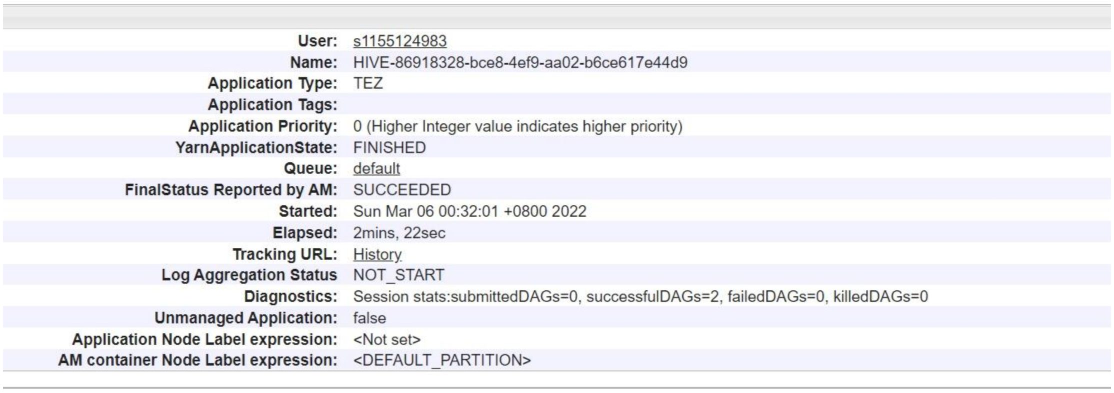
```
create external table bigram_ab (
            bigram STRING,
            year INT,
            match_count INT,
            volume_count INT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
stored as textfile
location '/user/s1155124983/hive_bigram_ab';

CREATE TABLE bigram_avg AS
select bigram, avg(match_count)
from bigram_ab as ab
group by ab.bigram
order by bigram;

INSERT OVERWRITE DIRECTORY "hdfs:///user/s1155124983/hive_bigram_avg"
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
SELECT *
FROM bigram_avg;

```

Output:



```
[s1155124983@dicvmd10 ~]$ hdfs dfs -cat hive_bigram_avg/* | head -10
A       1345741.1552941178
A!      128.160409556314
A!_     7.01010101010101
A!_.    2.792207792207792
A!_ADJ  4.273809523809524
A!_ADP  4.747967479674797
A!_ADV  1.9878048780487805
A!_DET  4.8
A!_NOUN 103.22775800711744
A!_NUM  13.169398907103826
```

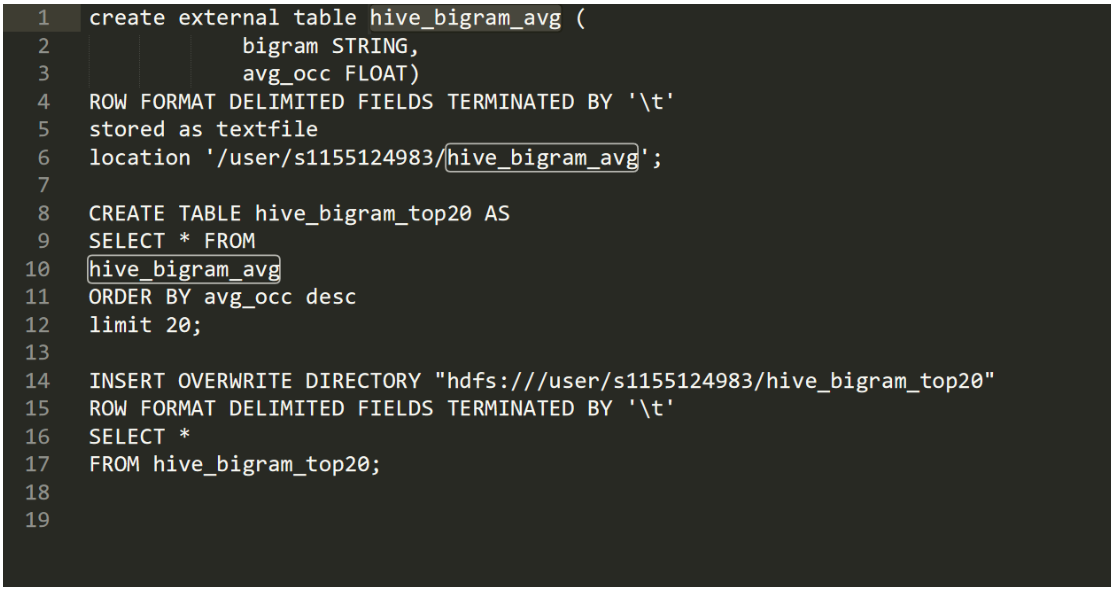
Time: 2 minutes



User:	s1155124983
Name:	HIVE-86918328-bce8-4ef9-aa02-b6ce617e44d9
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sun Mar 06 00:32:01 +0800 2022
Elapsed:	2mins, 22sec
Tracking URL:	History
Log Aggregation Status	NOT_START
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=2, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

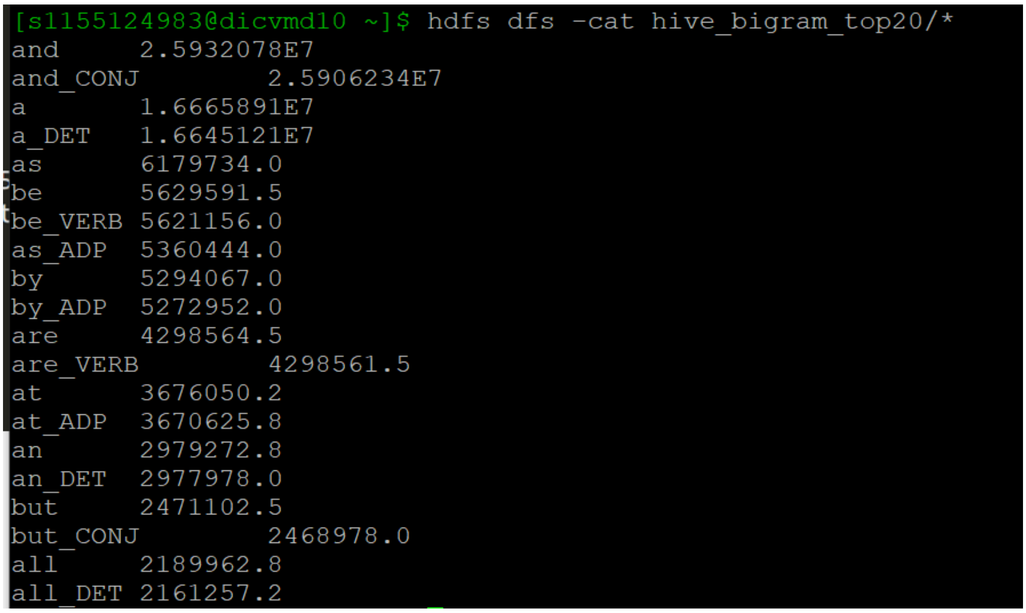
Redoing Q1d – top 20 average :

Code:



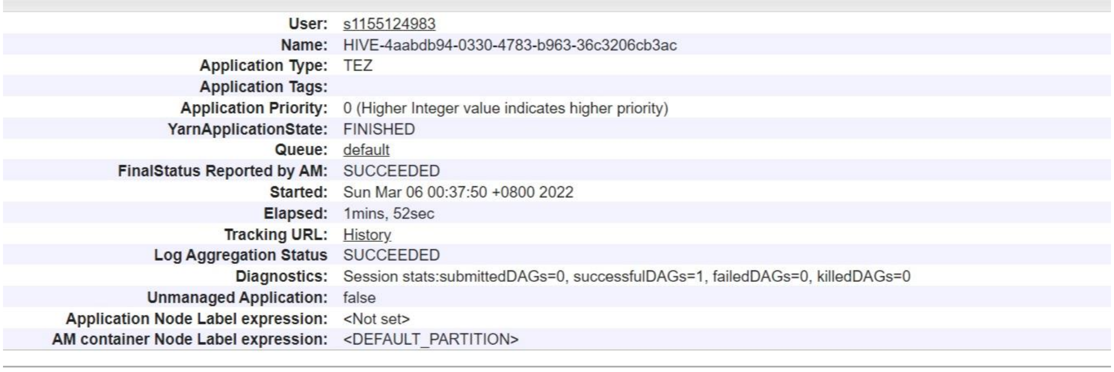
```
create external table hive_bigram_avg (
            bigram STRING,
            avg_occ FLOAT)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
stored as textfile
location '/user/s1155124983/hive_bigram_avg';

CREATE TABLE hive_bigram_top20 AS
SELECT * FROM
hive_bigram_avg
ORDER BY avg_occ desc
limit 20;

INSERT OVERWRITE DIRECTORY "hdfs:///user/s1155124983/hive_bigram_top20"
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
SELECT *
FROM hive_bigram_top20;


```

Output:



```
[s1155124983@dicvmd10 ~]$ hdfs dfs -cat hive_bigram_top20/*
and     2.5932078E7
and_CONJ        2.5906234E7
a       1.6665891E7
a_DET   1.6645121E7
as      6179734.0
be      5629591.5
be_VERB 5621156.0
as_ADP  5360444.0
by      5294067.0
by_ADP  5272952.0
are     4298564.5
are_VERB        4298561.5
at      3676050.2
at_ADP  3670625.8
an      2979272.8
an_DET  2977978.0
but     2471102.5
but_CONJ        2468978.0
all     2189962.8
all_DET 2161257.2
```

Time: 1 minutes



User:	s1155124983
Name:	HIVE-4aabdb94-0330-4783-b963-36c3206cb3ac
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Sun Mar 06 00:37:50 +0800 2022
Elapsed:	1mins, 52sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=1, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

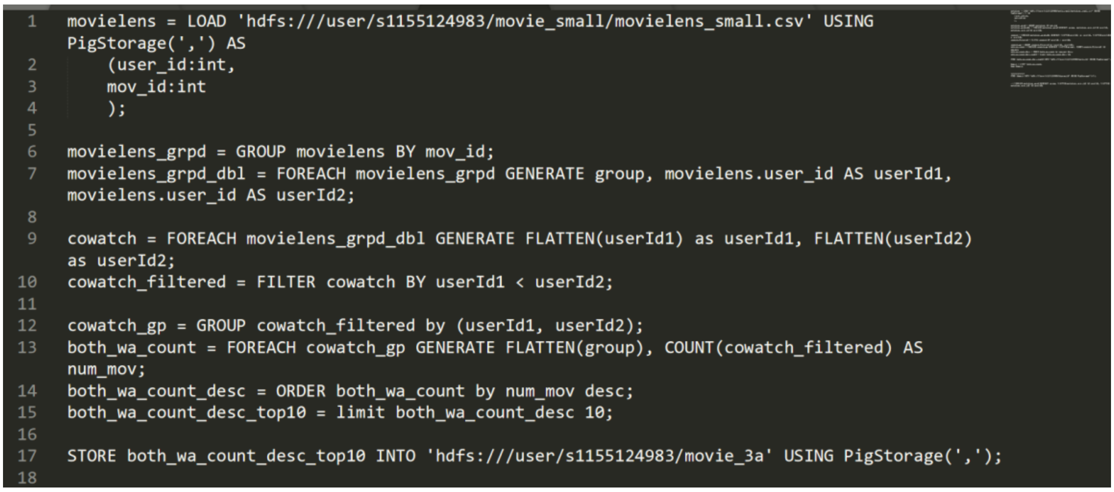
	Overall runtime:
Pig	14 mins
Hive	8 mins

From the above tasks, Hive is a faster than Pig. It may be due to the heavy usage of system in that time. With some studies on the performance of Hive and Pig, I found out that usually Pig is faster than Hive because of it use multi-query approach. Besides, Hive will create a lot of objects when performing join operation, this will further increase the runtime.

Q3

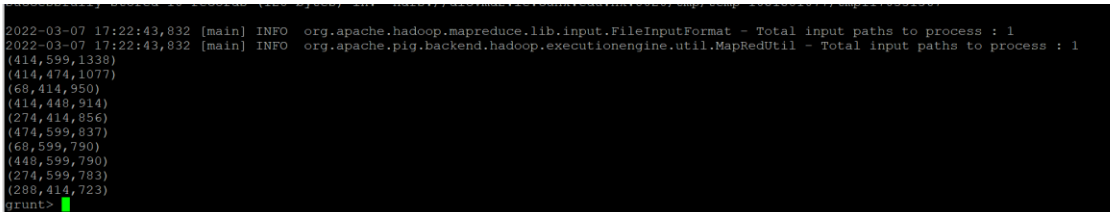
a.)

Code:



```
movielens = LOAD 'hdfs:///user/s1155124983/movie_small/movielens_small.csv' USING PigStorage(',') AS
    (user_id:int,
    mov_id:int
    );

movielens_grpd = GROUP movielens BY mov_id;
movielens_grpd_dbl = FOREACH movielens_grpd GENERATE group, movielens.user_id AS userId1, movielens.user_id AS userId2;

cowatch = FOREACH movielens_grpd_dbl GENERATE FLATTEN(userId1) as userId1, FLATTEN(userId2) as userId2;
cowatch_filtered = FILTER cowatch BY userId1 < userId2;

cowatch_gp = GROUP cowatch_filtered by (userId1, userId2);
both_wa_count = FOREACH cowatch_gp GENERATE FLATTEN(group), COUNT(cowatch_filtered) AS num_mov;
both_wa_count_desc = ORDER both_wa_count by num_mov desc;
both_wa_count_desc_top10 = limit both_wa_count_desc 10;

STORE both_wa_count_desc_top10 INTO 'hdfs:///user/s1155124983/movie_3a' USING PigStorage(',');
```

Output:

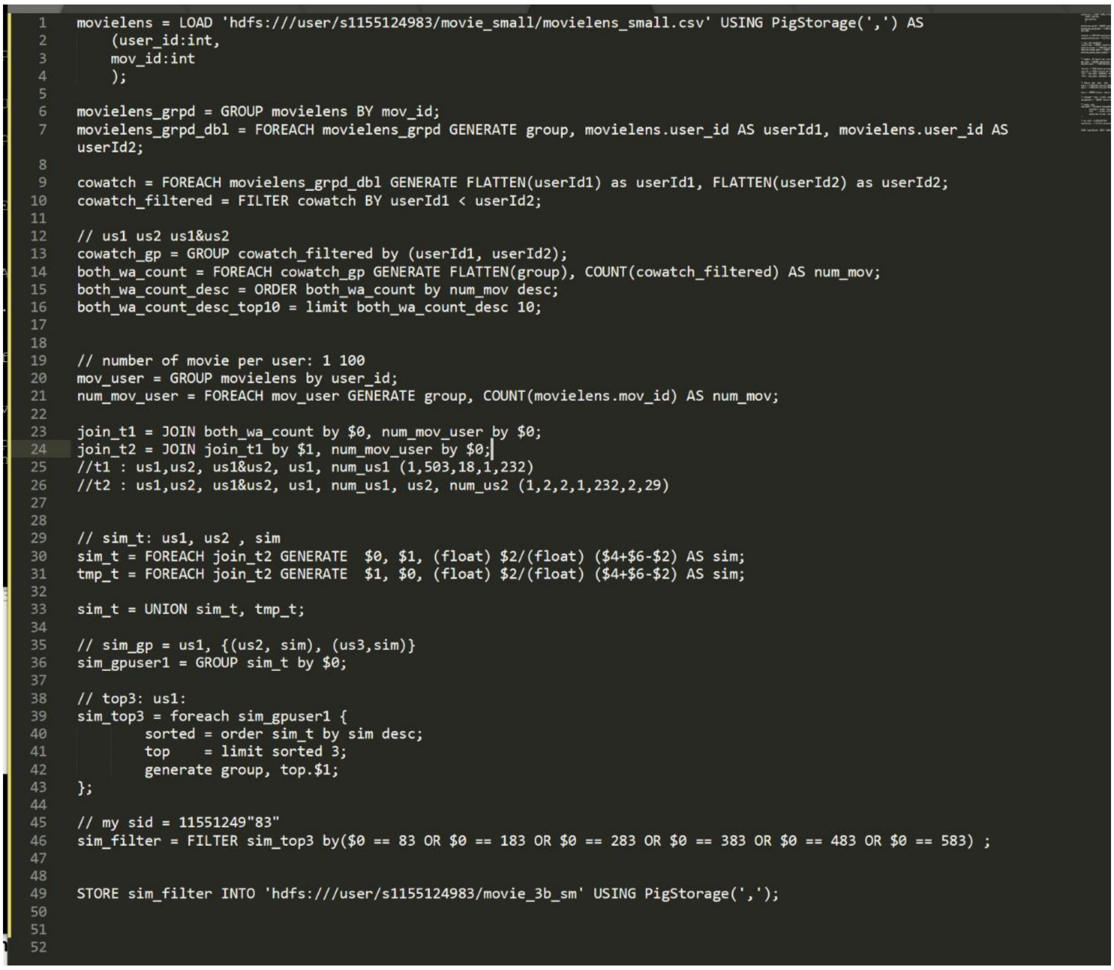


```
2022-03-07 17:22:43,832 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2022-03-07 17:22:43,832 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(414,599,1338)
(414,474,1077)
(68,414,950)
(414,448,914)
(274,414,856)
(474,599,837)
(68,599,790)
(448,599,790)
(274,599,783)
(288,414,723)
grunt> 
```

b.)

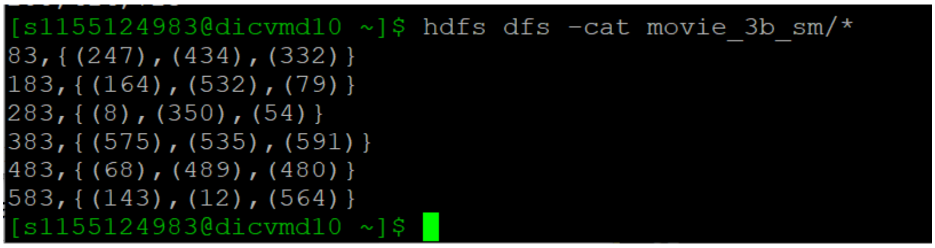
i) my SID is 11551249**83**

Code:



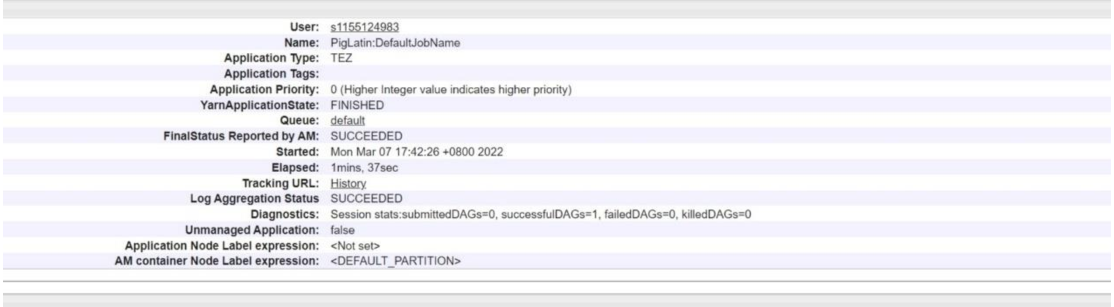
```
movielens = LOAD 'hdfs:///user/s1155124983/movie_small/movielens_small.csv' USING PigStorage(',') AS
    (user_id:int,
    mov_id:int
    );

movielens_grpd = GROUP movielens BY mov_id;
movielens_grpd_dbl = FOREACH movielens_grpd GENERATE group, movielens.user_id AS userId1, movielens.user_id AS userId2;

cowatch = FOREACH movielens_grpd_dbl GENERATE FLATTEN(userId1) as userId1, FLATTEN(userId2) as userId2;
cowatch_filtered = FILTER cowatch BY userId1 < userId2;

// us1 us2 us1&us2
cowatch_gp = GROUP cowatch_filtered by (userId1, userId2);
both_wa_count = FOREACH cowatch_gp GENERATE FLATTEN(group), COUNT(cowatch_filtered) AS num_mov;
both_wa_count_desc = ORDER both_wa_count by num_mov desc;
both_wa_count_desc_top10 = limit both_wa_count_desc 10;


// number of movie per user: 1 100
mov_user = GROUP movielens by user_id;
num_mov_user = FOREACH mov_user GENERATE group, COUNT(movielens.mov_id) AS num_mov;

join_t1 = JOIN both_wa_count by $0, num_mov_user by $0;
join_t2 = JOIN join_t1 by $1, num_mov_user by $0;
//t1 : us1,us2, us1&us2, us1, num_us1 (1,503,18,1,232)
//t2 : us1,us2, us1&us2, us1, num_us1, us2, num_us2 (1,2,2,1,232,2,29)


// sim_t: us1, us2 , sim
sim_t = FOREACH join_t2 GENERATE  $0, $1, (float) $2/(float) ($4+$6-$2) AS sim;
tmp_t = FOREACH join_t2 GENERATE  $1, $0, (float) $2/(float) ($4+$6-$2) AS sim;

sim_t = UNION sim_t, tmp_t;

// sim_gp = us1, {(us2, sim), (us3,sim)}
sim_gpuser1 = GROUP sim_t by $0;

// top3: us1:
sim_top3 = foreach sim_gpuser1 {
        sorted = order sim_t by sim desc;
        top    = limit sorted 3;
        generate group, top.$1;
};

// my sid = 11551249"83"
sim_filter = FILTER sim_top3 by($0 == 83 OR $0 == 183 OR $0 == 283 OR $0 == 383 OR $0 == 483 OR $0 == 583) ;


STORE sim_filter INTO 'hdfs:///user/s1155124983/movie_3b_sm' USING PigStorage(',');
```

Output:



```
[s1155124983@dicvmd10 ~]$ hdfs dfs -cat movie_3b_sm/*
83,{(247),(434),(332)}
183,{(164),(532),(79)}
283,{(8),(350),(54)}
383,{(575),(535),(591)}
483,{(68),(489),(480)}
583,{(143),(12),(564)}
[s1155124983@dicvmd10 ~]$
```

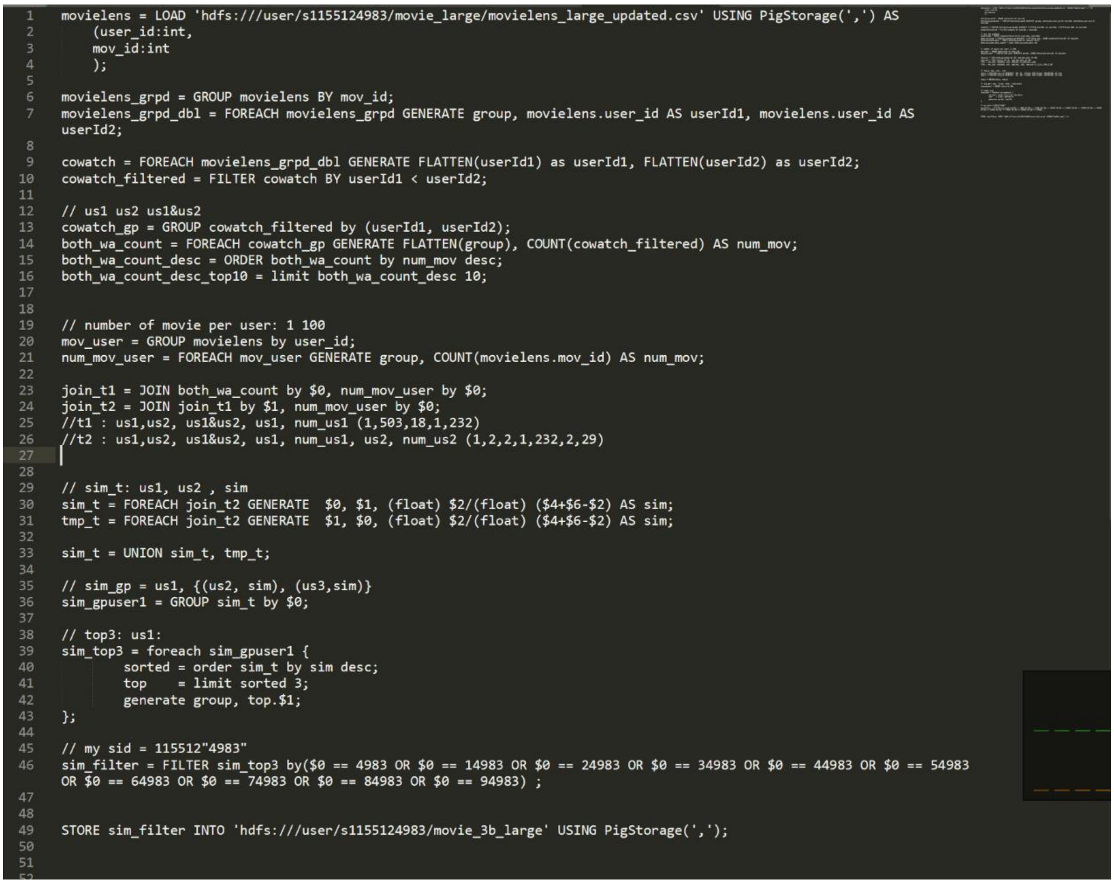
Time:2 minutes



User:	s1155124983
Name:	PigLatin:DefaultJobName
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Mon Mar 07 17:42:26 +0800 2022
Elapsed:	1mins, 37sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=1, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

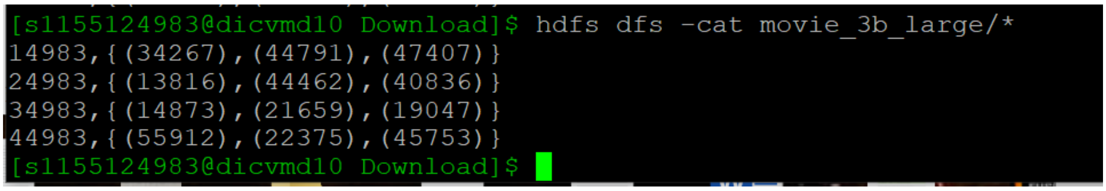
ii) Large dataset: my SID is 115512**4983**

Code:



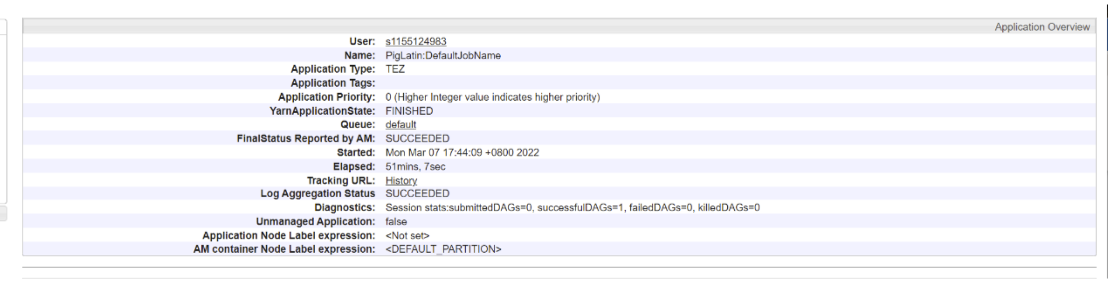
```
movielens = LOAD 'hdfs:///user/s1155124983/movie_large/movielens_large_updated.csv' USING PigStorage(',') AS
    (user_id:int,
    mov_id:int
    );

movielens_grpd = GROUP movielens BY mov_id;
movielens_grpd_dbl = FOREACH movielens_grpd GENERATE group, movielens.user_id AS userId1, movielens.user_id AS userId2;

cowatch = FOREACH movielens_grpd_dbl GENERATE FLATTEN(userId1) as userId1, FLATTEN(userId2) as userId2;
cowatch_filtered = FILTER cowatch BY userId1 < userId2;

// us1 us2 us1&us2
cowatch_gp = GROUP cowatch_filtered by (userId1, userId2);
both_wa_count = FOREACH cowatch_gp GENERATE FLATTEN(group), COUNT(cowatch_filtered) AS num_mov;
both_wa_count_desc = ORDER both_wa_count by num_mov desc;
both_wa_count_desc_top10 = limit both_wa_count_desc 10;


// number of movie per user: 1 100
mov_user = GROUP movielens by user_id;
num_mov_user = FOREACH mov_user GENERATE group, COUNT(movielens.mov_id) AS num_mov;

join_t1 = JOIN both_wa_count by $0, num_mov_user by $0;
join_t2 = JOIN join_t1 by $1, num_mov_user by $0;
//t1 : us1,us2, us1&us2, us1, num_us1 (1,503,18,1,232)
//t2 : us1,us2, us1&us2, us1, num_us1, us2, num_us2 (1,2,2,1,232,2,29)


// sim_t: us1, us2 , sim
sim_t = FOREACH join_t2 GENERATE  $0, $1, (float) $2/(float) ($4+$6-$2) AS sim;
tmp_t = FOREACH join_t2 GENERATE  $1, $0, (float) $2/(float) ($4+$6-$2) AS sim;

sim_t = UNION sim_t, tmp_t;

// sim_gp = us1, {(us2, sim), (us3,sim)}
sim_gpuser1 = GROUP sim_t by $0;

// top3: us1:
sim_top3 = foreach sim_gpuser1 {
        sorted = order sim_t by sim desc;
        top    = limit sorted 3;
        generate group, top.$1;
};

// my sid = 115512"4983"
sim_filter = FILTER sim_top3 by($0 == 4983 OR $0 == 14983 OR $0 == 24983 OR $0 == 34983 OR $0 == 44983 OR $0 == 54983 OR $0 == 64983 OR $0 == 74983 OR $0 == 84983 OR $0 == 94983) ;


STORE sim_filter INTO 'hdfs:///user/s1155124983/movie_3b_large' USING PigStorage(',');


```

Output:



```
[s1155124983@dicvmd10 Download]$ hdfs dfs -cat movie_3b_large/*
14983,{(34267),(44791),(47407)}
24983,{(13816),(44462),(40836)}
34983,{(14873),(21659),(19047)}
44983,{(55912),(22375),(45753)}
[s1155124983@dicvmd10 Download]$
```

Time: 51 minutes:



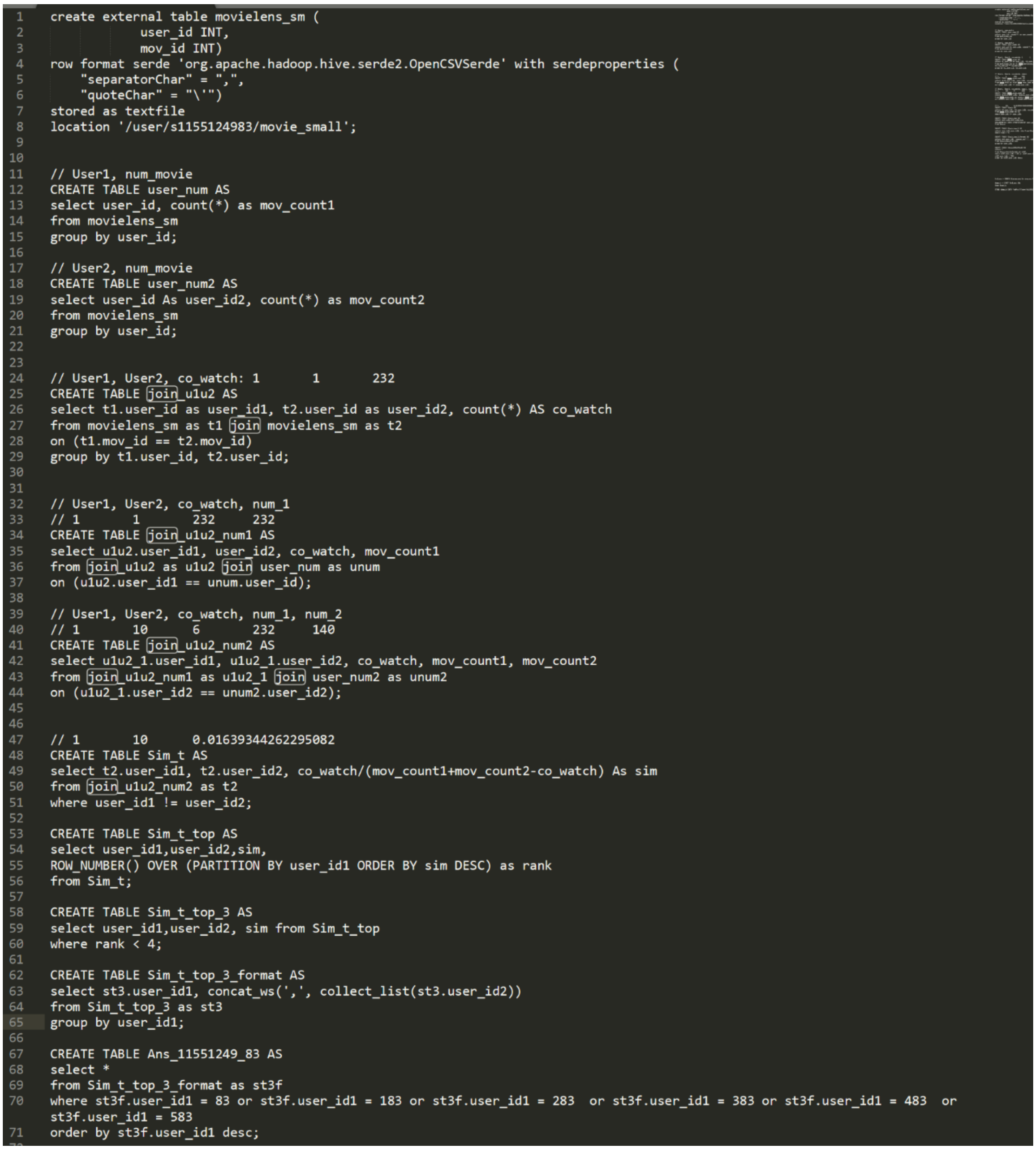
Application Overview

User:	s1155124983
Name:	PigLatin:DefaultJobName
Application Type:	TEZ
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Mon Mar 07 17:44:09 +0800 2022
Elapsed:	51mins, 7sec
Tracking URL:	History
Log Aggregation Status	SUCCEEDED
Diagnostics:	Session stats:submittedDAGs=0, successfulDAGs=1, failedDAGs=0, killedDAGs=0
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

c.)

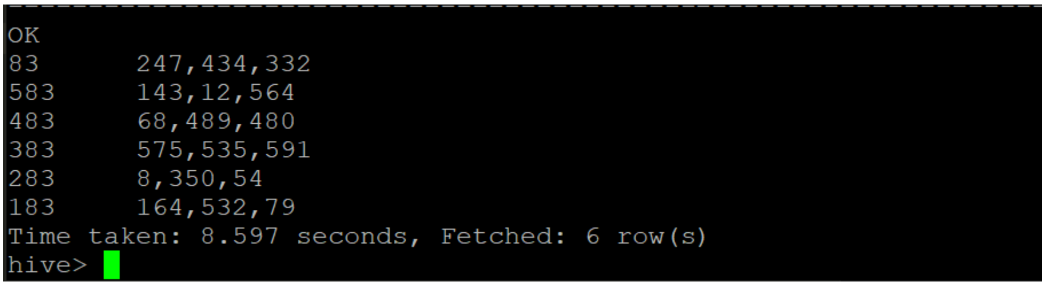
i)

Code:



```
create external table movielens_sm (
            user_id INT,
            mov_id INT)
row format serde 'org.apache.hadoop.hive.serde2.OpenCSVSerde' with serdeproperties (
    "separatorChar" = ",",
    "quoteChar" = "\'")
stored as textfile
location '/user/s1155124983/movie_small';


// User1, num_movie
CREATE TABLE user_num AS
select user_id, count(*) as mov_count1
from movielens_sm
group by user_id;

// User2, num_movie
CREATE TABLE user_num2 AS
select user_id As user_id2, count(*) as mov_count2
from movielens_sm
group by user_id;


// User1, User2, co_watch: 1       1       232
CREATE TABLE join_u1u2 AS
select t1.user_id as user_id1, t2.user_id as user_id2, count(*) AS co_watch
from movielens_sm as t1 join movielens_sm as t2
on (t1.mov_id == t2.mov_id)
group by t1.user_id, t2.user_id;


// User1, User2, co_watch, num_1
// 1       1       232     232
CREATE TABLE join_u1u2_num1 AS
select u1u2.user_id1, user_id2, co_watch, mov_count1
from join_u1u2 as u1u2 join user_num as unum
on (u1u2.user_id1 == unum.user_id);

// User1, User2, co_watch, num_1, num_2
// 1       10      6       232     140
CREATE TABLE join_u1u2_num2 AS
select u1u2_1.user_id1, u1u2_1.user_id2, co_watch, mov_count1, mov_count2
from join_u1u2_num1 as u1u2_1 join user_num2 as unum2
on (u1u2_1.user_id2 == unum2.user_id2);


// 1       10      0.01639344262295082
CREATE TABLE Sim_t AS
select t2.user_id1, t2.user_id2, co_watch/(mov_count1+mov_count2-co_watch) As sim
from join_u1u2_num2 as t2
where user_id1 != user_id2;

CREATE TABLE Sim_t_top AS
select user_id1,user_id2,sim,
ROW_NUMBER() OVER (PARTITION BY user_id1 ORDER BY sim DESC) as rank
from Sim_t;

CREATE TABLE Sim_t_top_3 AS
select user_id1,user_id2, sim from Sim_t_top
where rank < 4;

CREATE TABLE Sim_t_top_3_format AS
select st3.user_id1, concat_ws(',', collect_list(st3.user_id2))
from Sim_t_top_3 as st3
group by user_id1;

CREATE TABLE Ans_11551249_83 AS
select *
from Sim_t_top_3_format as st3f
where st3f.user_id1 = 83 or st3f.user_id1 = 183 or st3f.user_id1 = 283  or st3f.user_id1 = 383 or st3f.user_id1 = 483  or st3f.user_id1 = 583
order by st3f.user_id1 desc;
```

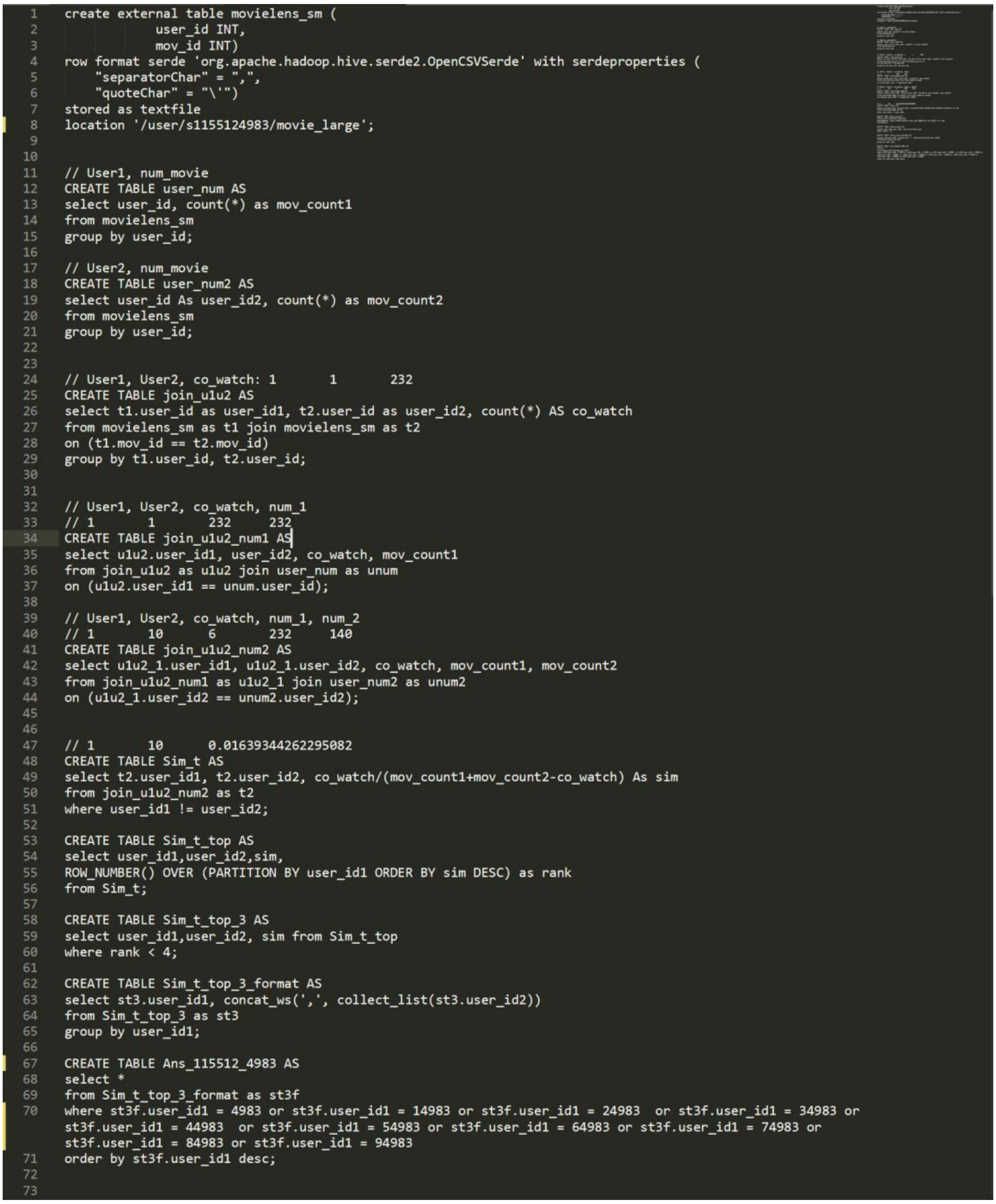
Output: The same as the answer in part b.



```
OK
83      247,434,332
583     143,12,564
483     68,489,480
383     575,535,591
283     8,350,54
183     164,532,79
Time taken: 8.597 seconds, Fetched: 6 row(s)
hive>
```

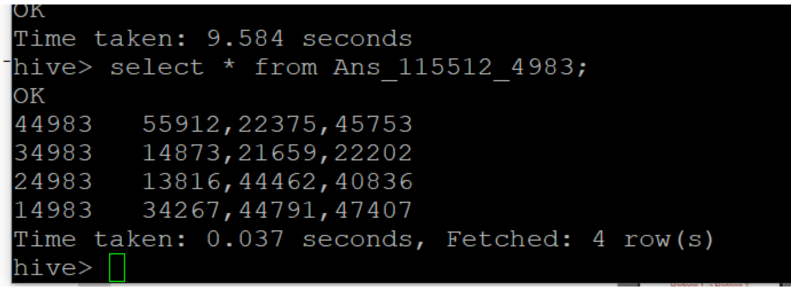
For large dataset:

Code:



```
create external table movielens_sm (
            user_id INT,
            mov_id INT)
row format serde 'org.apache.hadoop.hive.serde2.OpenCSVSerde' with serdeproperties (
    "separatorChar" = ",",
    "quoteChar" = "\'")
stored as textfile
location '/user/s1155124983/movie_large';


// User1, num_movie
CREATE TABLE user_num AS
select user_id, count(*) as mov_count1
from movielens_sm
group by user_id;

// User2, num_movie
CREATE TABLE user_num2 AS
select user_id As user_id2, count(*) as mov_count2
from movielens_sm
group by user_id;


// User1, User2, co_watch: 1       1       232
CREATE TABLE join_u1u2 AS
select t1.user_id as user_id1, t2.user_id as user_id2, count(*) AS co_watch
from movielens_sm as t1 join movielens_sm as t2
on (t1.mov_id == t2.mov_id)
group by t1.user_id, t2.user_id;


// User1, User2, co_watch, num_1
// 1       1       232     232
CREATE TABLE join_u1u2_num1 AS
select u1u2.user_id1, user_id2, co_watch, mov_count1
from join_u1u2 as u1u2 join user_num as unum
on (u1u2.user_id1 == unum.user_id);

// User1, User2, co_watch, num_1, num_2
// 1       10      6       232     140
CREATE TABLE join_u1u2_num2 AS
select u1u2_1.user_id1, u1u2_1.user_id2, co_watch, mov_count1, mov_count2
from join_u1u2_num1 as u1u2_1 join user_num2 as unum2
on (u1u2_1.user_id2 == unum2.user_id2);


// 1       10      0.01639344262295082
CREATE TABLE Sim_t AS
select t2.user_id1, t2.user_id2, co_watch/(mov_count1+mov_count2-co_watch) As sim
from join_u1u2_num2 as t2
where user_id1 != user_id2;

CREATE TABLE Sim_t_top AS
select user_id1,user_id2,sim,
ROW_NUMBER() OVER (PARTITION BY user_id1 ORDER BY sim DESC) as rank
from Sim_t;

CREATE TABLE Sim_t_top_3 AS
select user_id1,user_id2, sim from Sim_t_top
where rank < 4;

CREATE TABLE Sim_t_top_3_format AS
select st3.user_id1, concat_ws(',', collect_list(st3.user_id2))
from Sim_t_top_3 as st3
group by user_id1;

CREATE TABLE Ans_115512_4983 AS
select *
from Sim_t_top_3_format as st3f
where st3f.user_id1 = 4983 or st3f.user_id1 = 14983 or st3f.user_id1 = 24983  or st3f.user_id1 = 34983 or
st3f.user_id1 = 44983  or st3f.user_id1 = 54983 or st3f.user_id1 = 64983 or st3f.user_id1 = 74983 or
st3f.user_id1 = 84983 or st3f.user_id1 = 94983
order by st3f.user_id1 desc;
```

Output: Same as the answer in part b



```
OK
Time taken: 9.584 seconds
hive> select * from Ans_115512_4983;
OK
44983   55912,22375,45753
34983   14873,21659,22202
24983   13816,44462,40836
14983   34267,44791,47407
Time taken: 0.037 seconds, Fetched: 4 row(s)
hive> 
```