Illinois Institute of Technology

Department of Computer Science

CS430 - Section: 04 (live)

Introduction to Algorithms

IIT Assignment 4 - Report

Authors:	*Student Number:*
Bastien LEDUC	A20520860
Ashik SYED SHAFFIULLAH	A20516459
Kachikwu NWIKE	A20517226
Hongbo WANG	A20524770

December 3, 2022

Instructions

The objective of HW4 is to investigate the PageRank Algorithm proposed by Sergey Brin and Larry Page, the founders of Google. Before you start working on this assignment **watch this video** **and read the report (1)**. **Terminology and notation** used in the problem statement below (e.g. hyperlink graph, row-normalized hyperlink matrix, dangling node, Google matrix, damping factor, PageRank row vector) **have been introduced in (1)**.

1. Consider a 10-page mini web represented by a directed hyperlink graph (denoted as **W**) whose topology is specified by the following sequence of numbers (determined by the prime powers):
   13, 16, 17, 19, 23, 25, 27, 29, 31, 32, 37, 41, 43, 47, 49, 53, 59, 61, 64, 67, 71, 73, 79, 81, 83, 89, 97, 101, 103, 107, 109
   that define starting and ending vertices of all directed edges, e.g. 13 means that there is a hyperlink from page P_1 to page P_3 , 47 - a hyperlink from P_4 to P_7 , 107 - a hyperlink from P_{10} to P_7 , etc. (for a 3-digit number the first 2 digits represent a starting vertex).

   (a) **(1 points)** Find the row-normalized hyperlink matrix H of this graph. Are there any dangling nodes in this hyperlink graph?

   (b) **(2 points)** Find the Google matrix G of this graph. Assume the damping factor $d = 0.85$.

   (c) **(2 points)** You are **NOT** allowed to use the formula for the number of iterations (given in (1), page 15, item 2b) needed to obtain numerical values of the PageRank row vector v. Propose an alternative termination criterion for this iterative algorithm.

   (d) **(4 points)** Find iteratively numerical values of all elements of the PageRank row vector $v = [r(P_1), r(P_2), \ldots, r(P_{10})]$. **Use your termination criterion proposed in item (c)**. How many iterations do you need to obtain your numerical results? Assume the damping factor $d = 0.85$ and $v(0) = [0.1, 0.1, 0.1, \ldots, 0.1]$.
   Implement this iterative algorithm by yourself (absolutely no Excel or program libraries).

   (e) **(2 points)** Plot the evolution of numerical values of $r(P_5)$ and $r(P_7)$ for subsequent iteration steps of the algorithm for:

   (e.1) $d = 0.55$ and $\mathbf{v}^{(\mathbf{0})} = [0.1, 0.1, 0.1, \ldots, 0.1]$

   (e.2) $d = 0.85$ and $\mathbf{v}^{(\mathbf{0})} = [0.1, 0.1, 0.1, \ldots, 0.1]$

(f) **(2 points)** Assume $d = 0.85$ and run your algorithm for three different initial PageRank row vectors $v_1^{(0)}, v_2^{(0)}$ and $v_3^{(0)}$. What are the final numerical values of v_1, v_2 and v_3 ? Are they identical or different?

(g) **(1 points)** Compare the numerical values of the PageRank row vector v for two different values of the damping factor $d = 0.85$ and $d = 1$. What is the basic difference between them? Explain it.

Submit source codes of all the programs you need to produce your results.

Solutions

1. (a) We start by fisrt drawing the hyperlink table of our directed graph:

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**
P1	0	0	1	0	0	1	1	0	1	0
P2	0	0	1	0	1	0	1	0	1	0
P3	1	1	0	0	0	0	1	0	0	0
P4	1	0	1	0	0	0	1	0	1	0
P5	0	0	1	0	0	0	0	0	1	0
P6	1	0	0	1	0	0	1	0	0	0
P7	1	0	1	0	0	0	0	0	1	0
P8	1	0	1	0	0	0	0	0	1	0
P9	0	0	0	0	0	0	1	0	0	0
P10	1	0	1	0	0	0	1	0	1	0

Table 1: The hyperlink table of our directed graph W.

From there, we can write the corresponding hyperlink matrix A from which we can obtain the row-normalised hyperlink matrix H of W.

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \Longrightarrow H = \begin{pmatrix} 0 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \end{pmatrix}$$

We can see that there are no dangling nodes in this graph W as there are no rows with only zeroes in the matrices.

(b) We denote by G the **Google matrix** of the graph W as follows:

$$G = dH + (1-d)\left[\frac{1}{10}\right]_{10\times 10} \quad \text{where damping factor } d = 0.85,$$

$$= 0.85\begin{pmatrix} 0 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 \end{pmatrix} + (1-0.85)\begin{pmatrix} \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{3}{200} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{3}{200} & \frac{91}{400} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} \\ \frac{3}{200} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} \\ \frac{179}{600} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} \\ \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} \\ \frac{3}{200} & \frac{3}{200} & \frac{11}{25} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{11}{25} & \frac{3}{200} \\ \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} \\ \frac{179}{600} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} \\ \frac{179}{600} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{179}{600} & \frac{3}{200} \\ \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{173}{200} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} \\ \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{3}{200} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} & \frac{91}{400} & \frac{3}{200} \end{pmatrix}$$

(c) G is a Stochastic matrix since it is a square matrix whose rows are probability vectors and therefore, sums up to one. Hence, the Perron–Frobenius Theorem tells us that there is a non-negative eigenvector v such that $v = v \cdot G$. In our work, we start with the initial vector $v^{(0)} = [0.1, 0.1, 0.1, \ldots, 0.1]$ and multiply with G to get $v^{(1)}$. Doing this iteratively, we have the following:

$$\begin{aligned}
v^{(1)} &= v^{(0)} \cdot G \\
v^{(2)} &= v^{(1)} \cdot G \\
v^{(3)} &= v^{(2)} \cdot G \\
v^{(4)} &= v^{(3)} \cdot G \\
&\vdots \\
v^{(n)} &= v^{(n-1)} \cdot G
\end{aligned}$$

where n is the current iteration.

The termination criterion for this iteration is when $v^{(n)} = v^{(n-1)}$ as implied by the Perron–Frobenius Theorem. In our algorithm, for a given decimal accuracy of m digits, our algorithm will stop when each element in $v^{(n)}$ is equal to their respective element in $v^{(n+1)}$ up to the m^{th} digit.

(d) By using the algorithm in **(c)** above with a damping factor $d = 0.85$ and $v^{(0)} = [0.1, 0.1, 0.1, \ldots, 0.1]$, we obtain the following table:

Digits of accuracy	**Number of iterations**
1	4
2	7
3	10
4	13
5	15
6	18
7	20
8	22
9	26
10	29
11	32
12	35
13	38
14	40
15	42
16	45
17	46
18	46
19	46
20	46

Table 2: Calculation of PageRank scores using our algorithm.

The final positive vector v that we are converging to has the following shape:

$$v = [0.1719495, 0.0634558, 0.1710204, 0.0296028, 0.0284844, 0.0515393,\\ 0.2829276, 0.015, 0.1710204, 0.015]$$

(e) Plot the evolution of numerical values of $r(P_5)$ and $r(P_7)$ for subsequent iteration steps of the algorithm for:

(e.1) $d = 0.55$ and $\mathbf{v}^{(0)} = [0.1, 0.1, 0.1, \ldots, 0.1]$

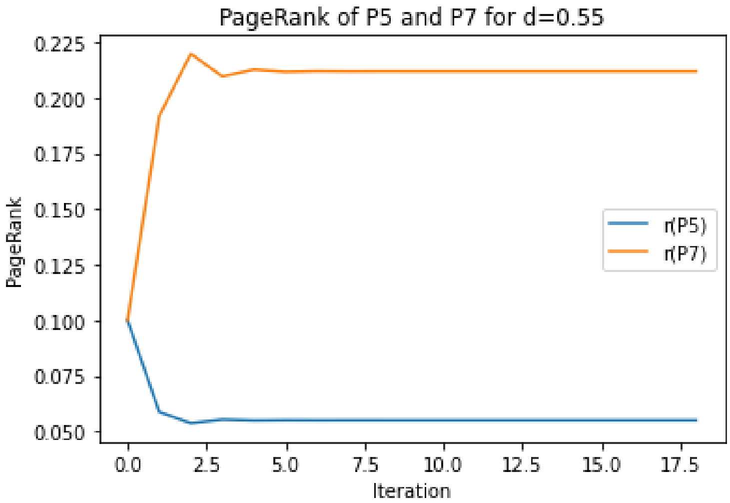

Figure 1: PageRank evolution of P_5 and P_7 for $d = 0.55$.

(e.2) $d = 0.85$ and $\mathbf{v}^{(0)} = [0.1, 0.1, 0.1, \ldots, 0.1]$

Figure 2: PageRank evolution of P_5 and P_7 for $d = 0.85$.

(f) The values of the final vectors v_1, v_2, and v_3 are closely similar even though their initial values $v_1^{(0)}$, $v_2^{(0)}$, and $v_3^{(0)}$ are different. This is because the algorithm is deterministic and the only thing that changes is the initial values of the vectors. The algorithm will always converge to the same values.

(g) In the paper (1), it is mentioned that:

"[...] from time to time, the web surfer Webster will become bored with following hyperlinks, and he will request a completely random web page. Once there, he will continue following hyperlinks until he becomes bored again."

The damping factor, d, is a parameter that represents proportion of the time that Webster continues to follow hyperlinks. On the other hand, the proportion of the time that he changes to another randomly chosen page is $1-d$.

Here we plot the final Page Rank vector v for different values of d (0.85 and 1) with a precision of $m = 8$ digits of accuracy.

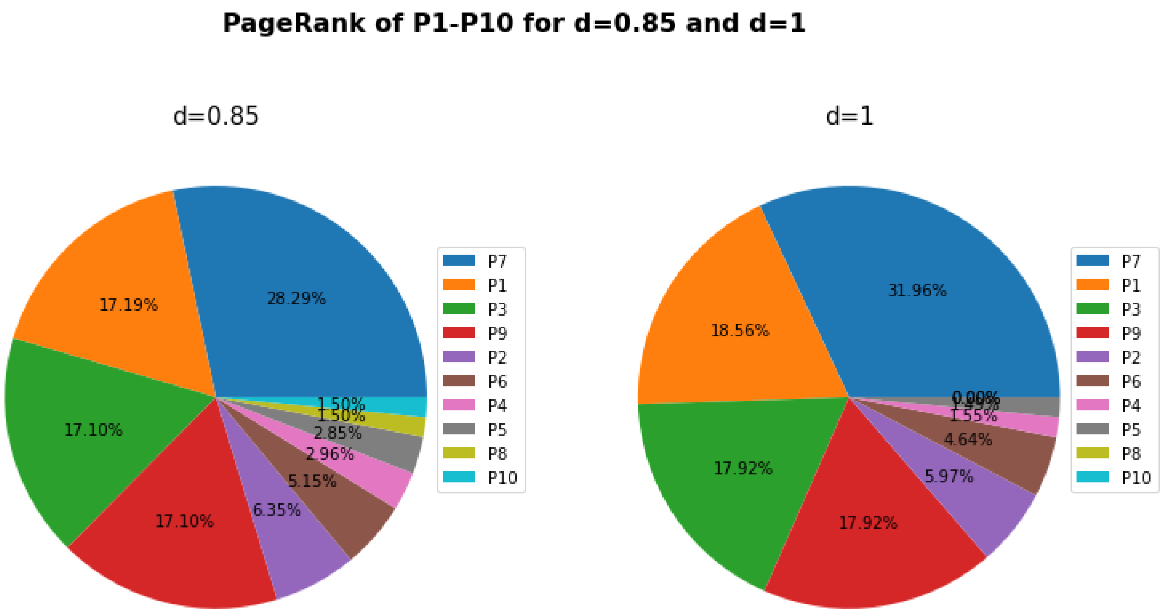

Figure 3: PageRank of v for different values of d.

We observe in this figure 3 that:

- The PageRank vector converges much quicker for $d = 0.85$ (22 iterations) than for $d = 1$ (30 iterations)(see Jupyter notebook for more details).
- For pages with a high rank (here, the top 4 pages: P_7, P_3, P_1, P_9), their rank values increased, while for the low rank pages $(P_2, P_4, P_5, P_6, P_8, P_{10})$, their rank values

decreased when d went from 0.85 to 1. (Note: PageRank values of P_8 and P_{10} are reduced to 0).

We can conclude that a high damping factor will have a more clearly defined page rank vector but it will need more iterations to reach it. Moreover, it will also keep the balance between the rank values, meaning that increasing a page rank value will lead to the decrease of another page rank value in order to keep the vector as a probability distribution (sum of all values equals to 1). This can lead to a variation in the final page rank affected by the change in d.

Team contribution

Bastien: (25%) coding parts, algorithm, report, damping factor explanation.

Ashik: (25%) matrix computation, algorithm, report.

Kachikwu: (25%) algorithm description and stopping rule.

Hongbo: (25%) coding parts, algorithms, report.

References

[1] Davey B.A, *Google PageRank*, Australian Mathematical Sciences Institute, University of Melbourne, 2013, https://www.amsi.org.au/teacher_modules/pdfs/Maths_delivers/Pagerank5.pdf