COEN 242 Big Data

Programming Assignment -1

Due Date: 04/21/2019 11:59 PM

K Most Popular Words

Goal

There are multiple factors that affect the performance of an application. The purpose of this assignment is to analyze such factors -- input size (or load), algorithms efficiency, data structures, system resource utilization (e.g., CPU, Memory), and so on. The assignment will give you an opportunity to understand the impact of the above factors on the performance. And also we'll get to know how the size of input dataset impacts performance and what are the techniques we should use to alleviate the problems introduced by increased input dataset size (crux of Big Data).

Input Dataset

We have provided three datasets of different sizes. The dataset is available here.
The dataset is a zip file that consists of three files with only English words.
1. data_1GB.txt – A text file of size 1GB.
2. data_8GB.txt – A text file of size 8GB.
3. data_32GB.txt – A text file of size 32GB.

The Problem Statement

Design and Implement an efficient java code to determine 100 most frequent/repeated words in a given dataset. The objective here is to obtain the result with the least possible execution Time (or with the best performance on your computer).

Execute your code on each of the three input data files separately. It is a good practice to execute your code initially on 1 GB dataset and then repeat on larger datasets.

This is an example screenshot of the result that is expected in the report.

```
Total Execution time is 11 s
The top 100 elements are
Word                    Frequency
the                     2465303
quot                    1668463
and                     1181273
ref                     840972
title                   692798
text                    515749
page                    469278
The                     465726
for                     450428
amp                     433487
User                    418369
was                     414969
of                      403197
name                    362929
talk                    360321
http                    325345
format                  318581
that                    316390
com                     310836
sha                     299727
model                   299577
```

Analyze the performance through different metrics such as running time, speedup, CPU utilization, memory usage, etc.

Present detailed analysis for why you use a particular algorithm or a particular data structure to solve this problem.

NOTE:

- The result should preserve case sensitivity i.e. words “Title” and “title” are considered as two different words.
- The input dataset contains only english alphabets and white spaces i.e. “a-z”,”A-Z”,”\s”.

Deliverables

NOTE: All these files should be in a SINGLE tar/zip file uploaded on Camino

1. Source Code (Java files, optional scripts for preprocessing if needed) (30 Marks)
Note: Code should be documented and commented. Good coding practices will get some extra points.

2. PDF report file containing the analysis of the results.

The Report should contain the following analysis:

- Code and Correct Output of your code: (10 points)
  Screenshot of the result containing the execution time and the top 100 words.

- Algorithm & Data Structures & Design: (15 points)
  Justify the algorithm and overall design used in your code. Data Structures: Reason for choosing the data structures and their advantages w.r.t the problem size/complexity.

- Presenting Performance Data: (10 points)
  This is the part you can show your skills of data visualization and data analysis. You can use a wide variety of ways to present data. For example, you can plot graphs of speed up vs execution parameters for each data set. Explain your observations with proper reasoning.

- Good Coding Practices: (5 points)

Please describe your system specification: List out your laptop configuration – CPU cores, memory, etc.

Extra points will be awarded for detailed and holistic analysis or out of the box thinking.

NOTE:
For JVM tuning refer https://docs.oracle.com/cd/E22289_01/html/821-1274/configuring-the-default-jvm-and-java-arguments.html