MapReduce with Spark

Background

Recall that MapReduce is a computing paradigm facilitating the processing of large datasets.

Programming in MapReduce requires the definition and implementation of two core functions.

The first one, known as Map, transforms the input data into key-value pairs. The second one, known as Reduce, perform the actual processing according to user's/application's needs.

Task 1

Download any text dataset from the HuggingFace repository, at https://huggingface.co/datasets. We recommend the following: "openwebtext", "cnn_dailymail" , and "imdb".

Your goal is to calculate the frequencies of each individual words in the dataset. For e.g., the word "finance" occurs 10 times, "markets" 8 times,...

Part a) Without MapReduce (2 marks)

Write a simple program to complete the task without adopting the MapReduce paradigm?

Record the time taken.

Part b) With MapReduce (5 marks)

Solve the task by adopting a MapReduce paradigm. Record the time taken, and compare it with the time of part a)

Hint

- Map: what will you choose as key and value?
- Intermediate Sorting: what will you sort? How?
- Reduce: how will the actual counting be implemented?

Part c) Display

Plot the top-10 most frequent words using your preferred graphical representation (histogram, pie charts,…).

Task 2

Part a) Matrix Multiplication

Generate 2 matrices of random numbers of size 1000x1000.

Multiply the 2 matrices ii) without MapReduce, ii) with MapReduce. Record the respective computation times.

Part b) Matrix Multiplication

Generate 2 matrices of random numbers of size 100 000x100 000.

Multiply the 2 matrices ii) without MapReduce, ii) with MapReduce. Record the respective computation times.

Compare the runtime with those of part a) and comment on the scalability.

Task 3

Part a) Architecture/Infrastructure

Briefly explain the following components, which are commonly used in Hadoop or Spark environments (max 5 bullet-points/sentences per component)

- Yarn
- Resource Manager
- Node Manager

Part b) Uber Architecture

Read, understand & summarize the Big Data architecture at Uber from the following article: https://arxiv.org/pdf/2104.00087.pdf

Deliverables

- Slides for each task/parts
- Source code

Organization

- Groups of 2-3.
- Deadline: April 10th (class presentation)
- Submit a zipped folder, comprising the slides, and source code on lol@