**AIT 580 – Data Analysis using Hadoop and Hive**

**Data Analysis using Hadoop and Hive:** Sales Records Dataset  
  
**Example Dataset Location:** https://repo.vse.gmu.edu/ait/AIT580/namelistexample.csv

**Assignment Dataset Location:** https://repo.vse.gmu.edu/ait/AIT580/salesrecords.csv

**Tool:**You need to use Terminal (command line interface) for this assignment *inside Virtual Box VM*. Please use the following instructions for learning, accessing and monitoring Hadoop and Hive environment for your assignment.

**Checking Hadoop Environment**

Hadoop is a distributed file system which is mainly used to store big data files by splitting them into smaller size called blocks and distributing them across multiple nodes/machines. This helps speed up the process by processing each file block on different machines in parallel. The framework used by Hadoop to process files in a distributive fashion and then collect the final output from different nodes is called MapReduce. It also provides redundancy/backup for data files since copies of same data is also distributed across nodes/machines.

1. Open Terminal by double clicking on Terminal icon on Desktop. This will open black terminal screen.
2. You should see a prompt administrator@virtualmachine:~$
3. For Hadoop environment, we use a different user called hduser. To switch to hduser, type the following command and press enter. Password is admin

su - hduser

1. You should see a prompt hduser@virtualmachine:~$
2. In order to check if all the daemons/processes required to run Hadoop environment, type the following command and press enter

jps

1. The output should look something like this

hduser@virtualmachine:~$ jps

3616 NodeManager

2437 NameNode

3494 ResourceManager

2765 DataNode

3133 SecondaryNameNode

4910 Jps

jps stands for *Java Virtual Machine Process Status Tool* which shows all the running java deamons/processes for Hadoop. First column represents process ID and second column represents process name. Your process ID can be different from above since every machine generated different process IDs. Just make sure you see all the process names in your system. For more details about these processes, please read Hadoop related material on blackboard under course content.

1. Once the Hadoop cluster is up and running check the web UI (user interface) of the components as described below. Open Chrome or Firefox browser on your Virtual Machine by clicking on the icon in the left menu panel.

|  |  |  |
| --- | --- | --- |
| Daemon/Process | Web Interface URL | Notes |
| NameNode | http://localhost:9870 | Default port is 9870 |
| ResourceManager | http://localhost:8088 | Default port is 8088 |

Please go through all the options available on the web interface and explore them for your understanding. Since we are using one singe VM, this Hadoop cluster is considered as Single Node Hadoop Clusters. While exploring the Web UI, you will find Number of Nodes as 1 or Active Nodes as 1. In actual production environment, you can setup as many clusters as needed. GMU also has a Hadoop cluster with multiple nodes so you can ask the system admin for access. For more details about different options and terminologies on web UI, please read Hadoop related material on blackboard under course content or search over internet.

1. Once you see the above expected output, you are ready to use Hadoop for data analysis.

**Checking Hive Environment**

Apache Hadoop also supports structures that allow SQL users (which is commonplace for relational data-base developers) to leverage the Hadoop platform. Apache Hive is an environment that allows SQL developers to write Hive Query Language (HQL) statements that are similar to standard SQL.

1. Make sure you are on the hduser prompt in Terminal hduser@virtualmachine:~$
2. Type the following command and press enter.

ls -l

-rw-r--r-- 1 hduser hadoop 22201 Aug 19 01:19 derby.log

-rw-r--r-- 1 hduser hadoop 8980 Aug 17 16:02 examples.desktop

drwxr-xr-x 5 hduser hadoop 4096 Aug 19 01:19 metastore\_db

drwxr-xr-x 5 hduser hadoop 4096 Aug 18 13:46 metastore\_db.tmp

1. You should see the following files

Hive needs a metastore service, where it stores metadata such as all the information related to our databases, tables, and relations. Whenever you want to know something about your database, you can find it easily in the Metastore (metastore\_db file above).

By default, Hive uses a built-in **Derby SQL** server which creates and stores actual tables in a relational database (derby.log stores all the logs for your queries).

1. Once you see the files above, type the command below to run hive command line interface

hive

1. You should see the following output on screen

hduser@virtualmachine:~$ hive

Hive Session ID = 9035e803-cc92-461c-90ff-b21766b6f288

Logging initialized using configuration in jar:file:/usr/local/hive/lib/hive-common-3.1.0.jar!/hive-log4j2.properties Async: true

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 releases.

Hive Session ID = 8d62cfc1-da3f-4d72-965f-61275f17e2ff

hive>

1. On the hive prompt hive> type the following command to see existing databases.

show databases;

1. You should see the following output

hive> show databases;

OK

default

Time taken: 0.694 seconds, Fetched: 1 row(s)

The default database name in hive is “default”

1. On the hive prompt hive> type the following command and press enter to see existing tables.

show tables;

1. You should see the following output

hive> show tables;

OK

Time taken: 0.068 seconds

hive>

Since we have not created any table yet, there is no table names appeared on the screen.

1. Use the following command in hive to terminate the hive session and going back to main terminal environment.

quit;

hive> quit;

hduser@virtualmachine:~$

1. Once you see the above expected output, you are ready to use Hive for data analysis.

**Copy/Store data into Hadoop Distributed File System (HDFS)**

1. Make sure you quit the hive interface and see this prompt hduser@virtualmachine:~$
2. Type the following command and press enter to download example dataset

wget https://repo.vse.gmu.edu/ait/AIT580/namelistexample.csv

1. To check if the file has been downloaded, use the command: ls –l
2. You should see the above file in your list.
3. Use the same link above and download the csv data file using web browser so that you can open the file and see the content in your VM.
4. You also need to know the path of your working directory. Type the following command and press enter to see your present working directory

pwd

1. You should see the following output

hduser@virtualmachine:~$ pwd

/home/hduser

/home/hduser is the present working directory. We need this path when we copy files from our local directory to Hadoop file system HDFS. We will use naming convention HDFS to refer to Hadoop file system from now on.

1. Create a new HDFS directory to store data files. Type the following command and press enter. You can use any path and directory name of your choice. Here we use /user/hive/warehouse.

hdfs dfs -mkdir -p /user/hive/warehouse

1. If you do not see any error message then directory has been created. Type the following command and press enter to see the HDFS directory.

hdfs dfs -ls /user/hive/

1. Please remember that this path does not exist in your local virtual machine folders. This is HDFS directory and can only be accessible through hdfs dfs commands.
2. To copy the csv data file from local VM to HDFS, type the following command and press enter

hdfs dfs -put namelistexample.csv /user/hive/warehouse/namelist

1. If you do not see any error message then directory has been created. The csv data file will be stored in an HDFS directory called namelist. To view the directory, type the following command and press enter

hdfs dfs -ls /user/hive/warehouse/namelist

The hdfs dfs -ls command will not show you the actual block size storage of the file. You can see the HDFS actual storage information using command

hdfs fsck /user/hive/warehouse/namelist -files -blocks -locations

Or you can use the NameNode UI http://localhost:9870 to see which hosts have the blocks for a file (On top menu, click Utilities > Browse the file system option). This method is recommended

In this case of namelistexample.csv which is a very small file, you do not see any split. This is because of the reason that default block size in Hadoop is 64MB, so if the file is less than 64MB it will be stored as single block/file. Therefore we only see one block for this file.

**Create Hive table to access HDFS data**

1. Make sure you are on the hduser prompt in Terminal hduser@virtualmachine:~$
2. Once you see the files above, type the command below to run hive command line interface

hive

1. **Create a Hive table** based on columns present in the csv file. You can open csv file in your VM and see the content.

create external table namelisttable (

id int,

name string,

gender string

)

row format delimited

fields terminated by ','

stored as textfile

location '/user/hive/warehouse/namelist\_hive\_table'

TBLPROPERTIES ("skip.header.line.count"="1");

This is very similar to standard SQL query. The new hive table name is namelisttable. Datatypes for each column are also defined. Comma is used as field terminator since we are using csv file. New location is provided in HDFS to store hive table data. TBLPROPERTIES is used to skip first line in the csv file since it contains a header and not the actual row of data.

1. If you go back to see NameNode UI http://localhost:9870 and browse file system, you will see two directories namelist (for the csv file) and namelist\_hive\_table for hive table.
2. Type the command to see the table schema describe namelisttable;
3. Now we have an empty table, we need to load data from csv data file into this table. Type the following command and press enter

load data inpath '/user/hive/warehouse/namelist' into table namelisttable;

/user/hive/warehouse/namelist is the path of HDFS where we store our csv file.

Please note that once you load the csv file from HDFS to Hive table, the original data directory will move inside Hive HDFS directorty. In our case /namelist will move into /namelist\_hive\_table/namelist. If you go back to see NameNode UI http://localhost:9870 and browse file system, you will see /user/hive/warehouse/namelist\_hive\_table/namelist as the updated HDFS directory.

1. If you see no error message, the loading is complete. You can run the following command to see if your table contains any data

select \* from namelisttable;

1. You can also use where clause such as

select \* from namelisttable where gender='F';

1. When you run more complex queries, you will see different messages appearing for launching map reduce jobs which are required to process and collect output in distributed environment. For example running the following query will show you some messages

select gender, count(\*) as cnt group by gender;

1. If you are able to see the expected output, you are ready to do assignment on a big dataset.

In Hive interface you can write query in multiple lines by pressing enter for new line. Once your query is completed, just put ;in the end and press enter to execute the whole query.

To exit from Hive, type quit; and press enter

To exit from hduser@virtualmachine:~$ type exit and press enter

If you are at the prompt administrator@virtualmachine:~$ type exit and press enter to close the terminal

**Assignment Questions:**

Perform the same procedure above with Assignment Dataset

**Assignment Dataset Location:** https://repo.vse.gmu.edu/ait/AIT580/salesrecords.csv

Please answer the following questions

1. Create a Hive table based on columns present in the csv file. Include the create table command and show the table schema using the describe <tablename> command [10]
2. Visit the NameNode web UI interface and report the number of blocks for the dataset file. Also include all the block information [10 points]
3. Write a Hive SQL query and show output of the following statements
   1. Sort the Region based on their count in Descending order [10 points]
   2. Count the number of Item Type and Sales Channel. Sort the output in Descending order with respect to count [10 points]
   3. Find the Item Type with count where Total Cost is less than Total Profit [10 points]

**References:**

Use this link to see Hive built-in functions <https://www.qubole.com/resources/hive-function-cheat-sheet/>