CSE 4510/5310 – Mgmt & Proc Big Data

Fall 2022

Hands-on Activity 3

Apache Spark Resilient Distributed Datasets (RDDs)

Total Points: 40

Date Assigned: Friday, Oct 21, 2022 **Due Date:** Monday, Oct 24, 2022

Submission Instructions: This is an individual activity. Please submit your work on Canvas as a Jupyter Notebook `ipynb` file named `cse4510_yourname_activity3.ipynb`. **Your code must be written in pyspark.**

Key Big Data Processing Methods Demonstrated

- To use resilient distributed datasets (RDDs) to solve a problem in Apache Spark
  - To model a problem using Spark's MapReduce model
  - To implement and apply map and reduce functions on a big data problem (dataset containing over 85k movies)

1. **MapReduce on Movies** (40 points)
   Given a dataset named `IMDB_Movies.csv`, which is available on Canvas, use the Spark Resilient Distributed Dataset (RDD) model to count the number of times a country is involved in a movie. Your output will be key-pairs represented as *`[country, text-based bar-chart (count)]`* and sorted by bar-length. You may create the text-based bar-chart using an extended ASCII character such as the block character (ASCII character 219. See more ASCII characters here: https://theasciicode.com.ar/). For scaling, let 1 ASCII character = 1000 movies. Make sure to collect and display your results in your notebook using the `rdd.collect()` method.

   RDD functions Allowed:

   - sc.textFile()
   - `map()`
   - `flatMap()`
   - `reduce()`
   - `reduceByKey()`
   - `sortBy()`
   - `groupBy()`

   (i.e., Use only Spark RDD. You may also use any packages from the Python standard library. Do not use DataFames)

   Expected Output saved as Spark text files and Collected using the `collect()` method

```
('USA', '(34.33k)')
('France', '(8.31k)')
('UK', '(7.49k)')
('India', '(6.37k)')
('Italy', '(5.06k)')
('Germany', '(3.72k)')
('Canada', '(3.62k)')
('Japan', '(3.70k)')
('Spain', '(2.73k)')
('Australia', '(1.18k)')
('Denmark', '(1.03k)')
('Belgium', '(1.35k)')
('Mexico', '(1.17k)')
('China', '(1.17k)')
('Netherlands', '(1.03k)')
('South Korea', '(1.30k)')
('Sweden', '(1.23k)')
('Russia', '(1.08k)')
('Hong Kong', '(1.88k)')
```