COMP9313 22T3 Project 2 (16 marks)

Problem statement:

Detecting popular and trending topics from the news articles is an important task for public opinion monitoring. In this project, your task is to perform text data analysis over a dataset of Australian news from ABC (Australian Broadcasting Corporation) using both **RDD** and **DataFrame** APIs of Spark with Python. The problem is to compute the weights of each term regarding each year in the news articles dataset and then select the top-k most important terms in each year.

Input files:

The dataset you are going to use contains data of news headlines published over several years. In this text file, each line is a headline of a news article, in format of "date,term1 term2 ... ... ". The date and texts are separated by a comma, and the terms are separated by the space character. A sample file is like below:

```
20030219,council chief executive fails to secure position
20030219,council welcomes ambulance levy decision
20030219,council welcomes insurance breakthrough
20030219,fed opp to re introduce national insurance
20040501,cowboys survive eels comeback
20040501,cowboys withstand eels fightback
20040502,castro vows cuban socialism to survive bush
20200401,coronanomics things learnt about how coronavirus economy
20200401,coronavirus at home test kits selling in the chinese community
20200401,coronavirus campbell remess streams bear making classes
20201015,coronavirus pacific economy foriegn aid china
20201016,china builds pig apartment blocks to guard against swine flu
```

This small sample file can be downloaded at:

https://webcms3.cse.unsw.edu.au/COMP9313/22T3/resources/81094

Term weights computation:

You need to ignore the stop words such as “to”, “the”, and “in”. There is also a stop word list stored in the file:

https://webcms3.cse.unsw.edu.au/COMP9313/22T3/resources/82213

To compute the weight for a term regarding a year, please use the TF/IDF model. Specifically, the TF and IDF can be computed as:

TF(term t, year y) = the number of headlines containing t in y

IDF(term t, dataset D) = $\log_{10}$ (the number of years in D/the number of years having t)

Finally, the term weight of term t regarding the year y is computed as:

Weight(term t, year y, dataset D) = TF(term t, year y)* IDF(term t, dataset D)

Please import math and use math.log10() to compute the term weights, and round the results to 6 decimal places.

Output format:

If there are N years in the dataset, you should output exactly N lines in your final output file, and these lines are sorted by years in ascending order. In each line, you need to output a list of k pairs in format of <term, weight>, and these pairs are sorted by term weights in descending order. If two terms have the same weight, sort them alphabetically. Specifically, the format of each line is like: "year\t $Term_1$,$Weight_1$;$Term_2$,$Weight_2$;… …;$Term_k$,$Weight_k$". For example, given the above data set and k=3, the output should be:

```
2003    council,1.431364;insurance,0.954243;welcomes,0.954243
2004    cowboys,0.954243;eels,0.954243;survive,0.954243
2020    coronavirus,1.908485;china,0.954243;economy,0.954243
```

Code format:

Please name your two python files as "project2_rdd.py" and "project2_df.py" for using RDD and DataFrame APIs, respectively. Compress it in a package named "zID_proj2.zip" (e.g. z5123456_proj2.zip).

Command of running your code:

We will use the following command to run your code:

```
$ spark-submit project2_rdd.py input output stopwords k
```

In this command, `input` is the input file, `output` is the output folder, `stopwords` is the stop words file, and `k` is the number of terms returned for each year.

Please ensure that the code you submit can be compiled. Any solution that has compilation errors will receive no more than 5 points.

Marking Criteria:

Your source code will be inspected and marked based on readability and ease of understanding. Each solution has 8 marks. Below is an indicative marking scheme:

Result correctness: 6
Efficiency and memory usage: 1
Code structure, Readability, and Documentation: 1

Submission:

Deadline: Monday 31st Oct 11:59:59 PM

You can submit through Moodle:

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 siqing.li@unsw.edu.au.

Late submission penalty

5% reduction of your marks for up to 5 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.