Large Scale Data Management

Instructor: Dr. Lefteris Sidirourgos, lsidir@aueb.gr, lsidir@gmail.com

Exercise for the Semester

Overview

You are requested to follow a series of steps to experience at first hand the process of setting up a large-scale DBMS and a distributed OLAP system, load data, query that data, and finally perform an analytical task.

You will have to install PySpark and the open-source column store MonetDB on your workstation or laptop with support for Python UDFs. Then you will be instructed to load data about real estate.

Prerequisites

You should be able to install and configure general software on your laptop. You should also be able to query data using SQL and write code in Python.

Some Introductory Reading Material

An overview presentation from Ying Zhang, MonetDB Solutions about MonetDB and TensorFlow XLDB-2018: Love at First Sight: MonetDB/TensorFlow

The documentation on the website of MonetDB (documentation) and the Blog will help you on almost all aspects of your interaction with MonetDB, including installation process, SQL querying, writing Python UDFS, and many more. There is always the user mailing list that you can use for extra advice and help, but try not to abuse the time of the volunteers of the list. Only post questions if you have searched the web and you have not found any answers, and make sure that your questions are detailed and informative.

Documentation for PySpark is available at https://spark.apache.org/docs/latest/api/python/.

The first easy steps to complete the assignment

Step 1a. Download and Install MonetDB

https://www.monetdb.org/Downloads

Make sure that you chose the correct distribution for your platform. For example, on 18.04 LTS (Bionic Beaver) Linux distribution the package to use can be found at https://www.monetdb.org/downloads/deb/dists/bionic/

If you are using deb-based Linux distributions, don't forget to read the instructions at https://www.monetdb.org/downloads/deb/ to setup the package sources. In general, you will find instructions for all distributions in the corresponding directories of the download repository.

If you are using Windows, you can try the following binaries. In case of trouble, don't forget to read any comments or instructions the MonetDB team provides you with.
https://www.monetdb.org/easy-setup/windows/

Make sure that you install MonetDB with support for Python3, since you will need this in a later step to implement UDFs with Python. For example, the command to install MonetDB with Python3 support in Ubuntu is:

```
$ sudo apt install monetdb-python3
```

You should be able at the end of this step to start MonetDB and perform simple tasks, such as issue SQL queries with the use of **mclient**.

You will find an SQL command overview here.

A small tutorial on using MonetDB can be found here.

Step 1b. Download, Install, and Run PySpark

For PySpark installation look at
https://spark.apache.org/docs/latest/api/python/getting_started/install.html

You need to install python3 for your OS. Then, the command to install PySpark with Python3 is:

```
$ python3 -m pip install pyspark
```

```
After this step, a PySpark library will be available and in order to use it
you will have to do the following steps.
```

```
Create a file .py and import pyspark package and all necessary pyspark
modules in python code as follows:
```

```
import pyspark
from pyspark.sql import *
from pyspark.sql.types import *
from pyspark import SparkContext, SparkConf
from pyspark.sql import *
from pyspark.sql.functions import udf
from pyspark.sql.types import *

Initialize Spark session with:

spark = SparkSession.builder.appName("zillow").getOrCreate()
```

Step 2. Loading the Data

You will be given a data file, namely:

```
zillow.csv
```

The origin of the data is from Zillow. It consists of a dataset including ~100K listings from the BOSTON, MA area in CSV format. You are requested to load these files in a MonetDB and PySpark.

To understand how to create a loader function for MonetDB make sure that you read the small tutorial found here.

For PySpark, you will load the CSV file into dataframes. To understand how to load data into Spark read the tutorial at https://sparkbyexamples.com/pyspark/pyspark-read-csv-file-into-dataframe/

Step 3. Preparing yourself for creating Python UDFs in MonetDB and PySpark

In the next step you are requested to write UDFs using Python.

Read this blog post on the MonetDB website on how to create Python UDFs in order to be prepared.

For PySpark you may use the guide at https://sparkbyexamples.com/pyspark/pyspark-udf-user-defined-function/ You may also look at pyspark_example.py which includes an implementation of an application on PySpark (including definition and execution of UDFs). This example Python file was presented and analyzed during the PySpark demonstration at 3/2/2022.

Step 3. Analysis

Analysis consists of the following operations:

- Extract number of bedrooms. You will implement a UDF that processes the `facts_and_features` column and extracts the number of bedrooms.
- Extract number of bathrooms. You will implement a UDF that processes the `facts_and_features` column and extracts the number of bathrooms.
- Extract sqft. You will implement a UDF that processes the `facts_and_features` column and extracts the sqft.
- Extract type. You will implement a UDF that processes the `title` column and returns the type of the listing (e.g. condo, house, apartment)
- Extract offer. You will implement a UDF that processes the `title` column and returns the type of offer. This can be `sale`, `rent`, `sold`, `forclose`.
- Filter out listings that are not for sale.
- Extract price. You will implement a UDF that processes the `price` column and extract the price. Prices are stored as strings in the CSV. This UDF parses the string and returns the price as an integer.
- Filter out listings with more than 10 bedrooms
- Filter out listings with price greater than 20000000 and lower than 100000
- Filter out listings that are not houses
- Calculate average price per sqft for houses for sale grouping them by the number of bedrooms.

Deliverable

You should deliver a small report explaining the schema that you used to load the data.

For MonetDB, you should deliver an SQL script file that contains all the necessary SQL queries and UDF definitions that implement the described steps. For PySpark, you should deliver a Python file including the UDFs and the Spark operations.