CSC 4760/6760 Big Data Programming

Assignment 2

Due Date: 11:59 pm, Wednesday, February 20, 2019

1. (100 points) (Computing PageRank in Spark)

Dataset:

The toy dataset is the following graph. The PageRank values are already known. We can use it to check your program.

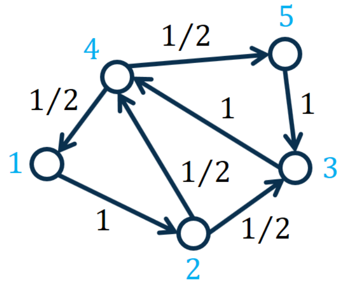

Figure 1: A toy graph for computing PageRank. The number on the edge represents the transition probability from one node to another.

The PageRank values are given in the following table (given that the decay factor $c = 0.85$):

Nodes	PageRank Values
1	0.1556
2	0.1622
3	0.2312
4	0.2955
5	0.1556

PageRank:

Compute the PageRank value of each node in the graph. Please refer to the slides for more details about the PageRank method. The key PageRank equation is as follows.

$$\mathbf{r} = c\mathbf{P}^\top\mathbf{r} + (1-c)\mathbf{1}/n$$

where $\mathbf{r}$ represents the $n \times 1$ PageRank vector with each element $\mathbf{r}_i$ representing the PageRank value of node i, n represents the number of nodes in the graph, $\mathbf{P}$ represents the $n \times n$ transition probability matrix with each element $\mathbf{P}_{i,j} = p_{i,j} = \frac{1}{d_i}$ representing the transition probability from node i to node j, d_i represents the degree of node i, $\mathbf{P}^\top$ represents the transpose of $\mathbf{P}$, $c \in (0,1)$ represents a decay factor, $\mathbf{1}$ represents a $n \times 1$ vector of all 1's, and n represents the number of nodes in the graph.

Please see the slides for more details.

In this assignment, we set the decay factor $c = 0.85$ and set the number of iterations to 30.

Implementation:

Design and implement a PySpark program to compute the PageRank values. A template “PageRank_Spark_Incomplete.py” file is given. You need to add 6 lambda functions in the file.

For example:

Line 13:   AdjList2 = AdjList1.map(lambda line : line) # 1. Replace the lambda function with yours

You need to replace “lambda line : line” with your own lambda function. The inputs to the lambda function should be not changed.

The outputs in the terminal of the ground-truth solutions is given in the file “TerminalOutputs.txt”. You may use it to understand the source code, debug your code, and verify your solution.

Example command to run the “.py” file:

```
$ spark-submit  PageRank_Spark_Incomplete.py
```

The files can be put in local file system.

Report:

Please write a report illustrating your experiments. You need to explain your basic idea about how to design the computing algorithm. You may add comments to the source code such that the source code can be read and understood by the graders.

In the report, you should include the answers to the following questions.

1) Explanation of the source code:

What are the functions of your lambda functions? Which kind of intermediate results are generated?

2) Experimental Results

2.1) Screenshots of the key steps. For example, the screenshot for the outputs in the terminal when you run the command. It will demonstrate that your program has no bug.

2.2) Explain your results. Does your implementation give the exact PageRank values?

Submission Materials:

a) Your report

b) Source code (.py file)

c) The outputs in the terminal (Intermediate results)

d) The output file of your program (PageRank values)