YCBS-257 - Data at Scale

Workshop 4

Apache Pig

General Instructions:

The purpose of this workshop is to get you started with Hadoop Analyzing tools. Here you will learn how to analyze your data using Apache Pig and Pig Latin script language.

Start your Cloudera QuickStart VM to complete the workshop.

Online resources:

https://pig.apache.org/docs/latest/

Exercise 1: WordCount Example with Pig

Wordcount is the "Hello World" for Hadoop. In this exercise you will write a simple and basic pig script for the word count problem. As input file you will use the file 5000-8.txt.

Complete the following Pig Latin script:

```
-- Load the text file
lines = LOAD '/home/cloudera/5000-8.txt' AS (line:chararray);

-- TOKENIZE splits the line into a field for each word.
-- flatten will take the collection of records returned by
-- TOKENIZE and produce a separate record for each one, calling the single
-- field in the record word.
words = FOREACH lines GENERATE FLATTEN(TOKENIZE(line)) as word;

-- Now group them together by each word.
grouped = GROUP words BY word;

-- Count them.
wordcount = FOREACH grouped GENERATE group, COUNT(words);

-- Print out the results.

DUMP wordcount;
```

Exercise 2: Analyzing Bixi data using Pig Latin

In this exercise you will load and manipulate the **Bixi** data to calculate top 5 start stations and top 5 cumulated duration for start stations.

To run Pig in local mode in order to use the **Bixi** data.

1. Open a new Terminal window
2. On the local file system (LFS).Create a new directory piglab1 under /home/cloudera/
3. Put the Bixi data files into this directory.
4. Run Apache Pig in local mode mode
5. `$ pig -x local`

- Write a Pig Latin Script to load the `Stations csv` file into a Pig relation

```
grunt> stationsRaw = Load '/home/cloudera/piglab1/Stations_2017.csv' using
PigStorage(',') as
(code:chararray,name:chararray,latitude:double,longitudeeid:double);
```

- Complete the Load statement to load the OD_2017-07.csv file into a Pig relation. (load start_date and end_date as chararray type)

```
grunt> bixi07Raw = Load ...
```

1. Remove the CSV file header from each relation using Filter BY statement

```
grunt> stations = Filter stationsRaw By not (code == 'code');

grunt> bixi07 = Filter . . .
```

2. To check the your filter result, create a relation limited to 5 rows and Dump the result to screen

```
grunt> st5 = Limit stations 5;

grunt> Dump st5 ;

grunt> bx5 = Limit . . .

grunt> Dump bx5 ;
```

3. Complete the Pig Latin script to :
   a. Print the top 5 **Count** of start_station_code and print the result as two columns: (Station name / Count)
   b. Get the top 5 cumulated duration_sec per start_station_code and print the result as two columns:
   (Station name / Total_Duration)

Exercise 3: Count the number of followers:

In this exercise we want to count the number of followers per Twitter user. The sample dataset provided **tw.txt** has the following format:

```
USER_ID \t No. of FOLLOWERs \n
```

Complete the following Pig Latin script:

```
-- Load the dataset
dataset = LOAD '/home/cloudera/pig/tw.txt' AS (id: long, fr: long);

-- check if user IDs are valid (e.g. not null) and clean the dataset
SPLIT dataset INTO good_dataset IF ....

-- organize data such that each node ID is associated to a list of
neighbors
nodes = GROUP .....

-- foreach node ID generate an output relation consisting of the node ID
and the number of "friends"
friends = FOREACH nodes ...... AS followers;

-- count the following
nodes2 = GROUP good_dataset ......

followings = FOREACH nodes2 GENERATE group, COUNT(......);

-- find the outliers (followers < 3)
outliers = FILTER friends ......

STORE friends INTO '/home/cloudera/pig/tw/';
STORE followings INTO '/home/cloudera/pig/tw/';
STORE outliers INTO '/home/cloudera/pig/tw/';
```