COMP9313 2017s2 Project 3

Problem 1 (8 pts): Find the top-*k* terms that appear in the most lines

Given a large text file, each term is contained in several lines. Your task is to find the top-*k* terms that appear in the most lines.

- Ignore the letter case, i.e., consider all words as lower case.
- Ignore terms starting with non-alphabetical characters, i.e., only consider terms starting with “a” to “z”.
- Use the following split function to split the documents into terms:

```
split("[\\s*$&#/\"'\\,.:;?!\\[\\](){}<>~\\-_]+")
```

You can use the text file pg100.txt (available at: http://www.gutenberg.org/cache/epub/100/pg100.txt) as the sample input.

Output format:

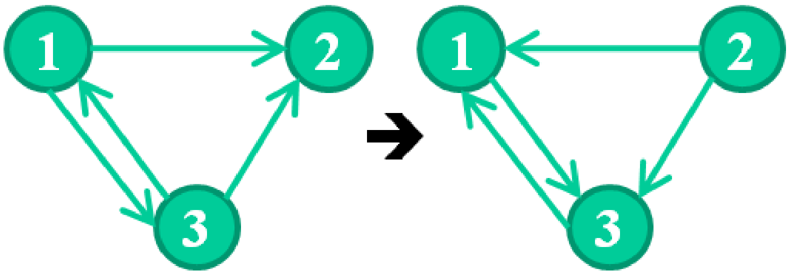
Your Spark program should generate a list of ***k*** key-value pairs, where the keys are the terms, the values are the number of lines containing the term, and keys and values are separated by “\t”. Sort the key-value pairs first according to the values in descending order, and then according to the keys in alphabetical order.

Code format:

Name your scala file as “Problem1.scala”, the object as “Problem1”, and put it in package “comp9313.ass3”. Store the final result in a text file on disk. Your program should take three parameters: the input text file, the output folder, and the value of ***k***.

Problem 2 (12 pts): Reverse graph edge direction

Given a directed graph, reverse the direction of all edges.



Input files:

In the input file, each line contains a pair of node ids: “FromNodeId\tToNodeId”. In the above example, the input contains four lines: “1\t2”, “1\t3”, “3\t1”, “3\t2”.

The sample file “tiny-web-Stanford.txt” can be downloaded at: https://webcms3.cse.unsw.edu.au/COMP9313/17s2/resources/12579

Output format:

The output is the adjacency list of the reversed graph, and the nodes are sorted in ascending order in each list. Format each line as: “NodeId\tNeighbor$_1$, Neighbor$_2$, …, Neighbor$_m$”, using only one comma to separate the node IDs in the list.

Given the above example, the output file contains three lines: “1\t3”, “2\t1, 3”, “3\t1”.

Code format

Name your scala file as “Problem2.scala”, the object as “Problem2”, and put it in package “comp9313.ass3”. Store the final result in a text file on disk. Your program should take two parameters: the input graph file and the output folder.

Documentation and code readability

Your source code will be inspected and marked based on readability and ease of understanding. The documentation (comments of the codes) in your source code is also important. Below is an indicative marking scheme:

Result correctness: 90%
Code structure, Readability, and Documentation: 10%

Submission:

Deadline: Sun 1st Oct 21:59:59

Log in any CSE server (williams or wagner), and use the give command below to submit your solutions:

```
$ give cs9313 assignment3 Problem1.scala Problem2.scala
```

Or you can submit through:
https://cgi.cse.unsw.edu.au/~give/Student/give.php

If you submit your assignment more than once, the last submission will replace the previous one. To prove successful submission, please take a screenshot as assignment submission instructions show and keep it by yourself.

Late submission penalty

10% reduction of your marks for the 1st day, 30% reduction/day for the following days.

Plagiarism:

The work you submit must be your own work. Submission of work partially or completely derived from any other person or jointly written with any other person is not permitted. The penalties for such an offence may include negative marks, automatic failure of the course and possibly other academic discipline. Assignment submissions will be examined manually.

Relevant scholarship authorities will be informed if students holding scholarships are involved in an incident of plagiarism or other misconduct.

Do not provide or show your assignment work to any other person - apart from the teaching staff of this subject. If you knowingly provide or show your assignment work to another person for any reason, and work derived from it is submitted you may be penalized, even if the work was submitted without your knowledge or consent.