

Analysis of Airport Data Using Hive & Pig

Case Study

19CSE357 – Big Data Analytics



Date: February 23, 2022

Group Details:

S. No	Name of the Student	Roll No.
1.	KOSURI DIVESH	CB.EN.U4CSE19422
2.	PENUGONDA KOUSHIK	CB.EN.U4CSE19449
3.	RAVELLA ABHINAV	CB.EN.U4CSE19453
4.	SINGADI SHANTHAN REDDY	CB.EN.U4CSE19459

Dataset Description

The main aim of the dataset is to develop a model for the airline data to provide a platform for new analytics based on the following queries as the problem faced is the existing has the ability to analyze limited data from the following databases

In our case study we are dealing with 3 different datasets named airports_mod, Final_airlines, routes

Fields:

Airports_mod:

- **Sample:** Goroka,Goroka,Papua New Guinea,GKA,AYGA,-6.081689,145.391881,5282,10,U,Pacific/Port_Moresby
- Dataset contains mainly unique Airport ID, Name of the airport, City of the respective airport, Country, 3-letter IATA code, Latitude & Longitude, Altitude, Timezone

Final_Airlines:

- **Sample:** 2,135 Airways,\N,,GNL,GENERAL,United States,N
- In this dataset it contains ID, Name of airline, Shortcut of airline, IATA, ICAO, Callsign, Country

Routes:

- **Sample:** 2B,410,AER,2965,KZN,2990,,0,CR2
- This dataset contains mainly 3-letter ICAO code, Airline ID, Source airport ID&Code, Destination ID & Code, Halts

Outcome:

We tried to explore detailed analysis on airline datasets such as listing airports operations, list of airlines having no halts etc., Here we mainly focussed on the processing of big datasets using hive component of Hadoop ecosystem in distributed environment.

At last, it will be useful in accessing and processing their user queries.

Loading the Dataset:

Hive:

```
hduser@abhinav:~$ hadoop fs -mkdir /casestudy
2022-03-21 15:23:13,807 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@abhinav:~$ hadoop fs -put /home/hduser/Downloads/airports_mod.dat /casestudy
2022-03-21 15:23:24,228 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@abhinav:~$ hadoop fs -put /home/hduser/Downloads/Final_airlines /casestudy
2022-03-21 15:23:34,139 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@abhinav:~$ hadoop fs -put /home/hduser/Downloads/routes.dat /casestudy
2022-03-21 15:23:45,366 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
hduser@abhinav:~$
```

Show entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	hduser	supergroup	308.83 KB	Mar 21 15:23	1	128 MB	Final_airlines	
<input type="checkbox"/>	-rw-r--r--	hduser	supergroup	722.18 KB	Mar 21 15:23	1	128 MB	airports_mod.dat	
<input type="checkbox"/>	-rw-r--r--	hduser	supergroup	2.27 MB	Mar 21 15:23	1	128 MB	routes.dat	

Showing 1 to 3 of 3 entries

Queries:

Hive:

1. Creating table airport for airports_mod dataset:

create table airports (airport_id int,airport_name string,airport_city string,airport_country string,airport_faa string,airport_icao string,airport_lat double,airport_long double,airport_alt double,airport_timezone double,airport_dst string,airport_tz string) row format delimited fields terminated by ',';

```
hive> create table airports (airport_id int,airport_name string,airport_city string,airport_country string,airport_faa string,airport_icao string,airport_lat double,airport_long double,airport_alt double,airport_timezone double,airport_dst string,airport_tz string) row format delimited fields terminated by ',';
OK
Time taken: 1.058 seconds
```

2. Creating table final airlines for Final_airlines :

create table final_airlines (airlineID string,airline_name string, airline_alias string, airline_iata string, airline_icao string,callsign string,territory string, active string) row format delimited fields terminated by ',';

```
hive> create table final_airlines (airlineID string,airline_name string, airline_alias string, airline_iata string, airline_icao string,callsign string,territory string, active string) row format delimited fields terminated by ',';
OK
Time taken: 1.069 seconds
hive> create table routes (route_iata string,route_airid int,route_source_iata string,route_source_airid int,route_des_iata string,route_des_airid int,route_codeshare string,route_stops int,route equip s string) row format delimited fields terminated by ',';
OK
Time taken: 0.12 seconds
```

3. Creating table route for routes.dat:

create table routes (route_iata string,route_airid int,route_source_iata string,route_source_airid int,route_des_iata string,route_des_airid int,route_codeshare string,route_stops int,route equip string) row format delimited fields terminated by ',';

```
hive> show tables;
OK
airports
final_airlines
routes
Time taken: 0.089 seconds, Fetched: 3 row(s)
hive>
```

4. loading data into airport table

load data inpath '/airports_mod.dat' into table airports;

```
hive> load data inpath '/casestudy/airports_mod.dat' into table airports;
Loading data to table default.airports
OK
Time taken: 1.27 seconds
hive>
```

5. loading data into final airlines table

load data inpath '/Final_airlines' into table final_airlines;

6. loading data into route table

load data inpath '/routes.dat' into table routes;

```
hive> load data inpath '/casestudy/airports_mod.dat' into table airports;
Loading data to table default.airports
OK
Time taken: 1.27 seconds
hive> load data inpath '/casestudy/Final_airlines' into table airports;
Loading data to table default.airports
OK
Time taken: 0.363 seconds
hive> load data inpath '/casestudy/routes.dat' into table airports;
Loading data to table default.airports
OK
Time taken: 0.344 seconds
hive>
```

RAVELLA ABHINAV – CB.EN.U4CSE19453

Queries:

Hive:

1. Find all the airlines that are active and have an alias names

Query:

create table alias_not_null_airlines as SELECT * FROM final_airlines
WHERE airline_alias IS NOT NULL AND active="Y";

Result :

```
hive> create table alias_not_null_airlines as SELECT * FROM final_airlines WHERE airline_alias IS NOT NULL AND active="Y";
Query ID = hduser_20220322100631_7b70eb96-cde3-4913-a9cd-52c2a86c6138
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Job running in-process (local Hadoop)
2022-03-22 10:06:34,839 Stage-1 map = 0%, reduce = 0%
2022-03-22 10:06:35,878 Stage-1 map = 100%, reduce = 0%
Ended Job = job_local216938443_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:54310/user/hive/warehouse/.hive-staging_hive_2022-03-22_10-06-31_052_1734083296723987011-1/-ext-10002
Moving data to directory hdfs://localhost:54310/user/hive/warehouse/alias_not_null_airlines
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 10503019 HDFS Write: 21692 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 5.629 seconds
hive>
```

- Querying the first 10 rows of the resulted table:

Query: SELECT * FROM alias_not_null_airlines LIMIT 10;


```
hive> SELECT * FROM alias_not_null_airlines LIMIT 10;
OK
324 All Nippon Airways ANA All Nippon Airways NH ANA ALL NIPPON Japan Y
576 AirAsia Air Asia AK AXM ASIAN EXPRESS Malaysia Y
641 Rossiya-Russian Airlines Pulkovo Aviation Enterprise FV SDM PULKOVO Russia Y
1437 bmi bmi British Midland BD BMA MIDLAND United Kingdom Y
1531 Brussels Airlines SN Brussels Airlines SN DAT BEE-LINE Belgium Y
1879 Contact Air Contactair C3 KIS CONTACTAIR Germany Y
1946 Czech Airlines CSA Czech Airlines OK CSA CSA-LINES Czech Republic Y
2183 Emirates Emirates Airlines EK UAE EMIRATES United Arab Emirates Y
2297 easyJet EasyJet Airline U2 EZY EASY United Kingdom Y
2417 AirAsia X FlyAsianXpress D7 XAX XANADU Malaysia Y
Time taken: 0.411 seconds, Fetched: 10 row(s)
hive>
```

Explanation:

Job is to find list of airlines with alias names and are still operating (Active). This can be achieved by querying using 'WHERE', 'IS NOT NULL' and 'AND' keywords. In the dataset, all the airlines that has no alias names have 'NULL' as the value in their respective cells. So 'IS' 'NOT' 'NULL' keywords are to be used to fetch all the values rows that have alias names and active status can be directly done using 'WHERE' clause.

2. Find the count of airlines that choose to have routes with 1 stop.

Query: select count(route_airid) from routes where route_stops like "%1%"

Output:

```
hive> select count(route_airid) from routes where route_stops like "%1%";
Query ID = hduser_20220322101423_0447074f-e6cc-4088-9903-196f3f8ede71
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)
2022-03-22 10:14:25,751 Stage-1 map = 100%, reduce = 0%
2022-03-22 10:14:26,790 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local1580320474_0002
MapReduce Jobs Launched:
Stage-Stage-1: HDFS Read: 25765422 HDFS Write: 43384 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
11
Time taken: 3.636 seconds, Fetched: 1 row(s)
hive>
```

Explanation:

Job here is to find the total count of airlines that has one stop in its routes. So we are to query on routes table we already created using one of the aggregate function “count”.

1. select all the route id's which have their no of stops equal to 1
2. Add the aggregate function “count” to count the no of ids that are resulted as a result of first query.

3. Find all airports in the world which lie at an altitude greater than 5000 ft.

Query: create table high_alt_airports as select * from airports where airport_alt > 5000;

Result:

```
hive> create table high_alt_airports as select * from airports where airport_alt > 5000;
Query ID = hduser_20220322101742_056c5cee-fd82-4137-9922-1ebec2490a38
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Job running in-process (local Hadoop)
2022-03-22 10:17:45,255 Stage-1 map = 100%, reduce = 0%
Ended Job = job_local2075807615_0003
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:54310/user/hive/warehouse/.hive-staging_hive_2022-03-22_10-17-42_629_8769889742244920012-1/-ext-10002
Moving data to directory hdfs://localhost:54310/user/hive/warehouse/high_alt_airports
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 16313974 HDFS Write: 52226 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 3.178 seconds
hive>
```

Subquery:

select * from high_alt_airports limit 10;

```
hive> select * from high_alt_airports limit 10;
OK
1      Goroka  Goroka  Papua New Guinea  GKA  AYGA  -6.081689  145.391881  5282.0  10.0  U  Pacific/Port_Moresby
3      Mount Hagen  Mount Hagen  Papua New Guinea  HGU  AYMH  -5.826789  144.295861  5388.0  10.0  U  Pacific/Port_Moresby
792    Brakpan  Brakpan  South Africa  FABB  -26.23865  28.301769  5300.0  2.0  U  Africa/Johannesburg
795    Bethlehem  Bethlehem  South Africa  FABM  -28.248392  28.336125  5561.0  2.0  U  Africa/Johannesburg
801    Ermelo  Ermelo  South Africa  FAEO  -26.495644  29.979764  5700.0  2.0  U  Africa/Johannesburg
802    Ficksburg  Ficksburg  South Africa  FAFB  -28.823119  27.9089  5315.0  2.0  U  Africa/Johannesburg
803    Grand Central  Johannesburg  South Africa  GCJ  FAGC  -25.986267  28.140061  5325.0  2.0  U  Africa/Johannesburg
810    Harritsmith  Harritsmith  South Africa  FAHR  -28.235072  29.106206  5585.0  2.0  U  Africa/Johannesburg
813    Johannesburg Intl  Johannesburg  South Africa  JNB  FAJS  -26.139166  28.246  5558.0  2.0  U  Africa/Johannesburg
816    Krugersdorp  Krugersdorp  South Africa  FAKR  -26.080978  27.725667  5499.0  2.0  U  Africa/Johannesburg
Time taken: 0.44 seconds, Fetched: 10 row(s)
hive>
```

Explanation:

Job is to find out the list of all the airports at higher altitudes (alt > 5000 ft) we use a binary operator “>” to select all the airports that have their airport alt > 5000.

1. We first use the select clause to find all the airports above air_alt > 5000, create a new table high_alt_airports and store the result of above query in that new table.
2. Now we query the table for 10 airports with altitude above 5000 using 'LIMIT' keyword.