School of Computer Science

COMP30770

Project 1
CLI (Bash) & Data Management for Big Data

Teaching Assistant:	Thomas Laurent
Coordinator:	Anthony Ventresque
Date:	Friday 29th January, 2021
Total Number of Pages:	4

General Instructions

- You are encouraged to collaborate with your peers on this project, but all written work must be your own. In particular we expect you to be able to explain every aspect of your solutions if asked.
- We ask you to hand in an archive (zip or tar.gz) of your solution: code/scripts, README.txt file describing how to run your programs, a short pdf report of your work.
- The report should list your answers and should also contain a short introduction and conclusion. The report should also contain (see Section 3 of this document):
  - a section on why CLI is a good way to approach big data (think blog posts, research papers if any, etc.) - a good paragraph is enough for this.
  - a section discussing the difference between relational and non relational (NoSQL) systems (1 page max)
  - a short (1 or 2 pages) description of one of the research papers you can find on Brightspace.
- The report should not be longer than 10 pages.
- The breakdown of marks for the project will be as follows:
  - Exercise 1: 35%
  - Exercise 2: 35%
  - Exercise 3: 30%
- **Due date: 14/02/2021**

1 Cleaning a Dataset with Bash

The first task of most data science projects consists in cleaning the datasets to limit the otherwise overwhelming datasets to a set of clean and potentially meaningful variables. In this first exercise, you will work with a snapshot of Reddit posts and you will clean its content in order to be able to process it further - e.g., finding trends in the posts. The "raw" dataset is available from Brightspace and here. You can open the csv file with some external programs (e.g., Excel) to have a look at the structure. However, we will only use Bash and the terminal in this project.

1. The first three columns contain index variables that can safely be removed. Column 34 contains a flag to see if the data is safe for work or not. As we only include safe for work data in this dataset we can drop this column too. Remove these columns with cut and an option (check man cut).

2. Some columns are completely empty, in that all cells in the column are empty. Identify and remove those columns (using a script).

   One idea is to go through all the columns using a for loop (get the number of columns and go through them from the last one to the first one), and run a cut command to get the content of this column. In another, nested, for loop go throught the lines in the result of the cut command (think command substitution) and test that each of the cells is not empty. If one of the cells is not empty then you can keep this column. Otherwise delete it as you did in the previous question.

3. Some variables are uninformative because they take only one value. Identify and remove those variables (using a script). Use a similar logic to the one we used in the previous question - but check that at least two values are different instead of checking that all values are not null.

4. Two variables, created_utc and retrieved_on are times represented in seconds since epoch (January 1, 1970). Convert those variables to a month, e.g. January, February, March, etc. (see the man page for the bash command date, in particular the "example" section)

5. How many posts have been have been made in each month?

6. The variable title contains the title of each post, stored as text strings. For this data to be useful for subsequent regression and classification analyses data scientists need to perform a few cleaning steps. Implement these steps using bash scripts to produce a cleaned dataset.

   (a) Convert all letters to lowercase. (check what the command tr does - check man tr).

   (b) remove the punctuation (man tr again)

   (c) remove the stop words (i.e., the words with little or no real meaning); use a list of stop words from the web (e.g., one of these lists) and a sed command (in a script).

   (d) reduce the words to their stem words using a basic stemming technique (using a dictionary, such as, this one).

Notes

- How to read a file line by line:

```
#!/bin/bash

input="$1"
while IFS= read -r var
do
        echo $var
done < "$input"
```

2 Data Management

In this second exercise, you're asked to upload the dataset in the two database management systems we have studied: MySQL and MongoDB.

1. Create a database in MySQL with the following tables:
   - user(author_id,author, author_cakeday) where author_id is the primary key
   - subreddit(subreddit) where subreddit is the primary key
   - post(id, author_id, subreddit, created_month) where id is the primary key, author_id references the author table, subreddit references the subreddit table, and created_month contains the data from 1.4 obtained from created_utc.

   Explain how you created such a database
2. Populate the database using a Bash script
3. Implement the following queries in SQL and explain how your queries work:
   (a) List of all author names.
   (b) List of all posts' title with their author's name and the subreddit they were posted in.
   (c) List of (subreddit, month) pairs, and the number of posts made in the subreddit during this month.
4. Create a MongoDB collection with every line in the CSV file as an entry/document. Populate the collection with the content of the CSV file (using a script for instance).
5. Write the same queries you used on your mysql database for your mongo database. Give the queries in your report, explain what they do.

Now we want to modify the structure of our database, and in particular we want to add multiple subreddits per post.

For each database system, answer the following questions:

1. Do you need to alter the previous records stored in the database in order to apply the new structure?
2. Are the previous queries that you used for the first versions of the databases still working?

3 Reflection

In this section, you are asked to

1. give your opinion on the suitability of CLI (Bash) for some Big Data tasks. A paragraph is enough for this question.

2. compare relational and NoSQL database management models and in particular give your impressions on why one is better than the other and in which context (there could be some overlap with the previous question but what we expect here is a more general discussion on the topic). You can use some external content (papers, blogs, etc.) to support some of your ideas. This section should not be more than a page.

3. write a short report (1 or two pages) on one of the research papers that are available on Brighspace. They are all different (one is older and pre-date the NoSQL era as such, one is less formal etc.) but they all contain some important information about the NoSQL world. The following list of sections is an indication of how to write your paper. Some of the items might not be relevant for all papers, and you might want to add some sections in your report (e.g., evaluate the posterity of a solution etc.). We will have an open mind when reading your report and we just want to see how you analyse a research paper and are able to discuss it - in short there is no one single perfect report, everything that shows you made an effort to understand and focus on the important parts (research methodology, hypotheses, etc.) will be welcome.

    - identify the question/challenge the paper addresses. Explain in your own words what the motivation for the research is.
    - describe briefly the related work, i.e., the other (related) solutions that the authors compare themselves to. Show the limitations of these related solutions
    - Give an outline of the solution proposed by the authors (no need to go into details) showing the main components
    - describe their scientific method: what are the research questions they evaluate, how do they evaluate
    - describe briefly their results
    - give your impression on the idea, what you liked about the paper and whether you see any limitations etc.