Big Data Processing
COSC 2637/2633
Assignment 2

Assessment Type	Individual assignment. Submit online via Canvas → Assignment 2. Marks awarded for meeting requirements as closely as possible. Clarifications/updates may be made via announcements or relevant discussion forums.
Due Date	Week 12, Friday 23:59, 16 October
Marks	40

Overview

Write MapReduce and Spark programs which gives your chance to understand the complexity of MapReduce and Spark programing, the essential components you learned in lectures, the unique debugging method, the impact of performance using different size clusters.

Learning Outcomes

The key course learning outcomes are:

CLO 1. Model and implement efficient big data solutions for various application areas using appropriately selected algorithms and data structures.

CLO 2. Analyze methods and algorithms, to compare and evaluate them with respect to time and space requirements and make appropriate design choices when solving real-world problems.

CLO 3. Motivate and explain trade-offs in big data processing technique design and analysis in written and oral form.

CLO 4. Explain the Big Data Fundamentals, including the evolution of Big Data, the characteristics of Big Data and the challenges introduced.

CLO 5. Apply non-relational databases, the techniques for storing and processing large volumes of structured and unstructured data, as well as streaming data.

CLO 6. Apply the novel architectures and platforms introduced for Big data, in particular Hadoop and MapReduce.

Task 1 – Compute Co-occurrence Matrix (25 marks)

Task 1.1 - Implement both “pairs approach” and “strips approach” to compute the co-occurrence matrix where the word-pair frequency is maintained. The context of a word is defined as the words in the same line. (10 marks)

Task 1.2 - Implement both “pairs approach” and “strips approach” to compute the co-occurrence matrix where the word-pair relative frequency is maintained. The context of a word is defined as the words in the same line. Note “pairs approach” should avoid the memory bottleneck issue. (15 marks)

You should use Java to develop your MapReduce program over AWS EMR (if you want to use other code language, please contact lecturer for approval).

Marking Guide:

(a) The codes for “pair approach” and “strips approach” in the Task-1.1 and Task 1.2 are entailed in a single Maven project and built in a single standalone jar file. (1 mark)

(b) Submit the complete Maven project source code in a .zip file (including a standalone jar file). The zip file should be named as sxxxxx_BDP_A2T1_S2_2020.zip (sxxxxx is your student id). (1 mark)

(c) You need include a “README” file in the zip file. In the README, you are asked to specify how to run each approach in Task-1.1 and Task-1.2 using the standalone jar in Hadoop. (1 mark)

(d) Paths of input file and output file should not be hard-coded. (1 mark)

(e) Conduct performance analysis on different numbers of nodes in EMR clusters. To this end, run the pairs approach and strips approach in Task-1.1 and Task-1.2 to process a large data set

s3a://commoncrawl/crawl-data/CC-MAIN-2018-17/segments/1524125936833.6/wet/CC-MAIN-20180419091546-20180419111546-00036.warc.wet.gz

when the number of nodes in EMR clusters is 3, 5, 7 respectively. Show the CPU_MILLISECONDS for each MAP task and each REDUCE task in the README file; and analyze what you observed (250-500 words). (1x4 marks)

(f) Your MapReduce program(s) must be well written, using good coding style and including appropriate use of comments. (1x4 marks)

Important Marking Information

The four approaches in Task-1.1 and Task-1.2 must be correctly implemented and runnable in AWS EMR using the submitted jar file.

- If one approach in Task-1.1 cannot be run using the submitted jar file, you will lose 5 marks. If one approach can run but the output is incorrect due to
  - Major logic errors in implementation, you will lose 5 marks.
  - Minor implementation errors, you will lose 2.5 marks.
- If one approach in Task-1.2 cannot be run using the submitted jar file, you will lose 7.5 marks. If one approach can run but the output is incorrect due to
  - Major logic errors in implementation, you will lose 7.5 marks.
  - Minor implementation errors, you will lose 3 marks.

Task 2 – Spark Streaming (15 marks)

Develop code in a Scala Maven project to monitor a folder in HDFS in real time such that any new file in the folder will be processed (in this assignment, you are required to load “3littlepigs”, “Melbourne” and “RMIT” files in the folder under monitoring in sequence order; note must wait for at least 10 seconds between two files). For each RDD in the stream, the following subtasks are performed concurrently:

(a) Count the word frequency and save the output in HDFS. (5 marks)

Note, for each word, make sure space (" "), comma (","), semicolon (";"), colon (":"), period ("."), apostrophe (“’”), quotation marks (“””), exclamation (“!”), question mark (“?”), and brackets ("[", “{”, “(”, “<”,"]", “)”, “}”,”>”) are trimmed.

(b) Filter out the short words (i.e., < 5 characters) and save the output in HDFS. (5 marks)

(c) Count the co-occurrence of words in each RDD where the context is the same line; and save the output in HDFS. (5 marks)

You should use Scala to develop your MapReduce program over AWS EMR (if you want to use other code language, please contact lecturer for approval).

Marking Guide:

(a) The codes are entailed in a single Scala Maven project. (1 mark)

(b) Submit the complete project source code in a .zip file (including a standalone jar file). The zip file should be named as sxxxxx_BDP_A2T2_S2_2020.zip (replace sxxxxx by your student id). (1 mark)

(c) You need include a "README" file in the zip file. In the README, you are asked to specify how to run the Scala project using the standalone jar in Hadoop. (1 mark)

(d) The path of the folder under monitoring, and the paths of three subtask outputs should be given as command line arguments (e.g., `Input subtaskA_output subtaskB_output subtaskC_output`) (1 mark)

(e) For each subtask, all outputs for each RDD along with time must be retained. You cannot overwrite the previous outputs. (1x 3 marks)

Important Marking Information

- If one subtask cannot be run correctly using the submitted jar file, no mark for this subtask. If one subtask can run but the output is incorrect due to
  - Major logic error in implementation, no mark for this subtask, i.e., 5 marks.
  - Minor implementation errors, you will lose half of the marks for this subtask, i.e., 2.5 marks.
- All subtasks must be executed concurrently. Otherwise, you will lose 5 marks

Submission

Create a single zip file named *sxxxxx_BDP_A2_S2_2020.zip* which includes

- sxxxxx_BDP_A2T1_S2_2020.zip
- sxxxxx_BDP_A2T2_S2_2020.zip

Submit via Canvas > Assignments > Assignment 2.

Assessment declaration: when you submit work electronically, you agree to the assessment declaration: https://www.rmit.edu.au/students/student-essentials/assessment-and-exams/assessment/assessment-declaration

Academic integrity and plagiarism (standard warning)

Academic integrity is about honest presentation of your academic work. It means acknowledging the work of others while developing your own insights, knowledge and ideas. You should take extreme care that you have:

- Acknowledged words, data, diagrams, models, frameworks and/or ideas of others you have quoted (i.e. directly copied), summarized, paraphrased, discussed or mentioned in your assessment through the appropriate referencing methods,
- Provided a reference list of the publication details so your reader can locate the source if necessary. This includes material taken from Internet sites.

If you do not acknowledge the sources of your material, you may be accused of plagiarism because you have passed off the work and ideas of another person without appropriate referencing, as if they were your own.

RMIT University treats plagiarism as a very serious offence constituting misconduct. Plagiarism covers a variety of inappropriate behaviors, including:

- Failure to properly document a source
- Copyright material from the internet or databases
- Collusion between students

For further information on our policies and procedures, please refer to https://www.rmit.edu.au/students/student-essentials/rights-and-responsibilities/academic-integrity