COMP9313 2017s2 Project 1

Problem statement:

Given a text file, compute the average length of words starting with each letter. This means that for every letter, you need to compute: the total length of all words that start with that letter divided by the total number of words that start with that letter.

- 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”.
- The length of the term is obtained by the length() function of String. E.g., the length of “text234sdf” is 10.
- Use the tokenizer below to split the documents into terms.
  StringTokenizer itr = new StringTokenizer(value.toString(),
  " *$&#/\t\n\f\'"\\,.:;?{}<>~-_");
- You do not need to configure the numbers of mappers and reducers. Default values will be used.

Input files:

Text documents

Output format:

Your MapReduce job should generate a list of key-value pairs, and ranked in alphabetical order, like (this example is only used to show the format):

a 3.5613186813186815

b 4.3849323131253675

… …

… …

z 7.909090909090909

The average length is of double precision (use DoubleWritable).

The sample input and output can be downloaded at:
https://webcms3.cse.unsw.edu.au/COMP9313/17s2/resources/12027

Your tasks:

You are required to write TWO versions of MapReduce program to solve this problem. The first version utilizes a combiner, and the second version utilizes the "in-mapper combining" approach to improve the efficiency.

Write each version in a single java file (like WordCount.java used in Lab 2).

Name your first version as "WordAvgLen1.java", and the second version as "WordAvgLen2.java", and put them in the package "comp9313.ass1".

Compile:

Your java code will be compiled and packaged as a jar file, and we will use the following commands to check the correctness of your solution:

```
$ $HADOOP_HOME/bin/hadoop jar YOURJAR.jar YOURCLASS input output

$ $HADOOP_HOME/bin/hdfs dfs -cat output/*
```

Please ensure that the code you submit can be compiled and packaged. Any solution that has compilation errors will receive no more than 3 points for the entire assignment. Your solution will be compiled by Java 1.7 and tested based on Hadoop-2.7.2.

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.

Marking

This assignment is worth 10 points. Below is an indicative marking scheme:

Result correctness:	7
Code structure and readability and necessary comments:	3

Submission:

Deadline: Sunday 27th August 09:59:59 pm

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

$ give cs9313 assignment1 WordAvgLen1.java WordAvgLen2.java

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. If you have any problems in submissions, please email to xuefeng.chen@student.unsw.edu.au.

Late submission penalty

You will receive zero marks for this assignment.

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.