Introduction

The objective of this document is to help you to get started with your project. It covers the most important technical tasks you have to perform to complete your hive project.

Loading gdelt data

There are multiple ways to obtain the gdelt data. We will keep it simple by using the application wget

The gdelt files are located here :

http://data.gdeltproject.org/gdeltv2/masterfilelist.txt

1.**To obtain the gdelt data:**

Log into the york server and open a terminal(Open Terminal (Applications -> Favorites -> Terminal):

At the terminal enter the following command to create a directory, called events by running the following commands at the terminal:

“‘console

```
>> mkdir events
```

“‘ change into the directory using

```
>> cd events
```

Run the following the command to download the events file

```
>> wget -c http://data.gdeltproject.org/gdeltv2/20180806114500.export.CSV.zip
```

This will download the file to the current directory: you can check this by running the following command at the terminal.

```
>> ls 20180806114500.export.CSV.zip
```

Unzip the file by running the following the command

```
>> unzip 20180806114500.export.CSV.zip
```

You now have a file 20180806114500.export.CSV

Copy the file to hdfs:

1. Create a directory in hadoop called gdelt/events by using the following command:

```
bash    >> hdfs dfs -mkdir -p  <your_id>/gdelt/events
```

For example if is 3283939 the above command should be:

```
bash    >>  hdfs dfs -mkdir -p 32833939/gdelt/events
```

2. You can now copy the event files you downloaded earlier to the hdfs directory you just created by running the following command

“‘sql >> hdfs dfs -put *.csv 32833939/gdelt/events/

“‘

3. Confirm that the files have been copied to hadoop my running the following command:

“‘sql >> hdfs dfs -ls 32833939/gdelt/events/

“‘ If the file was successfully copy you see the file listed in the output from the above command

4. You can load additional gdelt events file into the same folder by repeating the steps

Getting started with Hive:

Now that you now how to load data into hadoop, lets get started with processing it using apache hive

Writing a hive script.

Lets get started

1. Open the following text editor by typing the following command at the terminal

“‘bash

```
>> gedit &
```

‘‘‘

This will open the gedit text editor. Once the editor is open, save the document as gdelt_analysis.hql

Creating the database:

1. Enter the following hive commands in the document.

   “‘sql

CREATE DATABASE IF NOT EXISTS gdelt_ COMMENT 'This is the gdelt database' With dbproperties ('Created by' = 'Leotis Buchanan','Created on' = 'August-2018');

SHOW DATABASES;

"'

Replace the text with your login id The above sql snippet will create a database called gdelt

3. At the terminal enter the following command to create the hive database

```
hive -f gdelt_analysis.hql
```

This should print a lot of text including the gdelt_ database.

Creating the events table tables:

1. Add the following script to the gdelt_analysis.hql file. After do this your file should have the following content: Please ensure that the **LOCATION** value is set to the directory that you stored the data in hdfs. In my case it was '32833939/gdelt/events';

```
CREATE EXTERNAL TABLE gdelt.events IF NOT EXISTS (globaleventid INT,
sqldate INT,
monthyear INT,
year INT,
fractiondate FLOAT,
actor1code STRING,
actor1name STRING,
actor1countrycode STRING,
actor1knowngroupcode STRING,
actor1ethniccode STRING,
actor1religion1code STRING,
actor1religion2code STRING,
actor1type1code STRING,
actor1type2code STRING,
actor1type3code STRING,
actor2code STRING,
actor2name STRING,
actor2countrycode STRING,
actor2knowngroupcode STRING,
actor2ethniccode STRING,
actor2religion1code STRING,
actor2religion2code STRING,
actor2type1code STRING,
```

```
actor2type2code STRING,
actor2type3code STRING,
isrootevent INT,
eventcode STRING,
eventbasecode STRING,
eventrootcode STRING,
quadclass INT,
goldsteinscale FLOAT,
nummentions INT,
numsources INT,
numarticles INT,
avgtone FLOAT,
actor1geo_type INT,
actor1geo_fullname STRING,
actor1geo_countrycode STRING,
actor1geo_adm1code STRING,
actor1geo_adm2code STRING,
actor1geo_lat FLOAT,
actor1geo_long FLOAT,
actor1geo_featureid STRING,
actor2geo_type INT,
actor2geo_fullname STRING,
actor2geo_countrycode STRING,
actor2geo_adm1code STRING,
actor2geo_adm2code STRING,
actor2geo_lat FLOAT,
actor2geo_long FLOAT,
actor2geo_featureid STRING,
actiongeo_type INT,
actiongeo_fullname STRING,
actiongeo_countrycode STRING,
actiongeo_adm1code STRING,
actiongeo_adm2code STRING,
actiongeo_lat FLOAT,
actiongeo_long FLOAT,
actiongeo_featureid STRING,
dateadded BIGINT,
sourceurl STRING)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t'
STORED AS TEXTFILE
LOCATION '/32833939/gdelt/events';

DESCRIBE gdelt.events
```

This will create a table called events

Exploring cleaning your data with hive

In order to complete the project you will have to analyse the data. You will do this using sql queries.

1. Create a new file called data_exploration.hql
2. Enter the following sql statements in the file:

```
USE gdelt

select globaleventid,actor1code,sourceurl from gdelt.events;
```

You can perform a wide array of data exploration using sql queries. To learn more about hive sql queries see the recommended course textbook or visit the following website

https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF#LanguageManualUDF-Built-inOperators

Saving analytic results to hadoop

As you work you will want to save the intermediate results of your analysis. The best practice is to save all your work hdfs. You can save the results of query to hdfs by doing the following:

1. Create the following directory in hdfs by using the following command:

```
>> hdfs dfs -mkdir -p /32833939/gdelt/events/processed/positive_goldstein
```

2. After you have created the above folder run the following query:

```
USE

insert overwrite  directory '/32833939/gdelt/events/processed/positive_goldstein'
row format delimited
fields terminated by '\t'
stored as textfile
select actor1code, goldsteinscale from gdelt.events
where goldsteinscale > 0;
```

3. After running the above query the output of the select query will be stored in /32833939/gdelt/events/processed/positive_goldstein folder. You can verify this by running

```
>>  hdfs dfs -ls /32833939/gdelt/events/processed/positive_goldstein/
```

Retrieving results from hadoop

Once you have done your analysis and cleaning you will want to copy your processed data to your local computer for something like visualizing. This can be accomplished using the following hdfs command

```
>> hdfs dfs -copyToLocal /32833939/gdelt/events/processed/positive_goldstein    positive_gol
```

The data that you need will be in the positive_goldstein folder.

After you have copied your file to your local computer you can now visualize your data using python or R.