Comparison and Simulation of External-Sort on a huge dataset

Overview

It goes without saying that sorting plays an important task in most of the computer mechanisms. Most importantly, sorting algorithms can be used to solve other problems like counting duplicates, deciding rankings, finding medians in a dataset and many other problems. Commercially talking sorting can be used for tasks like event- driven simulations, searching for some specific information, numerical computations. Various sorting algorithms are in place to handle sorting tasks for a limited size of datasets.

When the size of dataset exceeds the RAM capacity of a specific machine, external sort algorithms should be used. Example: If you want to sort a dataset of 500 GB with a machine which has 12 GB of RAM (Random Access Memory), you can't do that using basic sorting algorithms like insertion sort, selection sort. Because sometimes you need to access all the data for a specific comparison in some algorithms. You will be required to use an altogether different approach. You will be required to make modifications in the general programming approach as well as the sorting algorithm you want to use.

External sort algorithms use a combination of in-place sorting algorithms and produce a better algorithm which can be utilized to sort a huge dataset with limited requirement of machine memory. Because of the memory constraint, the dataset must be broken down into chunks of manageable sizes like 9-10 GB which the machine can handle. Then each chunk of data is sorted using a suitable algorithm, so that the chunks created earlier are sorted. Then these sorted chunks of data are further broken down so that one portion from each chunk can be merged by machine to give a final sorted dataset.

Goals

Compare among various external sorting algorithms based on various sort benchmarks and simulate an algorithm which can sort a dataset of 20TB with a machine which has 12GB of RAM.

Deliverables

- Report on the comparative analysis of various external sort algorithms.
- Ideally, a program in Python, Java or C programming language where one of the external sort algorithms is accurately simluated in comparison to internal sort.

Knowledge Areas Needed for Project

- Various aspects of sorting algorithms
- Programming skills in Python, Java or C

Team Members: Hitesh Kumar Dasika, Jayendra Khandare

Task Division For Jayendra Khandare:

Writing a report on the workings of some external sorting algorithms and comparing their complexities.

Picking one external sort algorithm and simulating it by comparing it with an internal sort algorithm.