PageRank

[You voted for me and who else? How popular are you?]

••

University of Southern California

USC Viterbi
School of Engineering

USC

Overview

- Background on Citation Analysis
- Google’s PageRank Algorithm
- Simplified Explanation of PageRank
- Examples of how PageRank algorithm works
- Observations about PageRank algorithm
- Importance of PageRank
- Kleinberg’s HITS Algorithm

••

University of Southern California

USC Viterbi School of Engineering

History of Link Analysis

- **Bibliometrics has been active since at least the 1960's**

- **A definition from Wikipedia:**
- **"Bibliometrics** is a set of methods to quantitatively analyze academic literature. Citation analysis and content analysis are commonly used **bibliometric** methods. While **bibliometric** methods are most often used in the field of library and information science, **bibliometrics** have wide applications in other areas."

- Many research fields use **bibliometric** methods to explore
  - the impact of their field,
  - the impact of a set of researchers, or
  - the impact of a particular paper.

Bibliometrics

- **One common technique of Bibliometrics is *citation analysis***
- **Citation analysis** is the examination of the frequency, patterns, and graphs of citations in articles and books.
- citation analysis can observe links to other works or other researchers.
- **Bibliographic coupling:** two papers that cite many of the same papers
- **Co-citation:** two papers that were cited by many of the same papers
- **Impact factor (of a journal):** frequency with which the average article in a journal has been cited in a particular year or period

http://citeseerx.ist.psu.edu/stats/citations

Top Ten Most Cited Articles in CS Literature

CiteSeer^{x} is a search engine for academic papers

CiteSeer^{x} 10M

Most Cited Articles | Most Cited Citations | Most Cited Authors | Venue Impact Ratings

Most Cited Computer Science Citations

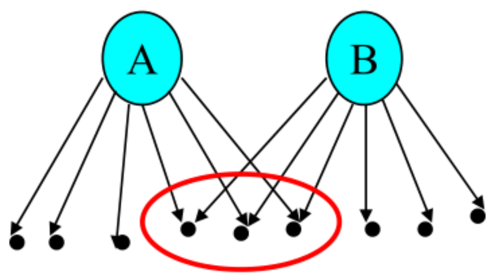
This list is generated from documents in the CiteSeer^{x} database as of March 19, 2015. This list is automatically generated and may contain errors. The list mode and citation counts may differ from those currently in the CiteSeer^{x} database, since the database is continuously updated.

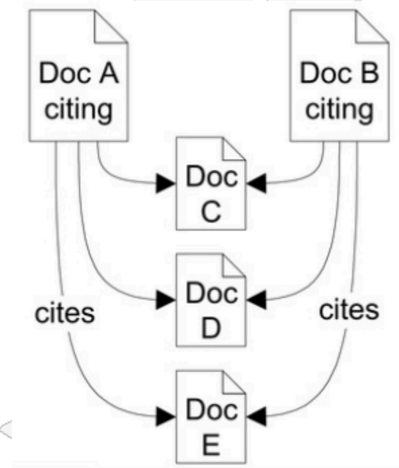
All Years | 1990 | 1991 | 1992 | 1993 | 1994 | 1995 | 1996 | 1997 | 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 | 2010 | | 2015

1. M R Garey, D S Johnson
   Computers and Intractability: A Guide to the Theory of NPCompleteness" W.H. Feeman and 1979
   11468
2. J Sambrook, E F Fritsch, T Maniatis
   Molecular Cloning: A Laboratory Manual, Vol. 1, 2nd edn Nucleic Acids Research, 1989
   10362
3. V Vapnik
   Statistical Learning Theory. 1998
   9898
4. T M Cover, J A Thomas
   Elements of Information Theory Series in Telecommunications, 1991
   9198
5. U K Laemmli
   Cleavage of structural proteins during the assembly of the head of bacteriophage T4. Nature 227:680–685 1970
   9092
6. T H Cormen, C E Leiserson, R L Rivest, C Stein
   Introduction to Algorithms. 1990
   9039
7. A P Dempster, N M Laird, D B Rubin
   Maximum likelihood from incomplete data via the EM algorithm. 1977
   8999
8. D E Goldberg
   Genetic Algorithms in Search, Optimization and Machine Learning, 1989
   8261
9. J Pearl
   Probabilistic Reasoning in Intelligent Systems 1988
   7473
10. C E Shannon, W Weaver
    The Mathematical Theory of Communication 1949
    7077

Bibliographic Coupling

- Measure of similarity of documents introduced by Kessler of MIT in 1963.
- The bibliographic coupling of two documents *A* and *B* is the number of documents cited by *both A* and *B*.
- Size of the intersection of their bibliographies.
- Maybe want to normalize by size of bibliographies?





USC Viterbi School of Engineering

Journal Impact Factor

- Developed by Garfield in 1972 to measure the importance (quality, influence) of scientific journals.
- Measure of how often papers in the journal are cited by other scientists.
- Computed and published annually by the Institute for Scientific Information (ISI).
  - It is now owned by Thomson Reuters
- The *impact factor* of a journal *J* in year *Y* is the average number of citations (from indexed documents published in year *Y*) to a paper published in *J* in year *Y*–1 or *Y*–2.
- Does not account for the quality of the citing article
- https://www.scimagojr.com/journalrank.php?area=1700&category=1702

University of Southern California

USC

USC Viterbi
School of Engineering

Top Journals for Computer Science

http://www.guide2research.com/journals/

Over all Computer Science

Top Journals for Computer Science and Electronics

Guide2Res... Like Page

Ranking is based on **Impact Factor**. **Vanity press** and poor-quality journals are not listed

All Categories | All Publishers | Search by keyword

Rank	Publisher	Journal Details	Impact Factor
1	IEEE	IEEE Communications Surveys and Tutorials ISSN:1553-877X , Quarterly	9.220
2	IEEE	IEEE Transactions on Fuzzy Systems ISSN:1063-6706 , Bimonthly	6.701
3	IEEE	IEEE Signal Processing Magazine ISSN:1053-5888 , Bimonthly	6.671
4	IEEE	IEEE Transactions on Industrial Electronics ISSN:0278-0046 , Monthly	6.383
5	Mary Ann Liebert	Soft Robotics ISSN:2169-5172 , Quarterly	6.130
6	World Scientific	International Journal of Neural Systems ISSN:0129-0657 , Bimonthly	6.085
7	IEEE	IEEE Transactions on Pattern Analysis and Machine Intelligence ISSN:0162-8828 , Monthly	6.077
8	IEEE	IEEE Transactions on Evolutionary Computation ISSN:1089-778X , Bimonthly	5.908
9	ELSEVIER	Remote Sensing of Environment ISSN:0034-4257 , Monthly	5.881
10	OXFORD University Press	Bioinformatics ISSN:1367-4803 , Semimonthly	5.766

Software Engineering

Top Journals for Computer Science and Electronics

Guide2Res... Like Page

Ranking is based on **Impact Factor**. **Vanity press** and poor-quality journals are not listed

Software Engineering | All Publishers | Search by keyword

Rank	Publisher	Journal Details	Impact Factor
90	WILEY	INFORMATION SYSTEMS JOURNAL ISSN:1350-1917 , Bimonthly	2.522
113	IEEE	IEEE Transactions on Reliability ISSN:0018-9529 , Quarterly	2.287
147	Springer	Business and Information Systems Engineering ISSN:1867-0202 , Bimonthly	2.059
185	ELSEVIER	Information Systems ISSN:0306-4379 , Bimonthly	1.832
215	ELSEVIER	Advances in Engineering Software ISSN:0965-9978 , Monthly	1.673
235	IEEE	IEEE Transactions on Dependable and Secure Computing ISSN:1545-5971 , Bimonthly	1.592
237	ELSEVIER	Journal of Computer and System Sciences ISSN:0022-0000 , Bimonthly	1.583
241	ELSEVIER	Information and Software Technology ISSN:0950-5849 , Monthly	1.569
254	IEEE	IEEE Transactions on Software Engineering ISSN:0098-5589 , Monthly	1.516
256	ACM	ACM Transactions on Software Engineering and Methodology ISSN:1049-331X , Quarterly	1.513

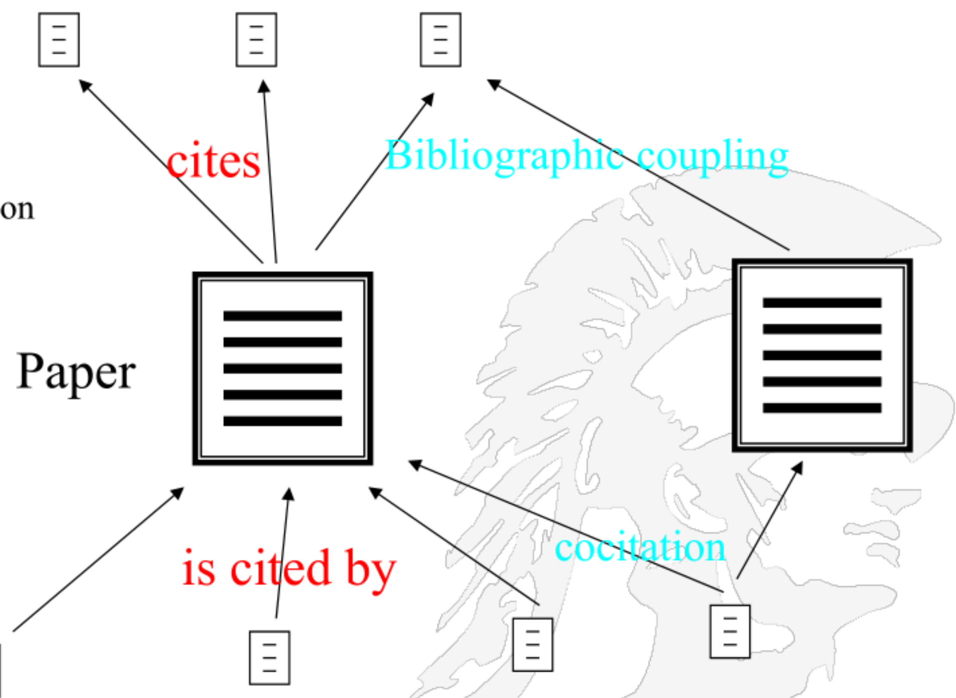
••

University of Southern California

USC Viterbi School of Engineering

Citation Graph

The structure of this graph, independent of content, can provide interesting information about the similarity of documents and the structure of information



Note that academic citations nearly always refer to the author's earlier work.

••

Citations vs. Web Links

- **Web links are a bit different than citations:**
  - Many links are navigational
  - Many pages with high in-degree are portals not content providers
  - Not all links are endorsements
  - Company websites normally don't point to their competitors
  - Citations to relevant literature is enforced by peer-review

••

University of Southern California

USC Viterbi
School of Engineering

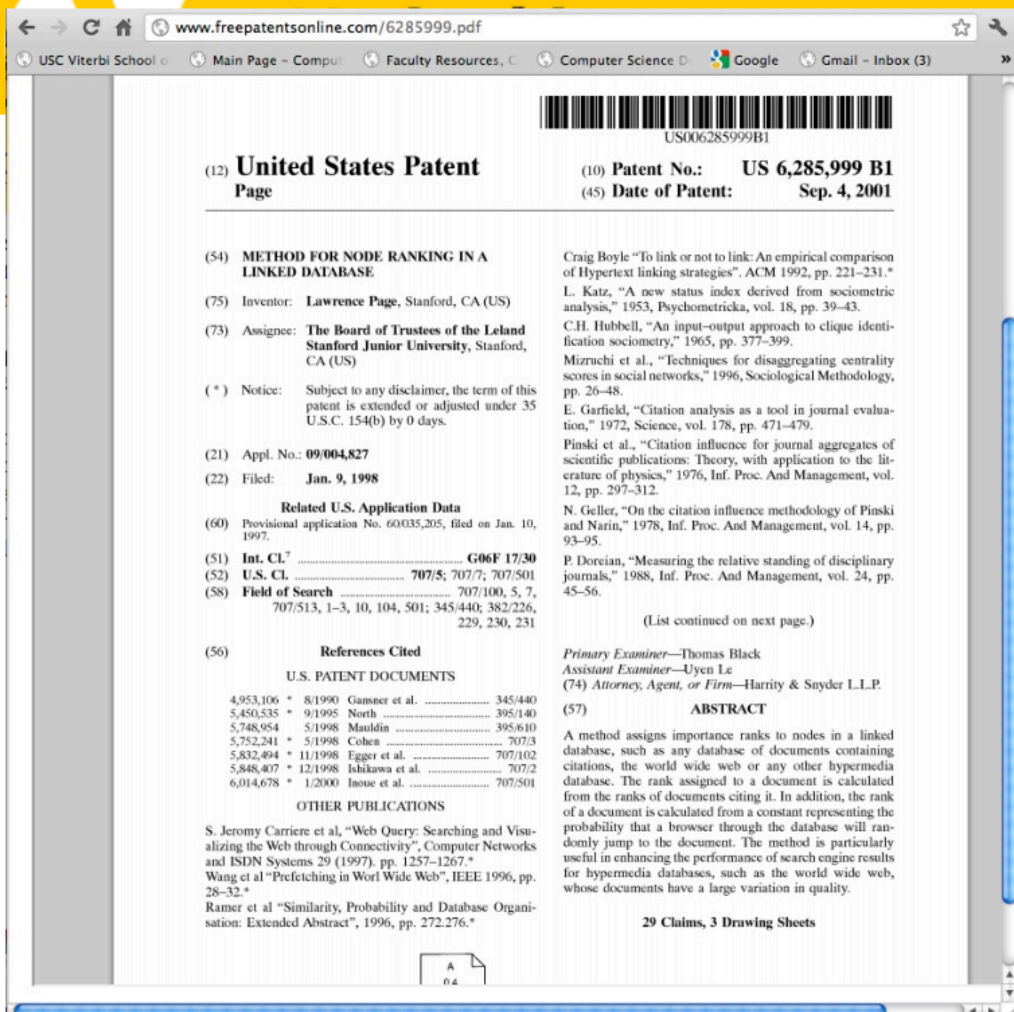
What is PageRank?

- PageRank is a **web link analysis algorithm** introduced by Google
- PageRank was developed at Stanford University by Google founders **Larry Page and Sergey Brin**
  - The paper describing PageRank was co-authored by Rajeev Motwani and Terry Winograd
- PageRank is a "**vote**", by all the other pages on the Web, about how important a page is.
- A link to a page counts as a vote of support.
- If there's no link there's no support (but it's an abstention from voting rather than a vote against the page).
- PageRank says <u>nothing</u> about the content or size of a page, the language it's written in, or the text used in the anchor of a link!
- Looked at another way, PageRank is a **probability distribution** used to represent the likelihood that a person randomly clicking on links will arrive at any particular page

••

University of Southern California

USC



US006285999B1

(12) **United States Patent**
Page

(10) **Patent No.: US 6,285,999 B1**
(45) **Date of Patent: Sep. 4, 2001**

(54) **METHOD FOR NODE RANKING IN A LINKED DATABASE**

(75) Inventor: **Lawrence Page**, Stanford, CA (US)

(73) Assignee: **The Board of Trustees of the Leland Stanford Junior University**, Stanford, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/004,827**

(22) Filed: **Jan. 9, 1998**

Related U.S. Application Data

(60) Provisional application No. 60/035,205, filed on Jan. 10, 1997.

(51) **Int. Cl.**[7] **G06F 17/30**
(52) **U.S. Cl.** **707/5**; 707/7; 707/501
(58) **Field of Search** 707/100, 5, 7, 707/513, 1–3, 10, 104, 501; 345/440; 382/226, 229, 230, 231

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,953,106 *	8/1990	Gansner et al.	345/440
5,450,535 *	9/1995	North	395/140
5,748,954	5/1998	Mauldin	395/610
5,752,241 *	5/1998	Cohen	707/3
5,832,494 *	11/1998	Egger et al.	707/102
5,848,407 *	12/1998	Ishikawa et al.	707/2
6,014,678 *	1/2000	Inoue et al.	707/501

OTHER PUBLICATIONS

S. Jeromy Carriere et al, "Web Query: Searching and Visualizing the Web through Connectivity", Computer Networks and ISDN Systems 29 (1997). pp. 1257–1267.*
Wang et al "Prefetching in Worl Wide Web", IEEE 1996, pp. 28–32.*
Ramer et al "Similarity, Probability and Database Organisation: Extended Abstract", 1996, pp. 272.276.*
Craig Boyle "To link or not to link: An empirical comparison of Hypertext linking strategies". ACM 1992, pp. 221–231.*
L. Katz, "A new status index derived from sociometric analysis," 1953, Psychometricka, vol. 18, pp. 39–43.
C.H. Hubbell, "An input–output approach to clique identification sociometry," 1965, pp. 377–399.
Mizruchi et al., "Techniques for disaggregating centrality scores in social networks," 1996, Sociological Methodology, pp. 26–48.
E. Garfield, "Citation analysis as a tool in journal evaluation," 1972, Science, vol. 178, pp. 471–479.
Pinski et al., "Citation influence for journal aggregates of scientific publications: Theory, with application to the literature of physics," 1976, Inf. Proc. And Management, vol. 12, pp. 297–312.
N. Geller, "On the citation influence methodology of Pinski and Narin," 1978, Inf. Proc. And Management, vol. 14, pp. 93–95.
P. Doreian, "Measuring the relative standing of disciplinary journals," 1988, Inf. Proc. And Management, vol. 24, pp. 45–56.

(List continued on next page.)

Primary Examiner—Thomas Black
Assistant Examiner—Uyen Le
(74) *Attorney, Agent, or Firm*—Harrity & Snyder L.L.P.

(57) **ABSTRACT**

A method assigns importance ranks to nodes in a linked database, such as any database of documents containing citations, the world wide web or any other hypermedia database. The rank assigned to a document is calculated from the ranks of documents citing it. In addition, the rank of a document is calculated from a constant representing the probability that a browser through the database will randomly jump to the document. The method is particularly useful in enhancing the performance of search engine results for hypermedia databases, such as the world wide web, whose documents have a large variation in quality.

29 Claims, 3 Drawing Sheets

Page Rank Patented

A copy of the front page of the patent of the PageRank algorithm; Larry Page is credited as the inventor; the patent was awarded to Stanford University;
the patent was filed January 1998

The PageRank patent expired in 2017. Google holds a perpetual license to the patent.

Google has never pursued other search engine companies for using the PageRank algorithm

••

USC Viterbi School of Engineering

Initial PageRank Formulation

$$PR(u) = \sum_{v \in in(u)} \frac{PR(v)}{|out(v)|}$$

➔ "the PageRank value for a page ***u*** is dependent on the PageRank values for each page ***v*** contained in the set ***in(u)*** (the set of all pages linking to page ***u***), divided by the number **|*out(v)*|** of outgoing links from page ***v***"

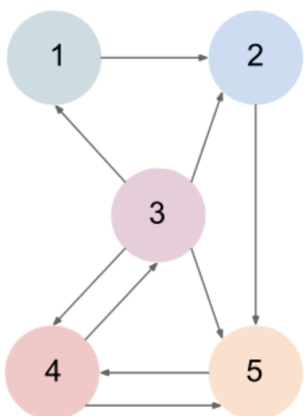
••

Steps for Simplified Algorithm

1. **Iteration 0:** Initialize all ranks to be 1/(number of total pages).
2. **Iteration 1:** For each page **u**, update **u**'s rank to be the sum of each incoming page **v**'s rank from the previous iteration, divided by the number total number of links from page **v**.

USC Viterbi School of Engineering — The Simplified PageRank Algorithm

Example 1



	Iteration 0	Iteration 1
P_1	1/5	1/20
P_2	1/5	5/20
P_3	1/5	1/10
P_4	1/5	5/20
P_5	1/5	7/20

1. <u>Iteration 0:</u> Initialize all pages to have rank $\frac{1}{5}$.
2. <u>Iteration 1:</u>
3. **P_1:** has 1 link from P_3, and P_3 has 4 outbound links, so we take the rank of P_3 from iteration 0 and divide it by 4, which results in rank $(\frac{1}{5})/4$ = 1/20 for P1

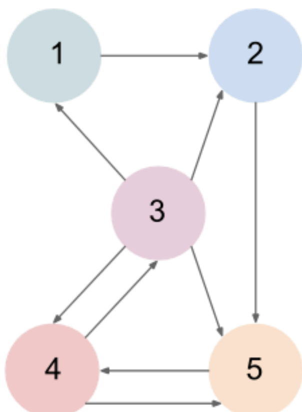
$PR(P_1) = (\frac{1}{5})/4 = 1/20$

4. **P_2:** has 2 links from P_1 and P_3, P_1 has 1 outbound link and P_3 has 4 outbound links, so we take (the rank of P_1 from iteration 0 and divide it by 1) and add that to (the rank of P_3 from iteration 0 and divided that by 4) to get $\frac{1}{5}$ + 1/20 = 5/20 for P_2

$PR(P_2) = \frac{1}{5} + (\frac{1}{5})/4 = 5/20$

The Simplified PageRank Algorithm

Example 1: After 2 iterations



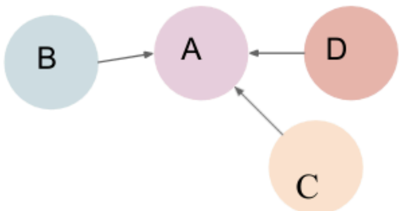
	Iteration 0	Iteration 1	Iteration 2	Final Ranking
P_1	1/5	1/20	1/40	5
P_2	1/5	5/20	3/40	4
P_3	1/5	1/10	5/40	3
P_4	1/5	5/20	15/40	2
P_5	1/5	7/20	16/40	1

$PR(P_5) = \frac{1}{5} + \frac{1}{5} * \frac{1}{4} + \frac{1}{5} * \frac{1}{2} =$
$7/20$

••

Another Example

Example 2



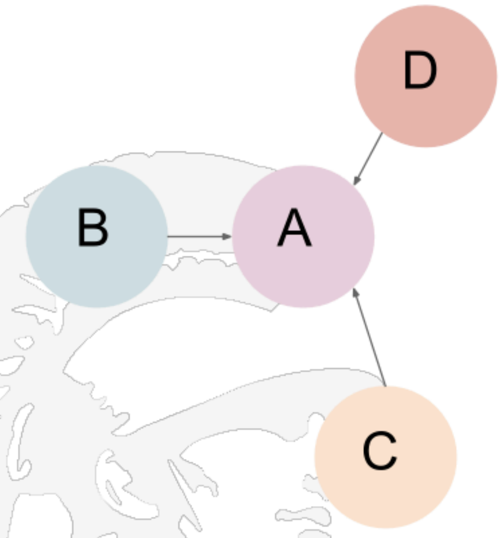
- ➔ Say we have four pages: **A**, **B**, **C** and **D**.
- ➔ Links from a page to itself are ignored (there aren't any in this example).
- ➔ Multiple links from one page to another are treated as a single link.
- ➔ In this example, every page would start out with a rank of 0.25

••

Another Example

Example 2

- Since B, C, and D all have outbound links to A, the Pagerank of A will be **0.75** upon the first iteration
  - (B with rank of 0.25) + (C with rank of 0.25) + (D with rank of 0.25) would transfer all of those ranks to A
- But wait! What about ranks of pages B,C, and D? Because B, C, and D have no incoming edges and they give all their rank to A, they will all end up with a rank of 0. This doesn't add up to 1 . . .
- ***So the simplified algorithm needs to be able to handle border cases, so we must generalize the PageRank algorithm!***



••

USC Viterbi School of Engineering

Complete PageRank Algorithm

- **Quoting from the original Google paper, PageRank is defined like this:**

  "We assume page A has pages T1...Tn which point to it (i.e., are citations). The parameter d is a damping factor which can be set between 0 and 1. We usually set d to 0.85. There are more details about d in the next section. Also C(A) is defined as the number of links going out of page A."*

  The PageRank of a page A is given as follows:

$$PR(A) = (1-d) + d\,(PR(T1)/C(T1) + \ldots + PR(Tn)/C(Tn))$$

- ***Note:***
  - ***That the PageRanks form a probability distribution over web pages, so the sum of all web pages' PageRanks will be one.***

- ******The Anatomy of a Large-Scale Hypertextual Web Search Engine by Brin and Page, http://infolab.stanford.edu/pub/papers/google.pdf***

••

University of Southern California

USC Viterbi
School of Engineering

Explanation

$$PR(A) = (1-d) + d\,(PR(T1)/C(T1) + \ldots + PR(Tn)/C(Tn))$$

- ***PR(A) is PageRank of Page A (one we want to calculate)***
- ***PR(T1) is the PageRank of Site T1 pointing to page A***
- ***C(T1) is the number of outgoing links of page T1***
- ***PR(Tn)/C(Tn) : If page A has a backlink from page "Tn" the share of the vote page A will get is "PR(Tn)/C(Tn)"***
- **d(…) : All these fractions of votes are added together but, to stop the other pages having too much influence, this total vote is "damped down" by multiplying it by the factor "d", (0.85)**
- **(1-d) : Since "*sum of all web pages' PageRanks will be one*". Even if the d(…) is 0 then the page will get a small PR of 0.15**

••

University of Southern California

USC Viterbi School of Engineering

How PageRank is Calculated

- PR of each page depends on PR of other pages which are pointing to it. But we don't know PR of a given page until the PR of other pages is calculated and so on…
- From the Google paper:

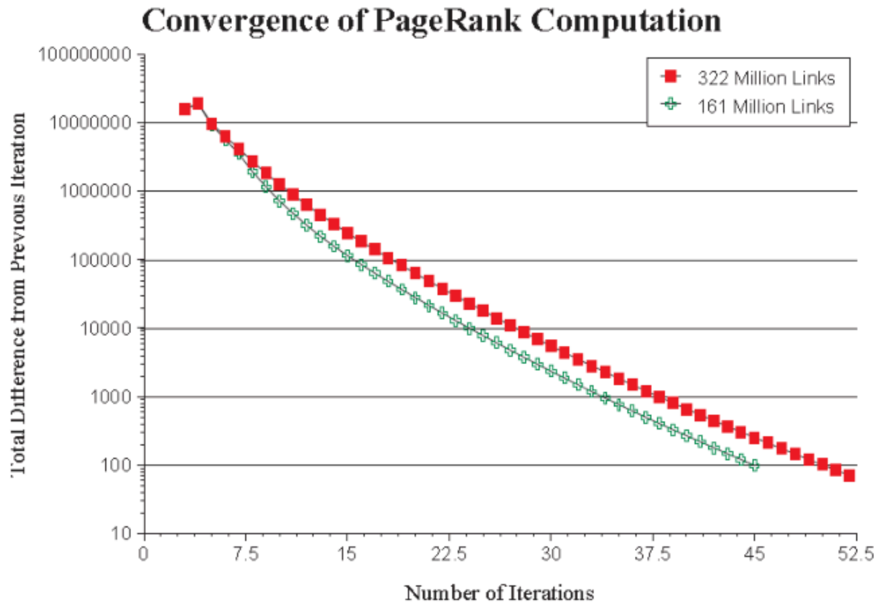
  PageRank or PR(A) can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the web
- What this means is that we can calculate the PR without knowing the final PR of other pages
  - Recurrence Relation: an equation that recursively defines a sequence of values once one or more initial terms are given;
- We calculate PR iteratively a number of times until we start converging to the same value.

••

USC Viterbi School of Engineering

How Fast Does the PageRank Algorithm Converge

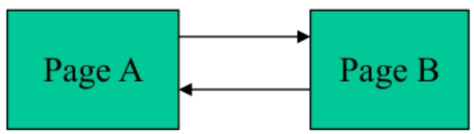
- Early experiments on Google used 322 million links
- PR (322 Million Links): 52 iterations
- PR (161 Million Links): 45 iterations
- Number of iterations required for convergence is empirically (but not formally derived) *O(log n)* (where *n* is the number of links)
- Hence the calculation is quite efficient



••

Computing PageRank by Iteration Example

- Consider 2 pages: Page A and Page B pointing to each other.



- C(A) = 1, number of outgoing links of A
- C(B) = 1, number of outgoing links of B
- What should be the initial value of P(A) or P(B) ?

••

Guess 1:

- Suppose the initial values are :
  - P(A) = 1 and P(B) = 1 and d= 0.85; then

    PR(A)= (1 – d) + d*(PR(B)/1)

    PR(B)= (1 – d) + d*(PR(A)/1)

i.e.

- PR(A)= 0.15 + 0.85 * 1
  = 1
- PR(B)= 0.15 + 0.85 * 1
  = 1
- In one iteration we are done
- Let's try another set of initial values.

••

USC Viterbi School of Engineering

Guess 2 With 3 Iterations

- Initial Values : P(A) = 0, P(B) = 0 and d= 0.85

PR(A)= (1 – d) + d(PR(B)/1)
PR(B)= (1 – d) + d(PR(A)/1)

- PR(A)= 0.15 + 0.85 * 0 = 0.15
  PR(B)= 0.15 + 0.85 * 0.15 = 0.2775

Iterating again we get:

- PR(A)= 0.15 + 0.85 * 0.2775 = 0.385875
  PR(B)= 0.15 + 0.85 * 0.385875 = 0.47799375

And iterating again

- PR(A)= 0.15 + 0.85 * 0.47799375 = 0.5562946875
  PR(B)= 0.15 + 0.85 * 0.5562946875 = 0.62285048437

••

University of Southern California

USC Viterbi School of Engineering

USC

Guess 2: Continued…

- After 20 iterations…

- PR(A) = 0.99
- PR(B) = 0.99

- Both approaching to 1.

	A	B	C	D
1	C(A)	1		
2	C(B)	1		
3				
4	Iterations	PR(A)	PR(B)	
5				
6	0	0	0	
7	1	0.15	0.2775	
8	2	0.385875	0.47799375	
9	3	0.556294688	0.622850484	
10	4	0.679422912	0.727509475	
11	5	0.768383054	0.803125596	
12	6	0.832656756	0.857758243	
13	7	0.879094506	0.89723033	
14	8	0.912645781	0.925748914	
15	9	0.936886577	0.94635359	
16	10	0.954400552	0.961240469	
17	11	0.967054399	0.971996239	
18	12	0.976196803	0.979767283	
19	13	0.98280219	0.985381862	
20	14	0.987574582	0.989438395	
21	15	0.991022636	0.99236924	
22	16	0.993513854	0.994486776	
23	17	0.99531376	0.996016696	
24	18	0.996614191	0.997122063	
25	19	0.997553753	0.99792069	
26	20	0.998232587	0.998497699	
27				

••

University of Southern California

USC Viterbi School of Engineering

Guess 3:

- Initial Values : P(A) = 40 and P(B) = 40
  d= 0.85

PR(A)= (1 – d) + d(PR(B)/1)

PR(B)= (1 – d) + d(PR(A)/1)

- Notice decreasing value of PR
- The page rank is again approaching to 1.
- So it doesn't matter where you start your guess, once the PageRank calculations have settled down, the "*normalized probability distribution*" (the average PageRank for these two pages) will be 1.0

	A	B	C	D
1	C(A)	1		
2	C(B)	1		
3				
4	Iterations	PR(A)	PR(B)	
5				
6	0	0	40	
7	1	34.15	29.1775	
8	2	24.950875	21.35824375	
9	3	18.30450719	15.70883111	
10	4	13.50250644	11.62713048	
11	5	10.03306091	8.678101769	
12	6	7.526386504	6.547428528	
13	7	5.715314249	5.008017112	
14	8	4.406814545	3.895792363	
15	9	3.461423509	3.092209982	
16	10	2.778378485	2.511621712	
17	11	2.284878455	2.092146687	
18	12	1.928324684	1.789075981	
19	13	1.670714584	1.570107397	
20	14	1.484591287	1.411902594	
21	15	1.350117205	1.297599624	
22	16	1.252959681	1.215015728	
23	17	1.182763369	1.155348864	
24	18	1.132046534	1.112239554	
25	19	1.095403621	1.081093078	
26	20	1.068929116	1.058589749	
27				

The last three slides (including this one) are a bit confusing - because they use the current (rather than previous) values of PR(A), to calculate PR(B). "Technically" we ARE supposed to be using PR(A) and PR(B) from the previous step, to update them for the current step. We can 'get away with it' because... it doesn't matter (which we use)! After enough steps, either way will result in proper, converged values.

The following, illustrate the "proper" way, but with arbitrary starting values - to show they don't matter.

bytes.usc.edu/~saty/tools/xem/run.html?x=bare#PGhlYWQ+CiA8IS0tIGFsc28gdmlhOiBiaXQubHkveGVtY...

'BAREBONES' LIVECODING - enter HTML+CSS+JS on the left, below.

```
 <style>
 .console-log-div {
  border: 1px solid gray;
  padding: 5px 10px;
  border-radius: 5px;
  width: 93% !important;
  background-color: #efefef;
  overflow: auto;
  height : 400px;
 }
 </style>
</head>
<!------------------------------------------------------------------->
<body>
 <script src="https://bytes.usc.edu/~saty/tools/xem/c/prtt.js"> </script>
 <p></p>
 <script>


   prAPrev=1, prBPrev=1;

   for(i=0; i<20; i++){
     prACurr = 0.15 + 0.85*prBPrev;
     prBCurr = 0.15 + 0.85*prAPrev;

     prtt(prACurr + "," + prBCurr);

     prAPrev = prACurr;
     prBPrev = prBCurr;
   }
   prtt(prACurr + "," + prBCurr);
 </script>
</body>
```

console output

```
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
1,1
```

BAREBONES' LIVECODING - enter HTML+CSS+JS on the left, below.

```
 .console-log-div {
  border: 1px solid gray;
  padding: 5px 10px;
  border-radius: 5px;
  width: 93% !important;
  background-color: #efefef;
  overflow: auto;
  height : 400px;
 }
 </style>
</head>
<!------------------------------------------------------------------->
<body>
 <script src="https://bytes.usc.edu/~saty/tools/xem/c/prtt.js"> </script>
 <p></p>
 <script>


   prAPrev=0, prBPrev=0;

   for(i=0; i<20; i++){
     prACurr = 0.15 + 0.85*prBPrev;
     prBCurr = 0.15 + 0.85*prAPrev;

     prtt(prACurr + "," + prBCurr);

     prAPrev = prACurr;
     prBPrev = prBCurr;
   }
   prtt(prACurr + "," + prBCurr);
 </script>
</body>
```

console output

```
0.15,0.15
0.27749999999999997,0.27749999999999997
0.38587499999999997,0.38587499999999997
0.47799375,0.47799375
0.5562946875,0.5562946875
0.622850484375,0.622850484375
0.67942291171875,0.67942291171875
0.7275094749609375,0.7275094749609375
0.7683830537167969,0.7683830537167969
0.8031255956592774,0.8031255956592774
0.8326567563103858,0.8326567563103858
0.857758242863828,0.857758242863828
0.8790945064342538,0.8790945064342538
0.8972303304691157,0.8972303304691157
0.9126457808987484,0.9126457808987484
0.9257489137639361,0.9257489137639361
0.9368865766993457,0.9368865766993457
0.9463535901944439,0.9463535901944439
0.9544005516652773,0.9544005516652773
0.9612404689154856,0.9612404689154856
0.9612404689154856,0.9612404689154856
```

BAREBONES' LIVECODING - enter HTML+CSS+JS on the left, below.

```
<head>
 <!-- also via: bit.ly/xembare -->
 <style>
 .console-log-div {
  border: 1px solid gray;
  padding: 5px 10px;
  border-radius: 5px;
  width: 93% !important;
  background-color: #efefef;
  overflow: auto;
  height : 600px;
 }
 </style>
</head>
<!------------------------------------------------------------------->
<body>
 <script src="https://bytes.usc.edu/~saty/tools/xem/c/prtt.js"> </script>
 <p></p>
 <script>


   prAPrev=0.15, prBPrev=0;  //!!

   for(i=0; i<60; i++){
     prACurr = 0.15 + 0.85*prBPrev;
     prBCurr = 0.15 + 0.85*prAPrev;

     prtt(prACurr + "," + prBCurr);

     prAPrev = prACurr;
     prBPrev = prBCurr;
   }
   prtt(prACurr + "," + prBCurr);
 </script>
</body>
```

console output

```
0.15,0.27749999999999997
0.38587499999999997,0.27749999999999997
0.38587499999999997,0.47799375
0.5562946875,0.47799375
0.5562946875,0.622850484375
0.67942291171875,0.622850484375
0.67942291171875,0.7275094749609375
0.7683830537167969,0.7275094749609375
0.7683830537167969,0.8031255956592774
0.8326567563103858,0.8031255956592774
0.8326567563103858,0.857758242863828
0.8790945064342538,0.857758242863828
0.8790945064342538,0.8972303304691157
0.9126457808987484,0.8972303304691157
0.9126457808987484,0.9257489137639361
0.9368865766993457,0.9257489137639361
0.9368865766993457,0.9463535901944439
0.9544005516652773,0.9463535901944439
0.9544005516652773,0.9612404689154856
0.9670543985781628,0.9612404689154856
0.9670543985781628,0.9719962387914384
0.9761968029727226,0.9719962387914384
0.9761968029727226,0.9797672825268142
0.9828021901477921,0.9797672825268142
0.9828021901477921,0.9853818616256234
0.9875745823817799,0.9853818616256234
0.9875745823817799,0.9894383950245129
0.9910226357708359,0.9894383950245129
0.9910226357708359,0.9923692404052106
0.993513854344429,0.9923692404052106
0.993513854344429,0.9944867761927646
0.99531375976385,0.9944867761927646
0.99531375976385,0.9960166957992724
0.9966141914293816,0.9960166957992724
0.9966141914293816,0.9971220627149744
0.9975537533077282,0.9971220627149744
0.9975537533077282,0.9979206903115689
0.9982325867648336,0.9979206903115689
0.9982325867648336,0.9984976987501086
```

'BAREBONES' LIVECODING - enter HTML+CSS+JS on the left, below.

```
<head>
 <!-- also via: bit.ly/xembare -->
 <style>
 .console-log-div {
  border: 1px solid gray;
  padding: 5px 10px;
  border-radius: 5px;
  width: 93% !important;
  background-color: #efefef;
  overflow: auto;
  height : 600px;
 }
 </style>
</head>
<!------------------------------------------------------------------->
<body>
 <script src="https://bytes.usc.edu/~saty/tools/xem/c/prtt.js"> </script>
 <p></p>
 <script>


   prAPrev=10.15  prBPrev=0;

   for(i=0; i<60; i++){
     prACurr = 0.15 + 0.85*prBPrev;
     prBCurr = 0.15 + 0.85*prAPrev;

     prtt(prACurr + "," + prBCurr);

     prAPrev = prACurr;
     prBPrev = prBCurr;
   }
   prtt(prACurr + "," + prBCurr);
 </script>
</body>
```

console output

```
0.9761968029727226,1.2177992527995871
1.185129364879649,0.9797672825268142
0.9828021901477921,1.1573599601477016
1.1337559661255463,0.9853818616256234
0.9875745823817799,1.1136925712067143
1.096638685525707,0.989438395024512 9
0.9910226357708359,1.0821428826968509
1.0698214502923231,0.9923692404052106
0.993513854344429,1.059348232748474 7
1.0504459978362035,0.9944867761927646
0.99531375976385,1.042879098160773
1.036447233436657,0.9960166957992724
0.9966141914293816,1.0309801484211585
1.0263331261579847,0.9971220627149744
0.9975537533077282,1.022383157234287
1.0190256836491438,0.9979206903115689
0.9982325867648336,1.0161718311017722
1.0137460564365064,0.9984976987501086
0.9987230439375924,1.0116841479710303
1.0099315257753758,0.9989145873469535
0.9990773992449105,1.0084417969090693
1.0071755273727088,0.9992157893581739
0.9993334209544478,1.0060991982668024
1.005184318526782,0.9994334078112806
0.9995183966395885,1.0044066707477646
1.0037456701355998,0.9995906371436503
0.9996520415721027,1.0031838196152598
1.0027062466729708,0.9997042353362874
0.9997486000358443,1.002300309672025
1.001955263221221 2,0.9997863100304677
0.9998183635258976,1.001661973738038
1.0014126776773322,0.9998456089970129
0.999868767647461,1.0012007760257324
1.0010206596218725,0.9998884525003419
0.9999051846252905,1.0008675606785915
1.0007374265768028,0.999919406931497
0.9999314958917724,1.0006268125902822
1.0005327907017398,0.9999417715080066
1.0005327907017398,0.9999417715080066
```

••

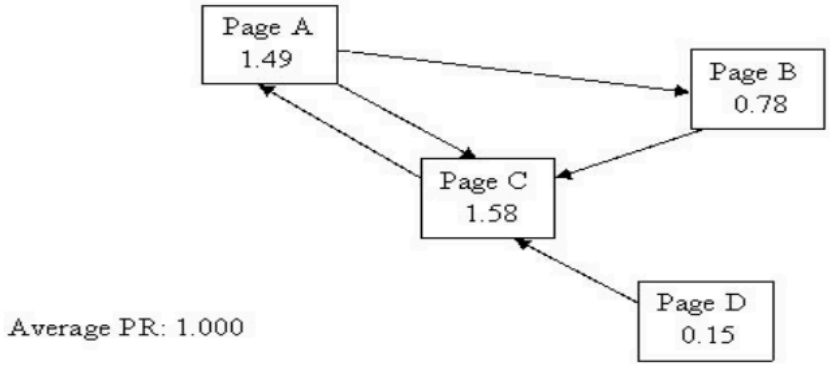
USC Viterbi School of Engineering

PageRank Convergence?

- Some observations about the damping factor
- The damping factor value and its effect:
  - For certain graphs the simple update rule can cause pagerank to accumulate and get stuck in certain parts of the graph
    - E.g. if a page has no outgoing links to other pages, it is called a sink
    - The simplified pagerank algorithm can get stuck in sinks
  - This is fixed by having each node on every round
    - Give a *d* fraction of its pagerank to its neighbors
    - Give a *(1-d)* fraction of its pagerank to everyone in the graph
    - As a result, pages with no incoming links will get some pagerank
  - If too high, more iterations are required
  - If too low, you get repeated over-shoot,
  - Both above and below the average – the numbers just swing like pendulum and never settle down.

••

Examples of Relative PageRanks 1:



- **Observations:**
  - **Every page has at least a PR of 0.15 to start out**
  - **Page D has no votes but still it has a PR of 0.15**
  - **It is believed that Google undergoes a post-spidering phase whereby any pages that have no incoming links at all are ignored wrt PageRank**

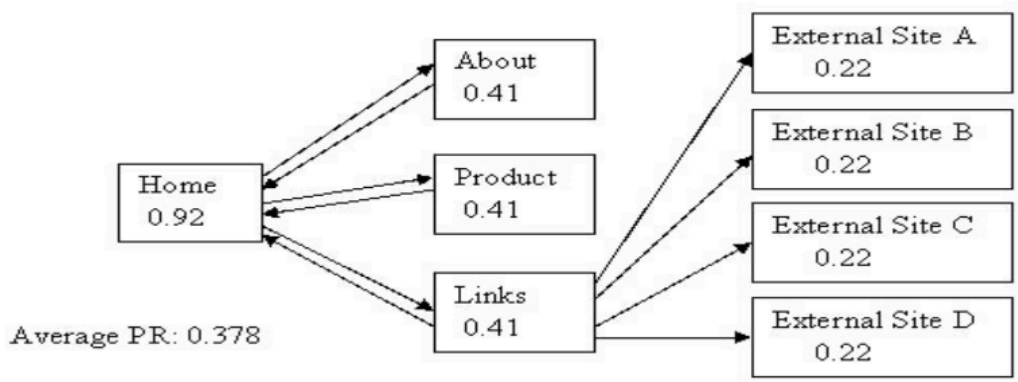
  - **Examples on the following pages are taken from http://www.sirgroane.net/google-page-rank/**

••

University of Southern California

USC Viterbi School of Engineering

Example 2: Simple hierarchy with Some Outgoing Links

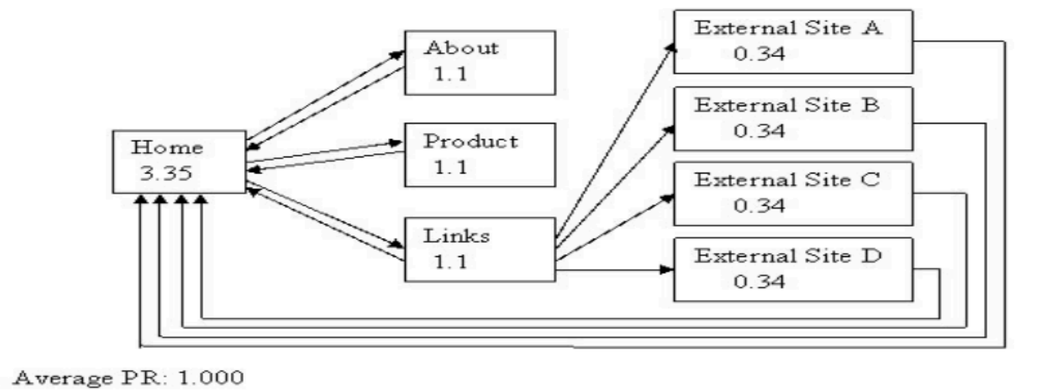


- **Observations:**
  - **Home has the most PR**
  - **But average PR is 0.378**
    - **"External site" pages are not voting for anyone.**
  - **Links within your own site can have a significant effect on PageRank.**

••

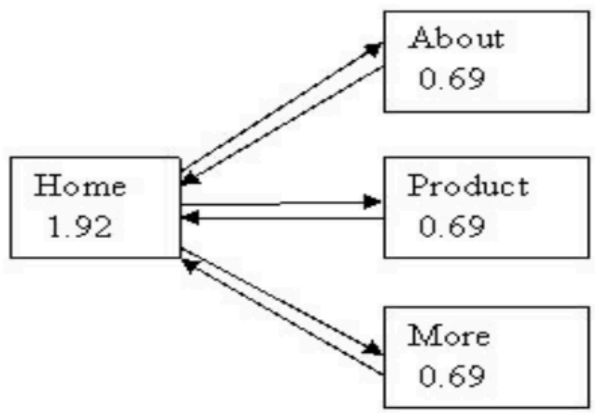
USC Viterbi School of Engineering

Example 3: Linking External Sites Back into our Home Page



- **Observations:**
  - **Comparing this example with the previous one, now the Pagerank of the Home Page has increased significantly.**
  - **Moreover, the average PageRank is better because of all the external sites linking back to the home page.**

Example 4: Simple Hierarchy



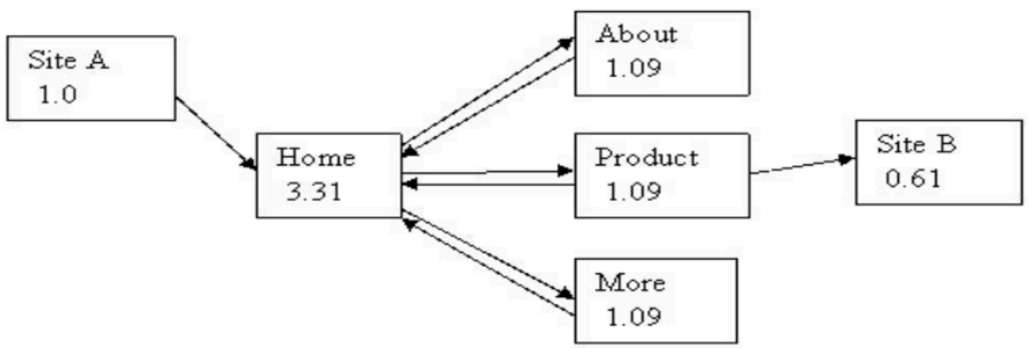
- **Observations:**
  - **Home Page has PageRank of 2.5 times the page rank of its child pages.**
  - **A hierarchy structure concentrates votes and PageRank into one page.**

••

University of Southern California

USC Viterbi School of Engineering

Example 5: Hierarchical – But with One Link In and One Out



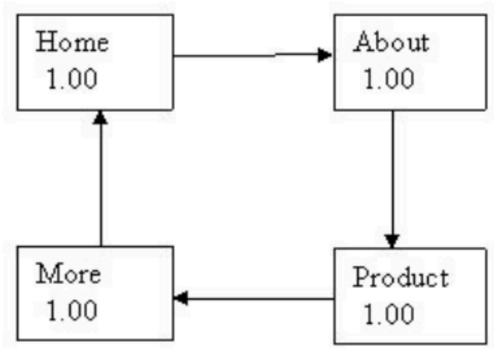
- **Observations:**
  - **The PageRank of Home page has increased from 1.92 (Previous Example) to 3.31**
  - **Site A contributed 0.85 PR to Home page and the raised PageRank in the "About", "Product" and "More" pages has had a lovely "feedback" effect, pushing up the home page's PageRank even further!**
- **A well structured site will amplify the effect of any contributed PR.**

••

University of Southern California

USC Viterbi School of Engineering

Example 6: Looping



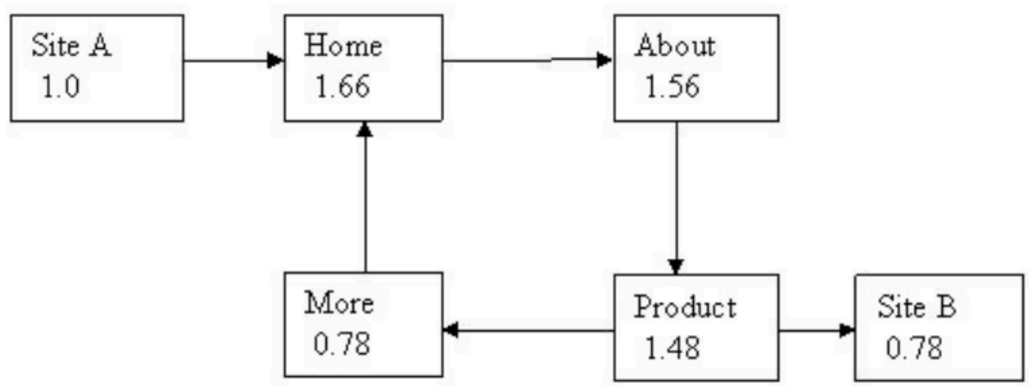
- **Observations:**
  - **All the pages have the same number of incoming links, i.e. all pages are of equal importance to each other.**
  - **Each page has PR of 1.0**
  - **Average PR is 1.0**

••

University of Southern California

USC Viterbi School of Engineering

Example 7: Looping – But with a Link in and a Link Out

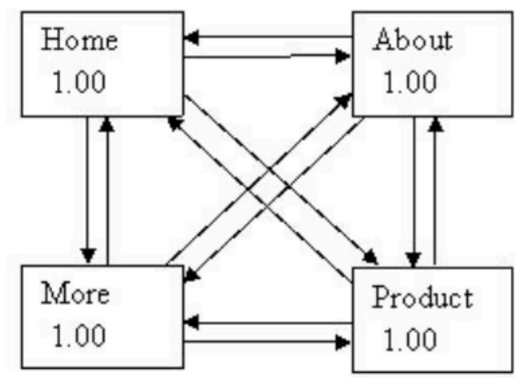


- **Observations:**
  - **PageRank of our home page has gone up a little, but what's happened to the "More" page? Its PageRank has gone down**
  - **Now the Pagerank value of the external Site B is equal to the "More" page.**
  - **The vote of the "Product" page has been split evenly between "More" page and the external site B.**
  - **This is good for Site B for otherwise its PageRank would be 0.15**

••

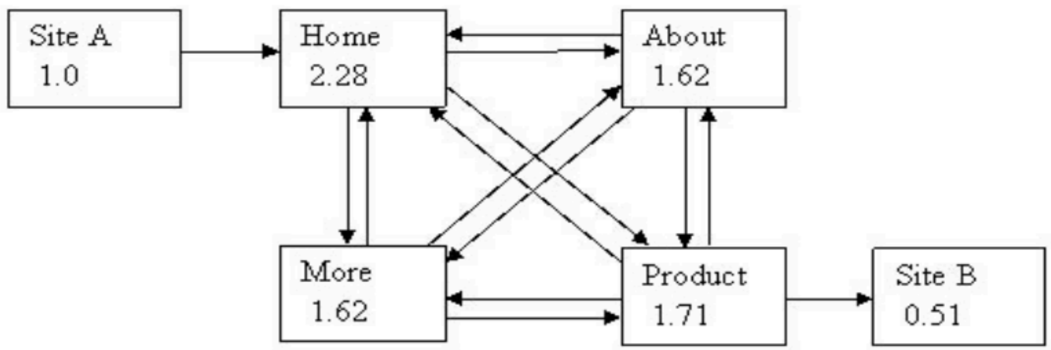
USC Viterbi School of Engineering

Example 8: Extensive Interlinking or Fully Meshed



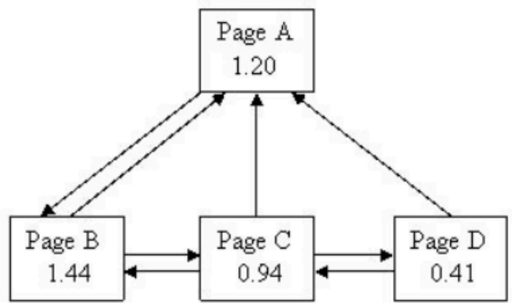
- **Observations:**
  - **All pages are of equal importance to each other so the resulting PageRank is no different than what was gotten in Example 6**

Example 9: Fully Meshed – But with One Vote in and One Vote Out



- **Observations:**
  - **"Product" page has kept three quarters of its vote within our site unlike example 9 where it was giving away fully half of it's vote to the external site!**
  - **Keeping just this small extra fraction of the vote within our site has had a very nice effect on the Home Page too.**

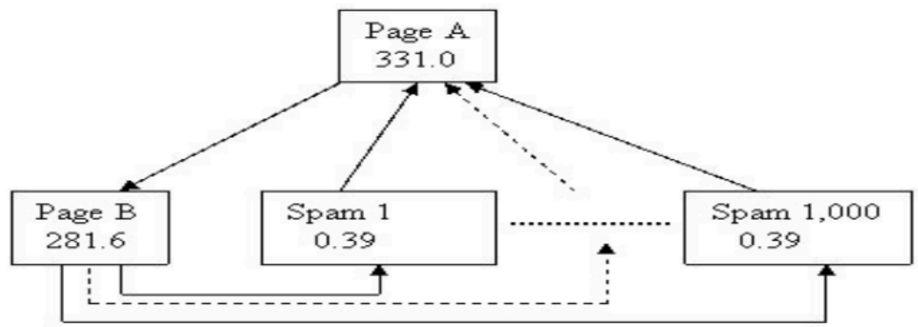
Example 10: Previous … Next … Documentation Page Layout



- The first page of the document has a higher PR than the Home Page! This is because page B is getting all the vote from page A, but page A is only getting fractions of pages B, C and D.
- In order to give users of your site a good experience, you may have to take a hit against your PR

••

Example 11: Getting Higher PR the Wrong Way!



Average PR: 1.000

- **Observations:**
  - **1000 incoming links and only one outgoing link**
  - **It doesn’t matter how many pages you have in your site, your average PR will always be 1.0 at best.**
  - **But a hierarchical layout can strongly concentrate votes, and therefore the PR, into the home page!**
  - **This is a technique used by some disreputable sites**

A link farm is set of web pages created with the sole aim of linking to a target page, in an attempt to improve that page's search engine ranking.

••

Some Suggestions Based on What We Have Seen in Examples.

- **Suggestions for improving your page rank**
  - Increasing the internal links in your site can minimize the damage to your PR when you give away votes by linking to external sites.
  - If a particular page is highly important – use a hierarchical structure with the important page at the "top".
  - Where a group of pages may contain outward links – increase the number of internal links to retain as much PR as possible.
  - Where a group of pages do not contain outward links – the number of internal links in the site has no effect on the site's average PR. You might as well use a link structure that gives the user the best navigational experience.
  - **Use Site Maps**: Linking to a site map on each page increases the number of internal links in the site, spreading the PR out and protecting you against your vote "donations" to other sites

••

USC Viterbi School of Engineering

Importance of PageRank

- **PageRank is just one factor Google uses to determine a page's relevance. It assumes that people will link to your page only if they think the page is good. But this is not always true.**
- **Content is still the king!!!**
  - Anchor, body, title tags etc. still are very important for search engines
- **From Chris Ridings' Paper, "PageRank Uncovered" (http://www.voelspriet2.nl/PageRank.pdf):**
  - You must do enough "on the page" work and/or anchor text work to get into that subset of top pages for your chosen key phrase, otherwise your high PageRank will be completely in vain.
- **PageRank is a multiplier factor.**
  - If Google wants to penalize any page they can set the PageRank equal to a small number, even 0. As a result it will be pushed down on the search results page.

••

You can play with PR here:
https://bytes.usc.edu/~saty/tools/xem/run.html?x=PR