Project Report I

Information Retrieval

Anuj Kulkarni (anuj@ccs.neu.edu)

10/20/2013

Contents

Problem

PageRank

In this project, you will compute PageRank on a collection of 183,811 web documents. Consider the version of PageRank described in class. PageRank can be computed iteratively as show in the following pseudocode:

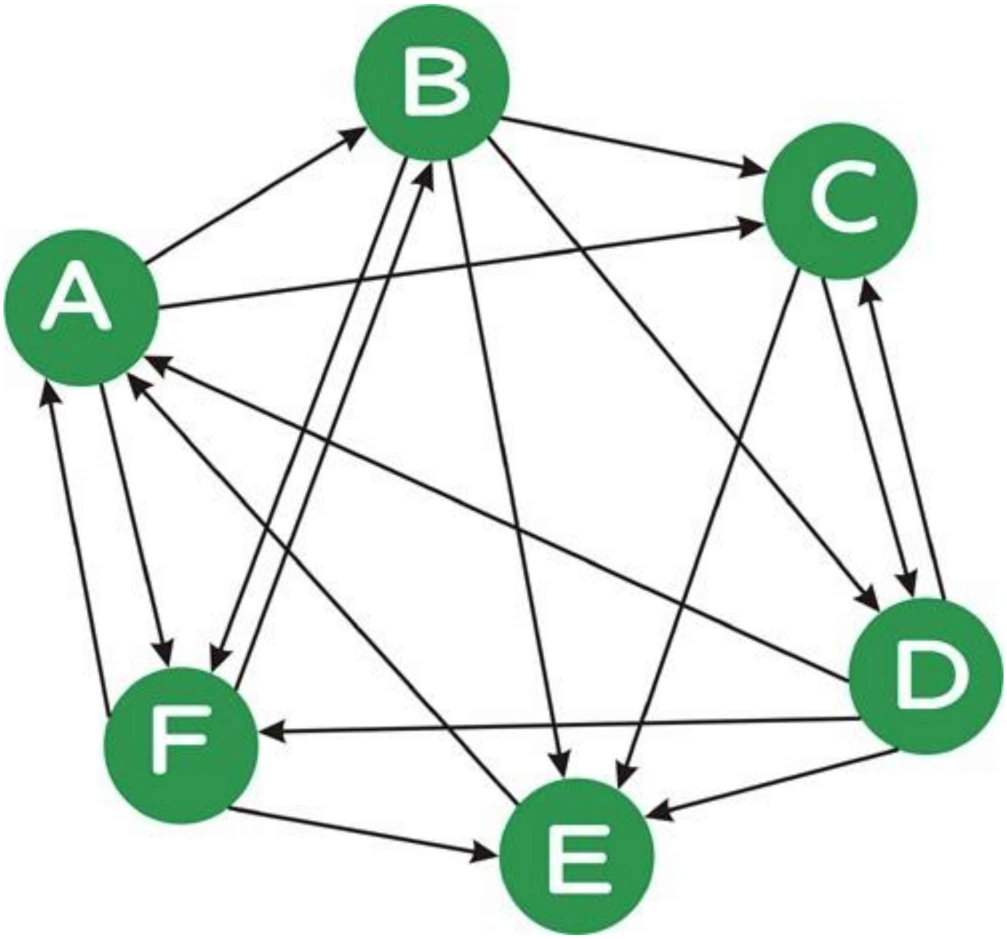
```
// P is the set of all pages; |P| = N
// S is the set of sink nodes, i.e., pages that have no out links
// M(p) is the set of pages that link to page p
// L(q) is the number of out-links from page q
// d is the PageRank damping/teleportation factor; use d = 0.85 as is typical
foreach page p in P
  PR(p) = 1/N                          /* initial value */

while PageRank has not converged do
  sinkPR = 0
  foreach page p in S                  /* calculate total sink PR */
    sinkPR += PR(p)
  foreach page p in P
    newPR(p) = (1-d)/N                 /* teleportation */
    newPR(p) += d*sinkPR/N                /* spread remaining sink PR evenly */
    foreach page q in M(p)             /* pages pointing to p */
      newPR(p) += d*PR(q)/L(q)            /* add share of PageRank from in-links */
  foreach page p
    PR(p) = newPR(p)

return PR
```

In order to facilitate the computation of PageRank using the above pseudocode, one would ideally have access to an in-link respresentation of the web graph, i.e., for each page p, a list of the pages q that link to p.

Consider the following directed graph:



We can represent this graph as follows:

A D E F

B A F

C A B D

D B C

E B C D F

F A B D

where the first line indicates that page A is linked from pages D, E, and F, and so on. Note that, unlike this example, in a real web graph, not every page will have in-links, nor will every page have out-links.

`

Deliverable Part I

Implement the iterative PageRank algorithm as described above. Test your code on the six-node example using the input representation given above. Be sure that your code handles pages that have no in-links or out-links properly. (You may wish to test on a few such examples.) In later parts of this project, your task will be easier if you don't require loading the entire link graph into memory.

To hand in: List the PageRank values you obtain for each of the six vertices after 1, 10, and 100 iterations of the PageRank algorithm.

Solution

The give graph structure:

```
A D E F
B A F
C A B D
D B C
E B C D F
F A B D
```

Now, after Running PageRank algorithm after 1, 10, 100 iterations we get these values at

After **1** iteration of PageRank algorithm we get these values at each vertex:

```
A 0.249305555555556
E 0.213888888888889
F 0.143055555555556
C 0.143055555555556
D 0.131250000000000
B 0.119444444444444
```

After **10** iterations of PageRank algorithm we get these values at each vertex:

```
A 0.252036376028172
E 0.187106610142917
F 0.151293765934751
C 0.151293765934751
D 0.131930650918251
B 0.118962972776899
```

After **100** iterations of PageRank algorithm we get these values at each vertex:

```
A 0.252127105375196
E 0.187104590699931
F 0.151306489866705
C 0.151306489866705
D 0.139306185318539
B 0.118907822573539
```

Deliverable Part II

Download the in-links file for the WT2g collection, a 2GB crawl of a subset of the web. This in-links file is in the format described above, with the destination followed by a list of source documents.

Run your iterative version of PageRank algorithm until your PageRank values "converge". To test for convergence, calculate the perplexity of the PageRank distribution, where perplexity is simply 2 raised to the (Shannon) entropy of the PageRank distribution, i.e., 2H(PR). Perplexity is a measure of how "skewed" a distribution is --- the more "skewed" (i.e., less uniform) a distribution is, the lower its preplexity. Informally, you can think of perplexity as measuring the number of elements that have a "reasonably large" probability weight; technically, the perplexity of a distribution with entropy h is the number of elements n such that a uniform distribution over n elements would also have entropy h. (Hence, both distributions would be equally "unpredictable".)

Run your iterative PageRank algorithm, outputting the perplexity of your PageRank distribution until the perplexity value no longer changes in the units position for at least four iterations. (The units position is the position just to the left of the decimal point.)

For debugging purposes, here are the first five perplexity values that you should obtain (roughly, up to numerical instability):

183811, 79669.9, 86267.7, 72260.4, 75132.4

To hand in: List the perplexity values you obtain in each round until convergence as described above.

Solution

```
Run 1: Perplexity - 183810.9999981843
Run 2: Perplexity - 79669.92319572593
Run 3: Perplexity - 86267.67410235935
Run 4: Perplexity - 72260.35360673653
Run 5: Perplexity - 75132.40765928668
Run 6: Perplexity - 68932.60291313533
Run 7: Perplexity - 71197.83341084897
Run 8: Perplexity - 67782.5377845268
Run 9: Perplexity - 69379.57741406372
Run 10: Perplexity - 67383.70755882388
Run 11: Perplexity - 68477.80188342396
Run 12: Perplexity - 67207.1847962372
Run 13: Perplexity - 68004.15388363267
Run 14: Perplexity - 67138.95537950807
Run 15: Perplexity - 67708.25939079146
Run 16: Perplexity - 67131.6639346713
Run 17: Perplexity - 67524.47691364767
Run 18: Perplexity - 67132.11109104136
Run 19: Perplexity - 67413.7101218584
Run 20: Perplexity - 67138.84981449715
Run 21: Perplexity - 67339.82543896341
Run 22: Perplexity - 67149.78500617412
Run 23: Perplexity - 67290.83065799328
Run 24: Perplexity - 67158.76207907012
Run 25: Perplexity - 67259.2257454245
Run 26: Perplexity - 67166.0293806807
Run 27: Perplexity - 67237.78022614133
Run 28: Perplexity - 67172.32050731525
Run 29: Perplexity - 67223.12743901908
Run 30: Perplexity - 67177.14437293068
Run 31: Perplexity - 67213.31945184508
Run 32: Perplexity - 67180.75113810251
Run 33: Perplexity - 67206.59757743869
Run 34: Perplexity - 67183.5491211406
Run 35: Perplexity - 67201.93310322228
Run 36: Perplexity - 67185.6323888267
Run 37: Perplexity - 67198.74209316482
Run 38: Perplexity - 67187.15939556045
Run 39: Perplexity - 67196.52571858228
Run 40: Perplexity - 67188.29833357994
Run 41: Perplexity - 67194.97937612096
Run 42: Perplexity - 67189.13216054361
Run 43: Perplexity - 67193.90362973847
Run 44: Perplexity - 67189.74037799244
Run 45: Perplexity - 67193.15073830378
Run 46: Perplexity - 67190.18473573797
Run 47: Perplexity - 67192.62137402293
Run 48: Perplexity - 67190.50788330668
Run 49: Perplexity - 67192.25035506621
Run 50: Perplexity - 67190.7423803229
Run 51: Perplexity - 67191.98879607832
Run 52: Perplexity - 67190.91290192968
Run 53: Perplexity - 67191.80442692328
Run 54: Perplexity - 67191.03608703314
Run 55: Perplexity - 67191.67433608793
Run 56: Perplexity - 67191.12532992226
```

Perplexity at Run 56: 67191.12532992226

Deliverable Part III

Sort the collection of web pages by the PageRank values you obtain.

To hand in: List the document IDs of the top 50 pages as sorted by PageRank, together with their PageRank values. Also, list the document IDs of the top 50 pages by in-link count, together with their in-link counts.

Solution

Top 50 pages sorted by PageRank

1 - WT21-B37-76 - 0.0026794094272144403
2 - WT21-B37-75 - 0.001525916643842787
3 - WT25-B39-116 - 0.0014694947334659239
4 - WT23-B21-53 - 0.0013723234635210242
5 - WT24-B40-171 - 0.00124499876031047
6 - WT23-B39-340 - 0.0012403968885748439
7 - WT23-B37-134 - 0.0012052153871083646
8 - WT08-B18-400 - 0.0011435407139305813
9 - WT13-B06-284 - 0.0011247805165849765
10 - WT24-B26-46 - 0.0010850456648765572
11 - WT13-B06-273 - 0.0010447001198702268
12 - WT01-B18-225 - 9.884436204738712E-4
13 - WT04-B27-720 - 9.364071908723442E-4
14 - WT23-B19-156 - 8.942304358025227E-4
15 - WT04-B30-12 - 8.164407175334276E-4
16 - WT24-B26-10 - 8.074275567873451E-4
17 - WT25-B15-307 - 8.04382203274152E-4
18 - WT07-B18-256 - 7.748821192033014E-4
19 - WT24-B26-2 - 7.713413346801215E-4
20 - WT14-B03-220 - 7.163920205376217E-4
21 - WT24-B40-167 - 7.074602423228856E-4
22 - WT14-B03-227 - 6.849553116296637E-4
23 - WT18-B31-240 - 6.601893167221362E-4
24 - WT04-B40-202 - 6.587031058942313E-4
25 - WT08-B19-222 - 6.434323149586122E-4
26 - WT27-B28-203 - 6.270012895766556E-4
27 - WT13-B15-160 - 6.212964933148186E-4
28 - WT13-B39-295 - 6.16960339048202E-4
29 - WT12-B30-56 - 6.022901125241008E-4
30 - WT10-B02-288 - 5.759932478475162E-4
31 - WT22-B38-403 - 5.745718155700847E-4
32 - WT14-B36-337 - 5.582571655384609E-4
33 - WT27-B34-57 - 5.555000124224854E-4
34 - WT23-B20-363 - 5.50861767428842E-4
35 - WT23-B01-40 - 5.504212166519096E-4
36 - WT27-B32-30 - 5.497707226706548E-4
37 - WT21-B40-37 - 5.481798551452625E-4
38 - WT21-B35-155 - 5.400525166054623E-4
39 - WT08-B08-60 - 5.356363695971233E-4
40 - WT04-B22-268 - 5.327403705638611E-4
41 - WT14-B02-400 - 5.325133188917705E-4
42 - WT18-B14-66 - 5.320578148059866E-4

```
43 - WT23-B27-31 - 5.256157925210381E-4
44 - WT23-B38-120 - 5.208891081529664E-4
45 - WT06-B35-151 - 5.201038862675114E-4
46 - WT06-B14-69 - 5.190670214570925E-4
47 - WT06-B35-161 - 5.182091589489328E-4
48 - WT10-B33-300 - 5.168066057564509E-4
49 - WT14-B36-335 - 5.154604149900399E-4
50 - WT14-B36-336 - 5.154604149900399E-4
```

Top 50 pages sorted by In Link count

```
1 - WT21-B37-76 - 2568
2 - WT18-B29-37 - 2269
3 - WT01-B18-225 - 2260
4 - WT23-B27-29 - 1940
5 - WT21-B37-75 - 1704
6 - WT27-B34-57 - 1257
7 - WT27-B32-30 - 1255
8 - WT08-B19-222 - 1041
9 - WT08-B18-400 - 1011
10 - WT10-B36-88 - 946
11 - WT10-B36-90 - 943
12 - WT10-B36-103 - 939
13 - WT10-B36-89 - 896
14 - WT21-B40-447 - 779
15 - WT18-B28-345 - 728
16 - WT12-B40-248 - 686
17 - WT24-B26-2 - 625
18 - WT25-B15-307 - 614
19 - WT27-B28-203 - 598
20 - WT18-B40-82 - 576
21 - WT21-B37-71 - 560
22 - WT22-B38-403 - 544
23 - WT08-B01-173 - 539
24 - WT13-B15-160 - 484
25 - WT23-B30-88 - 478
26 - WT18-B29-36 - 477
27 - WT27-B28-177 - 470
28 - WT13-B06-273 - 454
29 - WT13-B06-284 - 454
30 - WT07-B02-55 - 449
31 - WT13-B39-295 - 443
32 - WT17-B34-499 - 442
33 - WT17-B34-500 - 436
34 - WT24-B04-192 - 430
35 - WT14-B36-337 - 417
36 - WT17-B34-505 - 410
37 - WT10-B33-300 - 409
38 - WT23-B19-156 - 406
39 - WT17-B34-503 - 402
40 - WT23-B31-215 - 402
41 - WT23-B23-51 - 400
```

```
42 - WT08-B11-28 - 396
43 - WT23-B12-215 - 388
44 - WT23-B01-107 - 384
45 - WT23-B30-105 - 380
46 - WT17-B34-506 - 376
47 - WT17-B34-504 - 374
48 - WT17-B34-498 - 374
49 - WT14-B36-323 - 373
50 - WT07-B23-234 - 371
```

Deliverable Part IV

Examine the top 10 pages by PageRank and by in-link count in the Lemur web interface to the collection by using the "e=docID" option with database "d=0", which is the index of the WT2g collection. For example, the link

http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT04-B22-268

It will bring up document WT04-B22-268, which is an article on the Comprehensive Test Ban Treaty.

To hand in: Speculate why these documents have high PageRank values, i.e., why is it that these particular pages are linked to by (possibly) many other pages with (possibly) high PageRank values. Are all of these documents ones that users would likely want to see in response to an appropriate query? Which one are and which ones are not? For those that are not "interesting" documents, why might they have high PageRank values? How do the pages with high PageRank compare to the pages with many in-links? In short, give an analysis of the PageRank results you obtain.

Solution

Analysis of Top 10 pages obtained by PageRank algorithm:

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT21-B37-76
  Page : The Economist Homepage
  Analysis :
  - It is an interesting link as it's a homepage of a famous newspaper which has large number of high quality incoming links.
  - Also it has a number of inlinks which determine the page rank. Also outlink of page ranked 2nd is pointing to this page thus, it also transfers page rank of the 2nd ranked page.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT21-B37-75
  Page : The Economist : Copyright Notice
  Analysis :
  - It has an outbound link to the homepage of the newspaper website.
  - It is not an interesting page but links to trusted and high quality websites with high page rank helped in improving its page rank.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT25-B39-116
  Page : Security Assurance Requirements
  Analysis :
  - It is an interesting webpage with high domain period.
  - So it shows that it is relevant to a lot of user. It increased its page rank.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT23-B21-53
  Page : Homepage of Sea Exploring Web Development Team
  Analysis :
  - This page does not contain relevant information for every user but for only specific users.
  - Blogs and forum with backlinks have increased the page rank of this page.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT24-B40-171
  Page : The Evening News by CHCH
  Analysis :
    - This page gives daily evening news to its users.
    - This page contains lot of self-inlinks as well as links from popular and good websites.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT23-B39-340
  Page : STREETLINK financial reports
  Analysis :
    - This page contains financial reports about several companies.
    - Backlinks from the companies increases this page's page rank.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT23-B37-134
  Page : Copyright Information page of Health Department of WA
  Analysis :
    - It contains many inlinks from health department of WA and health related websites.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT08-B18-400
  Page : General Disclaimer page of TD Bank of Canada
  Analysis :
    - High page relevance with links from many finance and banking related websites.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT13-B06-284
  Page : L&LA Homepage
  Analysis :
    - It has higher page rank because it has many inlinks from other government websites.

- **Link :** http://fiji4.ccs.neu.edu/~zerg/lemurcgi_IRclass/lemur.cgi?d=0&e=WT24-B26-46
  Page : Huang Milton's Homepage
  Analysis :
    - This is a webpage of a famous psychiatrist and of a public US university.
    - It has many inlinks which increases its page rank.

Interesting pages by PageRank: 1, 3, 4, 5, 6, 10

1. The Economist Homepage
2. Security Assurance Requirements
3. STREETLINK financial reports
4. The Evening News by CHCH
5. Huang Milton's Homepage

Analysis:

- As we see, pages having high user traffic, high quality linking pages are having high page rank.
- Age of the domain also increases the page rank of the web page.
- Importance of web link on your webpage
- Age of your domain
- HTML Anchor text of a link
- Relevance between web pages

Non-interesting pages by PageRank: 2, 4, 7, 8, 9

1. The Economist: Copyright Notice
2. Homepage of Sea Exploring Web Development Team
3. Copyright Information page of Health Department of WA
4. General Disclaimer page of TD Bank of Canada
5. L&LA Homepage

Analysis:

- Outbound links to the higher ranked pages or the homepage helped the non-interesting pages improve their page ranks.
- Links to other trusted websites having high page ranks help non-interesting pages get higher page rank than they actually deserve.
- Blogs and Comments on a particular page even though it is non-relevant to all users (e.g. Homepage of Sea Exploring Web Development Team) will increase its Page Rank by use of back links.

Overall Analysis (Things that determine the page rank):

- High user traffic
- Age of the domain
- Link from a higher ranked pages

Source Code and Running instructions

The link to source code is as given: - PageRankImpl.java

The inLink file given in the assignment: WT2g.txt

The instructions to run the page algorithm code are as follows:

- Java JDK >= 1.5
- Just compile the code by javac PageRankImpl.java
- Now you can run the code as it has compiled by just passing the file path of the inLink formatted file. Eg. inLink file given in the assignment.
- Run the code by java PageRankImpl [the_path_of_the_file_in_inLink_format]
  e. g. java PageRankImpl [file_path]