Lab1 - Apache Spark

1 Introduction

In this assignment, you are going to use Apache Spark to explore the page views of Wikimedia projects. As a dataset, we use the page view statistics generated between 0-1am on Jan 1, 2016, which is available here. Each line of the dataset, delimited by a white space and contains the statistics for one Wikimedia page. The schema looks as follows:

- **Project code:** The project identifier for each page.
- **Page title:** A string containing the title of the page.
- **Page hits:** Number of requests on the specific hour.
- **Page size:** Size of the page.

2 Installing Spark

You can do this assignment either online at Databricks Cloud or locally by intslling Spark on your machine. This section includes the steps you need to go through in order to install Spark. It is assumed that you have installed Java SDK 8.

1. Download and extract Apache Spark 2.4.3 tarball in your home directory.

```
cd ~
wget https://archive.apache.org/dist/spark/spark-2.4.3/spark-2.4.3-bin-hadoop2.7.tgz
tar -xvf spark-2.4.3-bin-hadoop2.7.tgz
```

2. Add the following lines to `.bashrc` (and then `source` it):

```
export JAVA_HOME="/path/to/the/java/folder"
export SPARK_HOME="/path/to/the/spark/folder"
export PATH=$JAVA_HOME/bin:$SPARK_HOME/bin:$PATH
```

3. Run the command `spark-shell` in a terminal. If it works, you should see something like the screenshot below:

3 Task 1 - Spark

First, create a new Spark Session and load the dataset as below:

```
val spark = SparkSession.builder().getOrCreate()

val pagecounts = sc.textFile("directory_to/pagecounts.out")
```

Then, convert the `pagecounts` from `RDD[String]` into `RDD[Log]`:

1. Create a case class called `Log` using the four field names of the dataset.
2. Create a function that takes a string, split it by white space and converts it into a log object.
3. Create a function that takes an `RDD[String]` and returns an `RDD[Log]`.

In the remaining sections of this exercise, you have to make use of the `RDD[Log]` that you have created. For each of the questions below, implement a Scala function that takes as input an `RDD[Log]` and prints the requested values. You must include all of those results in your report.

1. Retrieve the first 15 records and print out the result.
2. Determine the number of records the dataset has in total.
3. Compute the min, max, and average page size.
4. Determine the record(s) with the largest page size. If multiple records have the same size, list all of them.
5. Determine the record with the largest page size again. But now, pick the most popular.
6. Determine the record(s) with the largest page title. If multiple titles have the same length, list all of them.
7. Use the results of Question 3, and create a new RDD with the records that have greater page size than the average.
8. Compute the total number of pageviews for each project (as the schema shows, the first field of each record contains the project code).
9. Report the 10 most popular pageviews of all projects, sorted by the total number of hits.
10. Determine the number of page titles that start with the article "The". How many of those page titles are not part of the English project (Pages that are part of the English project have "en" as the first field)?
11. Determine the percentage of pages that have only received a single page view in this one hour of log data.
12. Determine the number of unique terms appearing in the page titles. Note that in page titles, terms are delimited by "_" instead of a white space. You can use any number of normalization steps (e.g., lowercasing, removal of non-alphanumeric characters).
13. Determine the most frequently occurring page title term in this dataset.

4 Task 2 - Spark SQL

First, convert the `pagecounts` from `RDD[String]` into DataFrame (hint: you may need to transform `RDD[String]` into `RDD[Log]` and then DataFrame). Next, you must use your DataFrame to answer again to questions 3, 5, 7, 12, 13 of Task 1, but this time by running SQL queries programmatically.

5 What to deliver

Create a folder and put all the `.scala` files and a report that includes all the results of the questions. Then zip it with subject `lab1_<your group name>.zip`.