Assignment 7-1b

DSC 650

Jake Meyer

04/28/2023

Next, we are going to partition the dataset again, but this time we will partition by the hash value of the key. The following is a function that will create a SHA256 hash of the input key and return a hexadecimal string representation of the hash. import hashlib def hash_key(key): m = hashlib.sha256() m.update(str(key).encode('utf-8')) return m.hexdigest() We will partition the data using the first character of the hexadecimal hash. As such, there are 16 possible partitions. Create a new column called hashed that is a hashed value of the key column. Next, create a partitioned dataset based on the first character of the hashed key and save the results to results/hash. The directory should contain the following folders. hash ├── hash_key=0 ├── hash_key=1 ├── hash_key=2 ├── hash_key=3 ├── hash_key=4 ├── hash_key=5 ├── hash_key=6 ├── hash_key=7 ├── hash_key=8 ├── hash_key=9 ├── hash_key=A ├── hash_key=B ├── hash_key=C ├── hash_key=D ├── hash_key=E

In [14]:

```python
## Import the necessary packages for the assignment.
import pandas as pd
import pyarrow as pa
import pyarrow.parquet as parq
## import pathlib
from pathlib import Path
import hashlib
```

In [15]:

```python
## Print versions of essential packages
print("pandas version: {}".format(pd.__version__))
print("pyarrow version: {}".format(pa.__version__))
```

```
pandas version: 1.5.3
pyarrow version: 11.0.0
```

In [16]:

```python
## Setup directories
cwd = Path('C:/Users/jkmey/Documents/Github/DSC650_Course_Assignments/dsc650/dsc650
results_dir = cwd.joinpath('results')
pq_file = results_dir.joinpath('routes.parquet')
partitioned_pq_file = results_dir.joinpath('hash')
```

Load the dataset using read_parquet

In [17]:

```python
## Use read_parquet() to read routes.parquet
pq = pd.read_parquet(pq_file, engine = 'fastparquet')
```

In [18]:

```python
print(list(pq.columns.values))
```

```
['codeshare', 'equipment', 'airline.active', 'airline.airline_id', 'airline.alia
s', 'airline.callsign', 'airline.country', 'airline.iata', 'airline.icao', 'airlin
e.name', 'src_airport.airport_id', 'src_airport.altitude', 'src_airport.city', 'sr
c_airport.country', 'src_airport.dst', 'src_airport.iata', 'src_airport.icao', 'sr
c_airport.latitude', 'src_airport.longitude', 'src_airport.name', 'src_airport.sou
rce', 'src_airport.timezone', 'src_airport.type', 'src_airport.tz_id', 'dst_airpor
t.airport_id', 'dst_airport.altitude', 'dst_airport.city', 'dst_airport.country',
'dst_airport.dst', 'dst_airport.iata', 'dst_airport.icao', 'dst_airport.latitude',
'dst_airport.longitude', 'dst_airport.name', 'dst_airport.source', 'dst_airport.ti
mezone', 'dst_airport.type', 'dst_airport.tz_id']
```

Function to Create a SHA256 Hash of Input Key and Return Hexadecimal String

In [19]:

```python
## Function provided in the assignment instructions.
def hash_key(key):
    m = hashlib.sha256()
    m.update(str(key).encode('utf-8'))
    return m.hexdigest()
```

Create a New Hashed Column

In [20]:

```python
## Create the concatenated key with src_airport.iata + dst_airport.iata+ airline.ic
pq['key'] = pq['src_airport.iata'] + pq['dst_airport.iata'] + pq['airline.icao']
```

In [21]:

```python
## Create the hashed column with a lambda function utilizing hash_key().
pq['hashed'] = pq.apply(lambda x: hash_key(x.key), axis = 1)
```

Create a New Column Hash_Key

In [22]:

```python
## New column will only consist of abbreviation of the key.
pq['hash_key'] = pq['hashed'].str[:1]
```

Create Table

In [23]:

```python
## Create the table with pyarrow.
table = pa.Table.from_pandas(pq)
```

Use Parquet Write_to_Dataset

In [24]:

```python
## Use write_to_dataset to generate the directory.
parq.write_to_dataset(table, root_path = partitioned_pq_file, partition_cols = ['ha
```

Show the Table in Notebook

In [25]:

```python
## Use read_table() function on the partitioned file
partitioned_table = parq.read_table(partitioned_pq_file)
print(partitioned_table)
```

```
pyarrow.Table
codeshare: bool
equipment: list<item: string>
  child 0, item: string
airline.active: bool
airline.airline_id: int64
airline.alias: string
airline.callsign: string
airline.country: string
airline.iata: string
airline.icao: string
airline.name: string
src_airport.airport_id: double
src_airport.altitude: double
src_airport.city: string
src_airport.country: string
src_airport.dst: string
src_airport.iata: string
src_airport.icao: string
src_airport.latitude: double
src_airport.longitude: double
src_airport.name: string
src_airport.source: string
src_airport.timezone: double
src_airport.type: string
src_airport.tz_id: string
dst_airport.airport_id: double
dst_airport.altitude: double
dst_airport.city: string
dst_airport.country: string
dst_airport.dst: string
dst_airport.iata: string
dst_airport.icao: string
dst_airport.latitude: double
dst_airport.longitude: double
dst_airport.name: string
dst_airport.source: string
dst_airport.timezone: double
dst_airport.type: string
dst_airport.tz_id: string
key: string
hashed: string
hash_key: dictionary<values=string, indices=int32, ordered=0>
----
codeshare: [[false,false,false,false,false,...,true,false,false,false,true],[true,
true,true,true,true,...,false,false,false,false,false],...,[true,true,true,true,tr
ue,...,false,false,false,false,false],[false,false,false,false,false,...,false,fal
se,false,false,false]]
equipment: [[["CR2"],["A81"],...,["320","321","319"],["CR9","CR2","CRK"]],[["AT
7"],["320"],...,["738"],["738"]],...,[["AT7"],["320"],...,["738"],["738"]],[["73
8"],["EM2"],...,["SF3"],["734"]]]
airline.active: [[true,false,false,true,true,...,true,true,true,true,true],[true,t
rue,true,true,true,...,true,true,true,true,true],...,[true,true,true,true,tru
e,...,true,true,true,true,true],[true,true,true,true,true,...,true,true,true,true,
true]]
airline.airline_id: [[410,1654,1654,8359,2750,...,2822,2822,2822,2822,2822],[2822,
2822,2822,2822,2822,...,4573,4573,4573,4573,4573],...,[2822,2822,2822,2822,282
2,...,4573,4573,4573,4573,4573],[4573,-1,-1,3754,3754,...,4178,4178,4178,4178,1901
6]]
airline.alias: [["ANA All Nippon Airways","SN Brussels Airlines","SN Brussels Airl
ines","nan","TACA",...,"Horizon Airlines","Horizon Airlines","Horizon Airlines","H
orizon Airlines","Horizon Airlines"],["Horizon Airlines","Horizon Airlines","Horiz
on Airlines","Horizon Airlines","Horizon Airlines",...,"Swiss European","Swiss Eur
opean","Swiss European","Swiss European","Swiss European"],...,["Horizon Airline
s","Horizon Airlines","Horizon Airlines","Horizon Airlines","Horizon Airline
s",...,"Swiss European","Swiss European","Swiss European","Swiss European","Swiss
European"],["Swiss European","\N","\N","nan","nan",...,"Qantas Airways","Qantas Ai
rways","Qantas Airways","Qantas Airways","Apache"]]
airline.callsign: [["AEROCONDOR","WHITE PELICAN","WHITE PELICAN","nan","HELVETI
C",...,"IBERIA","IBERIA","IBERIA","IBERIA","IBERIA"],["IBERIA","IBERIA","IBERI
A","IBERIA","IBERIA",...,"SUNEXPRESS","SUNEXPRESS","SUNEXPRESS","SUNEXPRESS","SUNE
XPRESS"],...,["IBERIA","IBERIA","IBERIA","IBERIA","IBERIA",...,"SUNEXPRESS","SUNEX
PRESS","SUNEXPRESS","SUNEXPRESS","SUNEXPRESS"],["SUNEXPRESS","\N","\N","NAS EXPRES
S","NAS EXPRESS",...,"REX","REX","REX","REX","APACHE"]]
airline.country: [["Portugal","Italy","Italy","Peru","Switzerland",...,"Spain","Sp
ain","Spain","Spain","Spain"],["Spain","Spain","Spain","Spain","Spain",...,"Turke
y","Turkey","Turkey","Turkey","Turkey"],...,["Spain","Spain","Spain","Spain","Spai
n",...,"Turkey","Turkey","Turkey","Turkey","Turkey"],["Turkey","\N","\N","Saudi Ar
abia","Saudi Arabia",...,"Australia","Australia","Australia","Australia","United S
tates"]]
airline.iata: [["2B","2G","2G","2I","2L",...,"IB","IB","IB","IB","IB"],["IB","I
B","IB","IB","IB",...,"XQ","XQ","XQ","XQ","XQ"],...,["IB","IB","IB","IB","I
B",...,"XQ","XQ","XQ","XQ","XQ"],["XQ","-","-","XY","XY",...,"ZL","ZL","ZL","Z
L","ZM"]]
airline.icao: [["ARD","CRG","CRG","FOF","OAW",...,"IBE","IBE","IBE","IBE","IBE"],
["IBE","IBE","IBE","IBE","IBE",...,"SXS","SXS","SXS","SXS","SXS"],...,["IBE","IB
E","IBE","IBE","IBE",...,"SXS","SXS","SXS","SXS","SXS"],["SXS","nan","nan","KN
E","KNE",...,"RXA","RXA","RXA","RXA","IWA"]]
airline.name: [["Aerocondor","Cargoitalia","Cargoitalia","Star Peru (2I)","Helveti
c Airways",...,"Iberia Airlines","Iberia Airlines","Iberia Airlines","Iberia Airli
nes","Iberia Airlines"],["Iberia Airlines","Iberia Airlines","Iberia Airlines","Ib
eria Airlines","Iberia Airlines",...,"SunExpress","SunExpress","SunExpress","SunEx
press","SunExpress"],...,["Iberia Airlines","Iberia Airlines","Iberia Airlines","I
beria Airlines","Iberia Airlines",...,"SunExpress","SunExpress","SunExpress","SunE
xpress","SunExpress"],["SunExpress","Unknown","Unknown","Nas Air","Nas Air",...,"R
egional Express","Regional Express","Regional Express","Regional Express","Apache
Air"]]
...
```