TU59 & TU60 - Programming for Big Data - Hadoop/Map-Reduce Assignment (40%)

Due Before :
Monday 27th February @23:00

Overview

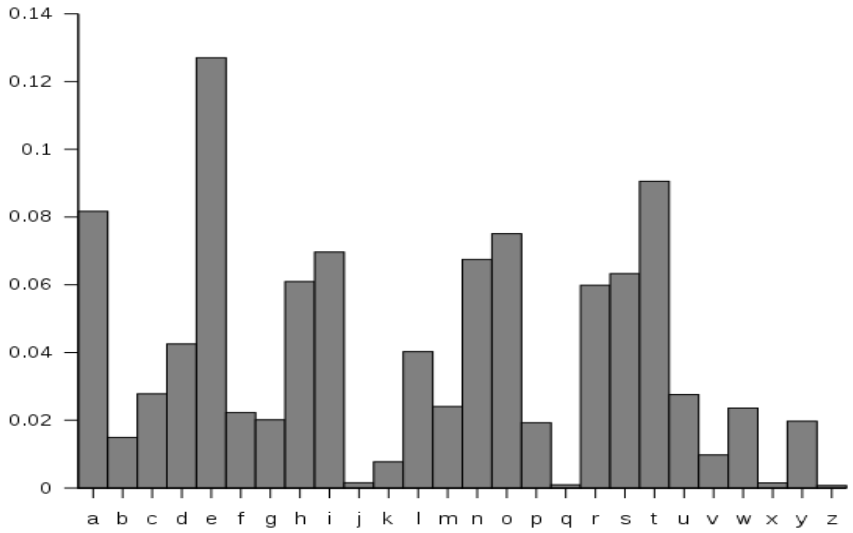
This assignment requires you to compile a set of data, load this data into *HDFS* and write a MapReduce process to efficiently extract and aggregate data, expanding the functionality of the MapReduce process and present the data as outlined in the following sections.

Background

Analysis of letter frequency in text has been used in a variety of different areas, including encryption, word puzzle games, even the television show The Wheel of Fortune. Linotype machines which were used in printing Morse code and even the design of keyboard layouts are all based on letter frequencies.

There is no exact letter frequency distribution for a language as all writers write differently and the distribution also depends on the subject under discussion in the text. Scientific texts, press reporting, religious texts, general fiction will all display slightly different letter frequency distributions. Accurate average letter frequencies can only be gathered by analyzing a large amount of representative text.

The graph below shows the average letter frequencies for the English language derived from an analysis of entries in the concise Oxford dictionary.



Required Tasks

The task of this assignment is use *HDFS* and *MapReduce* to calculate average letter frequencies across a number of European languages (at least 3 languages) using the books available in ***Project Guttenburg*** (http://www.gutenberg.org/). You will need to write a *Map-Reduce* process to perform the necessary calculations, utilizing as many features as possible. The outputs (which can contain several files) should contain the **language, letter and number**.

To complete this work using *Map-Reduce*, you will need to use the full set of features, **including chained processes**.

Important: Only **one** Map-Reduce process (including chained processes) should be used to perform all necessary tasks. No hard coding of languages, directories, etc. should be included in your code. You many need to consider how the input data files are organized. The input files may contain additional characters etc which you may need to ignore.

TU59 & TU60 - Programming for Big Data - Hadoop/Map-Reduce Assignment (40%)

Due Before :
Monday 27th February @23:00

Using Python or R (e.g. ggplot2), create charts to compare the letter frequencies across the various languages. The files produced by the map-reduce process should be used as the input data sets for the analysis and charts. **No additional** aggregations or other data processing should be performed. The Python or R code should only display the resulting outputs from the map-reduce process.

Deliverables

You will be required to document your approach for processing the data and producing the required outputs using Map-Reduce. The analysis, charts and code used for this task should also be included.

Your report (saved as a PDF document) should contain the following:

- Explanation of the steps you performed for loading the data sets into HDFS.
- Detaild design, including diagrams and detailed explanations of each part of the Map-Reduce process. This should not be an explanation of how MapReduce works but instead focuses on how your solutions works.
- Explanations of any design decisions (evaluating alternatives), including any assumptions made.
- Clear explanations of each Map-Reduce featured used in your solution.
- Well written and fully commented Java code for the map-reduce process.
- Output files from the map-reduce process illustrating the data produced at each stage and for at least 3 languages.
- Advanced Map-Reduce functions.
- R/Python code used to load the data sets and create the comparison charts. You should provide some commentary for each of the charts included in your report.

The output files from the map-reduce process should be included. If these are not included then your assignment mark will be reduced by 30%. These should be submitted separate to the PDF report.

Submission Details

You should create **one document/report** containing **all** the material for each item listed in the deliverables. **Convert this document into a PDF**. It is this PDF document that should be submitted.
All images should be imbedded in this document. This PDF document should contain all your Map-Reduce code and the R/Python code used to produce the analysis. Any additional files can be zipped and submitted alongside the PDF

In addition to the report include:

- Signed Assignment Cover Sheet.
- all Map-Reduce code.
- the output files from the map-reduce. You will need to extract these files from HDFS.
- R/Python code to analyse the output files.

The Report, Output Files and all code should be ZIPPED (**only zip format will be accepted**) and it is this ZIP file that should be submitted.

You will need to submit your assignment on Brightspace before the deadline. Make sure that you submit your assignment files to the correct assignment on BrightSpace. You should receive an acknowledgment from Brightspace when you have submitted your assignment. You should keep a copy of this acknowledgement.

TU59 & TU60 - Programming for Big Data - Hadoop/Map-Reduce Assignment (40%)

Due Before :
Monday 27th February @23:00

Assignment Update Information

Minor Updates/Clarifications will be posted in the Q&A section of the website for this part of module. Check it regularly for any updates.

Other more important updates will be posted on the BrightSpace module – Page/Information for this Assignment. Check it regularly for any updates.

Marking Scheme

The marking scheme for this assignment is:

- 5% Explanation of the steps you performed for loading the data sets into HDFS.
- 20% Design and structure of the map-reduce process.
- 30% Well written and fully commented Java code for the map-reduce process. This will be judged on quality, scalability and efficiency.
- 30% Extent of use of advanced map-reduce features and scalability. Provide detailed description, explanations on why and how they are used, and fully commented Java code
- 5% Output files from the map-reduce process.
- 10% Evaluation, commentary, comparison charts and R/Python code.

This assignment accounts for **40%** of your module mark for Programming for Big Data module.

The documentation for your assignment must clear contain the following details

- Full Name
- Student Number
- Programme Code (e.g. TU59, TU60, etc)
- Stream (ASD, DS, PhD, etc)
- Year of Programme (1st Year, 2nd Year)
- Module Name

Failure to give this information will incur a 10% penalty.

Important Information

The assignment most be performed **individually.**

Each submission must be original work as **plagiarism** will result in a **zero** mark (0%).

The output files from the map-reduce process should be included. If these are not included then your assignment mark will be reduced by 30%.

There will be a 10% penalty deduction applied for each day the assignment is late.
There is no penalty for submitting early.

TUDublin **Plagiarism Policy** : https://tudublin.libguides.com/c.php?g=674049&p=4794713
https://www.tudublinsu.ie/advice/exams/breachesofregulations/

Assignment Feedback

Feedback will be via Brightspace, where a mark for your assignment will be given and a short comment on the assignment. I will endeavor to have these returned within two weeks.