Assignment 3: Using Spark to Process Data

CS328 - Distributed and Cloud Computing

DDL: 23:59, May 8, 2023

1 Requirements

A dataset parking_data_sz.zip of parking lot utilization in Shenzhen is provided. The dataset is in CSV format. Each line of the data represents one particular car. The headers of the dataset are as follows:

Name	Description	Example
in_time	Time when the car goes into the parking lot	"2018-09-01 10:10:00"
out_time	Time when the car goes out of the parking lot	"2018-09-01 12:00:00"
berthage	Unique id of the parking lot	"201091"
section	Section to which the parking lot belongs	" 荔园路 (蛇口西段)"
admin_region	District of the parking lot	" 南山区"

You need to perform some analysis on this dataset using *Apache Spark*, and output the results as specified by the following five requirements to five separate CSV files.

1. Output the total number of parking lots in each section. The output file should have two columns, with the headers being **section** and **count**.

2. Output all unique ids (berthages), associated with their section. The output file should have two columns, with the headers being **berthage** and **section**.

3. Output for each section: the average parking time of a car in that section. The output file should have two columns, with the headers being **section** and **avg_parking_time**. The average parking time should be counted in seconds as an integer.

4. Output the average parking time for each parking lot, sorted in descending order. The output file should have two columns, with the headers being **berthage** and **avg_parking_time**. The average parking time should be counted in seconds as an integer.

5. Output for each section: the total number of parking lots in use ("in use" means there is at least one car in that parking lot) and the percentage out of the total number of parking lots in that section, in a 30-minute interval (e.g. during 09:00:00-09:29:59). The output file should have five columns, with the headers being `start_time`, `end_time`, `section`, `count` and `percentage`. The percentage value should be rounded to two decimal places. The data format of `start_time` and `end_time` should be "`HH:MM:SS`", e.g. 12:00:00

This assignment does not impose any requirement on the programming language and Spark API (RDD, Dataset, or DataFrames) you use. You can choose whatever is convenient for you.

2 Hints

1. Filter out invalid data (e.g., `out_time` ≤ `in_time`) in advance.
2. Some optimizations can be done, like converting the data (`in_time`, `out_time`) to (`in_time`, `parking_time_length`) before doing further computations.
3. `https://spark.apache.org/docs/latest/rdd-programming-guide.html`

3 What to Submit

1. Source code (as files)
2. A report (using a provided template) in PDF format, including:
   - A screenshot of your Spark job's DAG, using the Spark Web UI (like this). For convenience, you could use the interactive Spark shell. Running on a cluster is also encouraged and regarded as a **bonus**, but it is not compulsory. Notice that the Web UI can be accessed only when the job is running. If you choose not to use the interactive shell, a simple hack (in Java) is to add "`Thread.sleep(1000000);`" at the end of the code to avoid exiting, so that you can access the Web UI.
   - Three figures showing the results in Requirement 5, with time as X axis and percentage in use as Y axis. Select three sections from the overall results and plot the figures.
3. Five separate CSV files containing the results of the five requirements. The names of files should be: `r1.csv, r2.csv, r3.csv, r4.csv, r5.csv`.

Pack all files into `SID_NAME_A3.zip`, where `SID` is your student ID and `NAME` is your name (e.g., 11710106_ 张三 _A3.zip).