Introduction to

Information Retrieval

ΠΛΕ70: Ανάκτηση Πληροφορίας

Διδάσκουσα: Ευαγγελία Πιτουρά

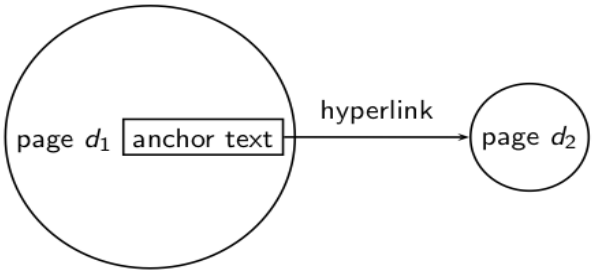
Διάλεξη 10: Ανάλυση Συνδέσμων.

Τι θα δούμε σήμερα

Ανάλυση συνδέσμων (Link Analysis)

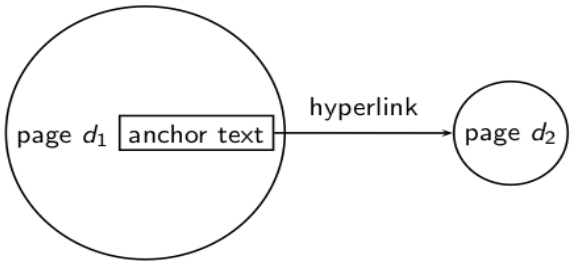
- Σημασία της άγκυρας (anchor text)
- PageRank
- HITS (Κομβικές σελίδες και σελίδες κύρους)

Σημασία των συνδέσεων



- Assumption 1: A hyperlink is a quality signal.
  - The hyperlink $d_1 \rightarrow d_2$ indicates that d_1's author deems d_2 high-quality and relevant.

Κείμενο Άγκυρας



Anchor text (κείμενο άγκυρας) κείμενο που περιβάλει τον σύνδεσμο

- Example: "You can find cheap cars <a href =http://…>here . "
- Anchor text: "You can find cheap cars here"

- Assumption 2: The anchor text describes the content of d_2.

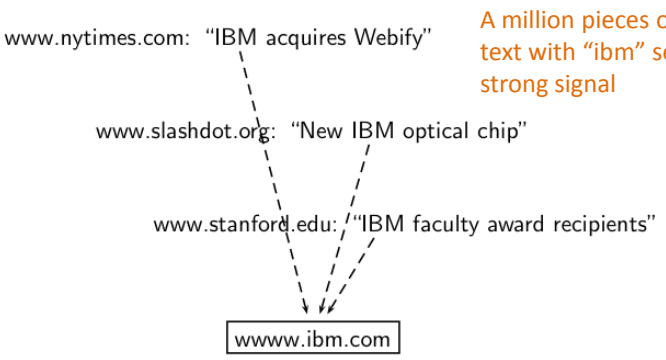
Κείμενο Άγκυρας

Χρήση μόνο **[text of d_2]** ή **[text of d_2] + [anchor text → d_2]**

- Αναζήτηση του [text of d_2] + [anchor text → d_2] συχνά πιο αποτελεσματική από την αναζήτηση μόνο του [text of d_2]
- Παράδειγμα: Ερώτημα *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article
  - May not match IBM home page! if IBM home page is mostly graphics

Κείμενο Άγκυρας

- Αναζήτηση με χρήση του [anchor text → d_2] καλύτερη για το ερώτημα IBM
  - Η σελίδα με τις περισσότερες εμφανίσεις του όρου *IBM* είναι η www.ibm.com

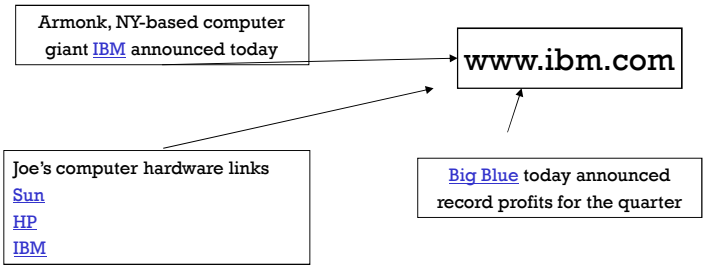


A million pieces of anchor text with "ibm" send a strong signal

Κείμενο Άγκυρας στο Ευρετήριο

Thus: Anchor text is often a better description of a page's content than the page itself.

- When indexing a document *D*, include (with some weight) anchor text from links pointing to *D*.



Google Bombs

Google bomb: a search with "bad" results due to maliciously manipulated anchor text.

Google introduced a new weighting function in January 2007

✓ *Can score anchor text with weight depending on the authority of the anchor page's website*

E.g., if we were to assume that content from cnn.com or yahoo.com is authoritative, then trust the anchor text from them

- Still some remnants: [dangerous cult] on Google, Bing, Yahoo
  - Coordinated link creation by those who dislike the Church of Scientology
- Defused Google bombs: [dumb motherf…], [who is a failure?], [evil empire] [cheerful achievement]

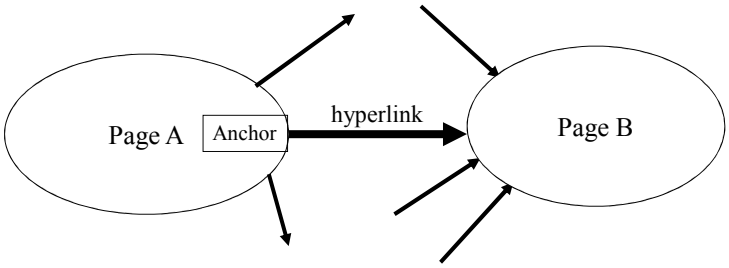
Assumption 2: annotation of target

9

Anchor Text

- Other applications
  - Weighting/filtering links in the graph
  - Generating page descriptions from anchor text

The Web as a Directed Graph

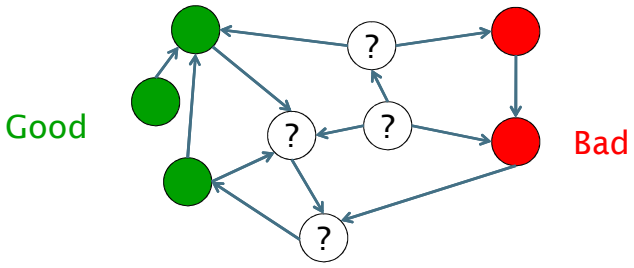


Assumption 1: A hyperlink between pages denotes a conferral of authority (quality signal)

Assumption 2: The text in the anchor of the hyperlink describes the target page (textual context)

Οι σύνδεσμοι είναι παντού!

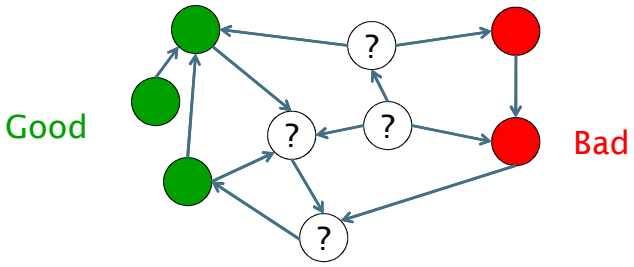
- Powerful sources of authenticity and authority
  - Mail spam – which email accounts are spammers?
  - Host quality – which hosts are "bad"?
  - Phone call logs
- The Good, The Bad and The Unknown



12

Simple iterative logic

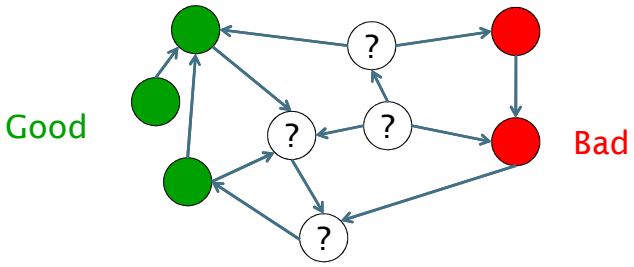
- The Good, The Bad and The Unknown
  - Good nodes won't point to Bad nodes
  - All other combinations plausible



13

Simple iterative logic

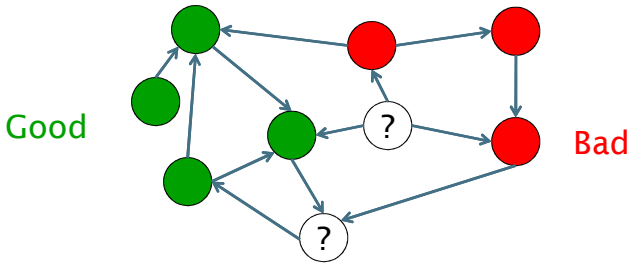
- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



14

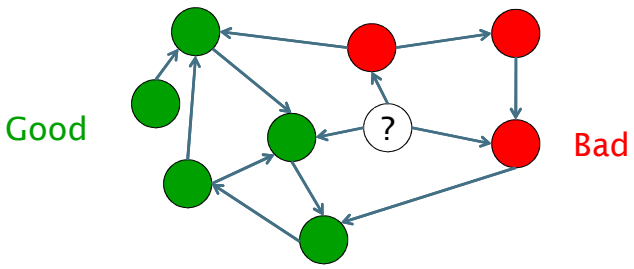
Simple iterative logic

- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



Simple iterative logic

- Good nodes won't point to Bad nodes
  - If you point to a Bad node, you're Bad
  - If a Good node points to you, you're Good



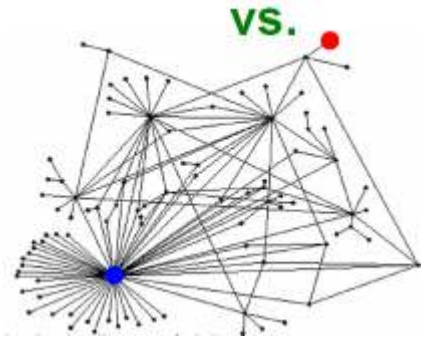
Many other examples of link analysis

- Social networks are a rich source of grouping behavior
- E.g., Shoppers' affinity – Goel+Goldstein 2010
  - Consumers whose friends spend a lot, spend a lot themselves
- http://www.cs.cornell.edu/home/kleinber/networks-book/

PageRank

Link Analysis

Not all web pages are equal



PageRank

- A page is important if it has many links (ingoing? outgoing?)
- Are all links equal important?
- A page is important if it is cited by other important pages

PageRank: Βασική ιδέα

Κάθε σελίδα έχει ένα PageRank

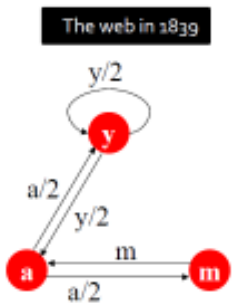
- Κάθε σελίδα μοιράζει το PageRank στις σελίδες που δείχνει
- Το PageRank μιας σελίδας είναι το άθροισμα των PageRank των σελίδων που δείχνουν σε αυτή

PageRank: Παράδειγμα

- A "vote" from an important page is worth more
- A page is important if it is pointed to by other important pages
- Define a "rank" r_j for node j

$$r_j = \sum_{i \to j} \frac{r_i}{d_i}$$

d_i ... out-degree of node i



"Flow" equations:

$r_y = r_y/2 + r_a/2$

$r_a = r_y/2 + r_m$

$r_m = r_a/2$

PageRank: Αλγόριθμος

In a network with n nodes, assign all nodes the same initial PageRank = 1/n.

- Perform a sequence of *k* updates to the PageRank values, using the following rule:

Basic PageRank Update Rule:

- Each page ***divides*** *its current PageRank equally across its* ***out-going*** *links*, and passes these equal shares to the pages it points to.

  (If a page has no out-going links, it passes all its current PageRank to itself.)
- Each page updates its new PageRank to be the sum of the shares it receives.

PageRank: Αλγόριθμος

Given a web graph with n nodes, where the nodes are pages and edges are hyperlinks

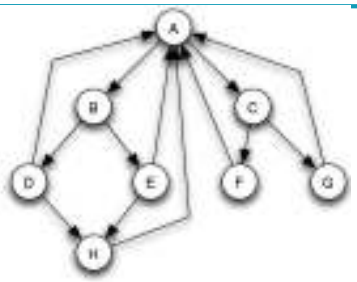
- Assign each node an initial page rank
- Repeat until convergence ($\Sigma_i |r_i^{(t+1)} - r_i^{(t)}| < \varepsilon$)
  - Calculate the page rank of each node

$$r_j^{(t+1)} = \sum_{i \to j} \frac{r_i^{(t)}}{d_i}$$

d_i .... out-degree of node i

PageRank: Αλγόριθμος

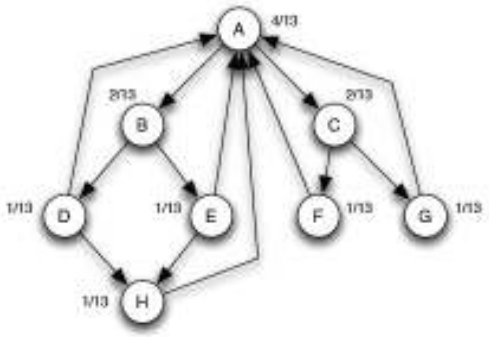
Initially, all nodes PageRank 1/8



Step	A	B	C	D	E	F	G	H
1	1/2	1/16	1/16	1/16	1/16	1/16	1/16	1/8
2	3/16	1/4	1/4	1/32	1/32	1/32	1/32	1/16

✓ As a kind of "fluid" that circulates through the network
✓ The total PageRank in the network remains constant (no need to normalize)

PageRank: Αλγόριθμος



✓ A simple way to check whether an assignment of numbers s forms an equilibrium set of PageRank values: check that they sum to 1, and check that when apply the Basic PageRank Update Rule, we get the same values back.

✓ If the network is strongly connected then there is a unique set of equilibrium values.

PageRank: Διανυσματική αναπαράσταση

Stochastic Adjacency Matrix

Πίνακας M – πίνακας γειτνίασης του web

Αν j -□ i, τότε Mij = 1/outdegree(j)

Αλλιώς, Mij = 0

Page Rank Vector

Ένα διάνυσμα με μία τιμή για κάθε σελίδα (το PageRank της σελίδας)

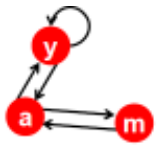
PageRank: Διανυσματική αναπαράσταση

- **Stochastic adjacency matrix M**
  - Let page j has d_j out-links
  - If $j \rightarrow i$, then $M_{ij} = \frac{1}{d_j}$ else $M_{ij} = 0$
    - M is a **column stochastic matrix**
      - Columns sum to 1
- **Rank vector r:** vector with an entry per page
  - r_i is the importance score of page i
  - $\sum_i r_i = 1$
- **The flow equations can be written**

$$r = M \cdot r \qquad r_j = \sum_{i \rightarrow j} \frac{r_i}{d_i}$$

PageRank: Διανυσματική αναπαράσταση

- **Power Iteration:**
  - Set $r_j = 1$
  - $r_j = \sum_{i \to j} \frac{r_i}{d_i}$
    - And iterate



	y	a	m
y	½	½	0
a	½	0	1
m	0	½	0

$r_y = r_y/2 + r_a/2$
$r_a = r_y/2 + r_m$
$r_m = r_a/2$

- **Example:**

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 1/3 & 5/12 & 9/24 & \dots & 6/15 \\ 1/3 & 3/6 & 1/3 & 11/24 & & 6/15 \\ 1/3 & 1/6 & 3/12 & 1/6 & & 3/15 \end{matrix}$$

Iteration 0, 1, 2, …

29

PageRank: Διανυσματική αναπαράσταση

$$r_j^{(t+1)} = \sum_{i \to j} \frac{r_i^{(t)}}{d_i} \quad \text{or equivalently} \quad r = Mr$$

- **Does this converge?**
- **Does it converge to what we want?**
- **Are results reasonable?**

30

PageRank: Random Walks

Random walk interpretation

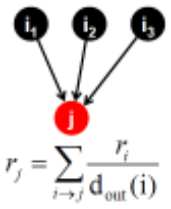
Someone randomly browse a network of Web pages.

- Starts by choosing a page at random, picking each page with equal probability.
- Then follows links for a sequence of *k steps: each step,* picking a random out-going link from the current page, and follow it to where it leads.

Claim: The probability of being at a page X after k steps of this random walk is precisely the PageRank of X after k applications of the Basic PageRank Update Rule.

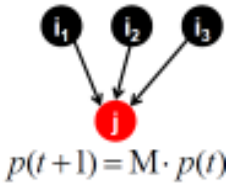
PageRank: Random Walks

- **Imagine a random web surfer:**
  - At any time t, surfer is on some page i
  - At time $t + 1$, the surfer follows an out-link from i uniformly at random
  - Ends up on some page j linked from i
  - Process repeats indefinitely



- **Let:**
  - $\boldsymbol{p}(\boldsymbol{t})$ … vector whose i^{th} coordinate is the prob. that the surfer is at page i at time t
  - So, $\boldsymbol{p}(\boldsymbol{t})$ is a probability distribution over pages

PageRank: Random Walks



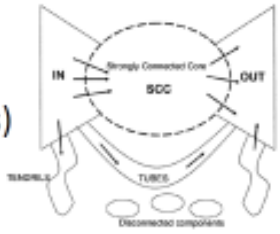
- **Where is the surfer at time *t+1*?**
  - Follows a link uniformly at random
    $p(t+1) = M \cdot p(t)$
- Suppose the random walk reaches a state
  $p(t+1) = M \cdot p(t) = p(t)$
  then $\boldsymbol{p}(t)$ is stationary distribution of a random walk
- Our original rank vector $\boldsymbol{r}$ satisfies $\boldsymbol{r} = \boldsymbol{M} \cdot \boldsymbol{r}$
  - So, $\boldsymbol{r}$ is a stationary distribution for the random walk

33

PageRank: Extensions

2 problems:

- **(1)** Some pages are **dead ends** (have no out-links)
  - Such pages cause importance to "leak out"

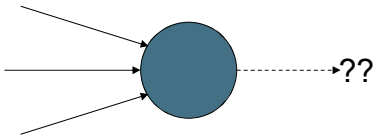


- **(2) Spider traps**
  (all out-links are within the group)
  - Eventually spider traps absorb all importance

34

PageRank: Αδιέξοδα

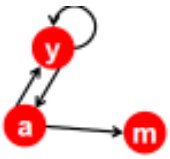
Αδιέξοδα (dead ends): σελίδες που δεν έχουν outlinks



Random walk can get stuck in dead-ends

PageRank: Αδιέξοδα

- **Power Iteration:**
  - Set $r_j = 1$
  - $r_j = \sum_{i \to j} \frac{r_i}{d_i}$
    - And iterate



	y	a	m
y	½	½	0
a	½	0	0
m	0	½	0

$r_y = r_y/2 + r_a/2$
$r_a = r_y/2$
$r_m = r_a/2$

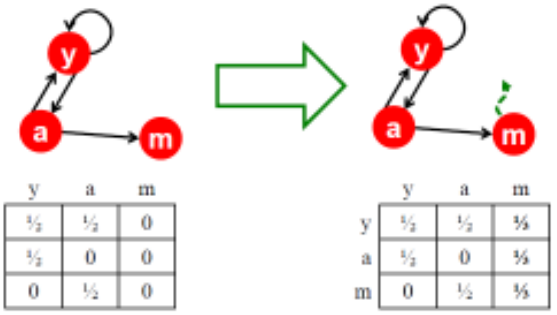
- **Example:**

$$\begin{pmatrix} r_y \\ r_a \\ r_m \end{pmatrix} = \begin{matrix} 1/3 & 2/6 & 3/12 & 5/24 & & 0 \\ 1/3 & 1/6 & 2/12 & 3/24 & \dots & 0 \\ 1/3 & 1/6 & 1/12 & 2/24 & & 0 \end{matrix}$$

Iteration 0, 1, 2, …

PageRank: Αδιέξοδα

- **Teleports:** Follow random teleport links with probability 1.0 from dead-ends
  - Adjust matrix accordingly



	y	a	m
y	½	½	0
a	½	0	0
m	0	½	0

	y	a	m
y	½	½	⅓
a	½	0	⅓
m	0	½	⅓

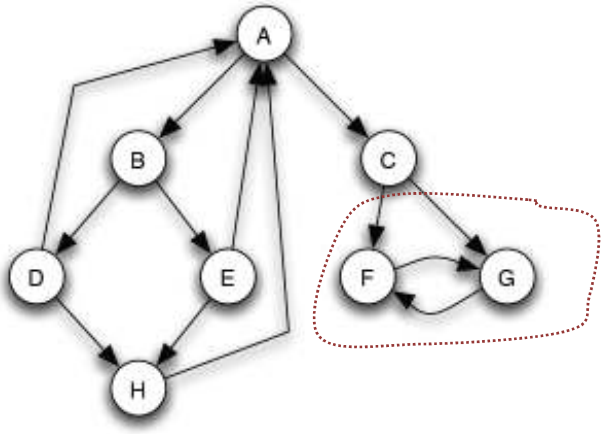
PageRank: Αδιέξοδα

- **Google's solution:** At each step, random surfer has two options:
  - With probability 1-β, follow a link at random
  - With probability β, jump to some random page
- **PageRank equation** [Brin-Page, 98]

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{n}$$

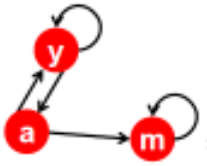
d_i ... out-degree of node i

PageRank: Spider Traps



PageRank: Spider Traps

- **Power Iteration:**
  - Set $r_j = 1$
  - $r_j = \sum_{i \to j} \frac{r_i}{d_i}$
    - And iterate



	y	a	m
y	½	½	0
a	½	0	0
m	0	½	1

$r_y = r_y/2 + r_a/2$
$r_a = r_y/2$
$r_m = r_a/2 + r_m$

- **Example:**

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 2/6 & 3/12 & 5/24 & & 0 \\ 1/3 & 1/6 & 2/12 & 3/24 & \dots & 0 \\ 1/3 & 3/6 & 7/12 & 16/24 & & 1 \end{matrix}$$

Iteration 0, 1, 2, …

PageRank: Spider Traps

Scaled PageRank Update Rule:
First apply the Basic PageRank Update Rule.
Then scale down all PageRank values by a factor of s.
This means that the total PageRank in the network has shrunk from 1 to s.
Divide the residual 1 − s units of PageRank equally over all nodes, giving (1 − s)/n to each.

PageRank: Spider Traps

Random walk with jumps

With probability *s, the walker* follows a random edge as before; and with probability 1 − *s, the walker jumps to a random* page anywhere in the network, choosing each node with equal probability

PageRank: Spider Traps

- **The Google solution for spider traps: At each time step, the random surfer has two options**
  - With prob. β, follow a link at random
  - With prob. 1-β, jump to some page uniformly at random
  - Common values for β are in the range 0.8 to 0.9
- **Surfer will teleport out of spider trap within a few time steps**

PageRank: Spectral Analysis

- **PageRank as a principal eigenvector**
  $r = M \cdot r$ or equivalently $r_j = \sum_i \frac{r_i}{d_i}$
- **But we really want:**
  $r_j = \beta \sum_i \frac{r_i}{d_i} + (1-\beta)\frac{1}{n}$
- **Let's define:**
  $M'_{ij} = \beta M_{ij} + (1-\beta)\frac{1}{n}$
- **Now we get what we want:**
  $\boldsymbol{r = M' \cdot r}$
- **What is $1-\beta$?**
  - In practice 0.15 (5 links and jump)

d_i ... out-degree of node i

Note: M is a sparse matrix but M' is dense (all entries $\neq 0$). In practice we never "materialize" M but rather we use the "sum" formulation

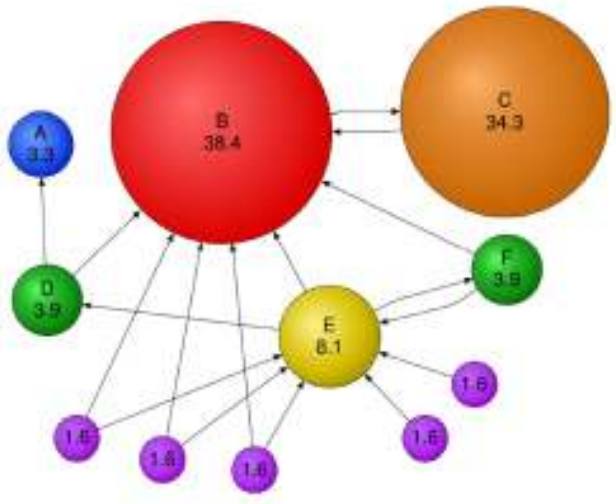
PageRank: Spectral Analysis

See P. Berkhin, *A Survey on PageRank Computing*, Internet Mathematics, 2005.

- **Input:** A and β
  - Adjacency matrix A of a directed graph with spider traps and dead ends
  - Parameter β
- **Output:** PageRank vector r
  - Set: $r_j^{(0)} = 1/n$
  - **Repeat until:** $\sum_j \left| r_j^{(t)} - r_j^{(t-1)} \right| < \varepsilon$
    - $\forall j$: $r'^{(t)}_j = \sum_{i \to j} \beta \frac{r_i^{(t-1)}}{d_i}$, if in-deg. of j is 0 then $r'^{(t)}_j = 0$
    - Now re-insert the leaked PageRank:
      $\forall j$: $r_j^{(t)} = r'^{(t)}_j + (1-S)/n$
      Where: $S = \sum_j r'^{(t)}_j$

45

PageRank: Example



46

Personalized PageRank

- **Goal:** Evaluate pages not just by popularity but by how close they are to the topic
- Teleporting can go to:
  - Any page with equal probability
    - (we used this so far)
  - A topic-specific set of "relevant" pages
    - Topic-specific (personalized) PageRank (S ...teleport set)

$$M'_{ij} = (1-\beta)\,M_{ij} + \beta/|S| \quad \text{if } i \in S$$
$$= (1-\beta)\,M_{ij} \quad \text{otherwise}$$

    - Useful for measuring "proximity" of other nodes to S

PageRank: Trust Rank

- Link Farms: networks of millions of pages design to focus PageRank on a few undeserving webpages
- To minimize their influence use a teleport set of trusted webpages
  - E.g., homepages of universities

Pagerank summary

- Preprocessing:
  - Given graph of links, build matrix **P**.
  - From it compute **a** – left eigenvector of **P**.
  - The entry a_i is a number between 0 and 1: the pagerank of page *i*.
- Query processing:
  - Retrieve pages meeting query.
  - Rank them by their pagerank.
  - But this rank order is query-*independent* …

The reality

- Pagerank is used in google and other engines, but is hardly the full story of ranking
  - Many sophisticated features are used
  - Some address specific query classes
  - Machine learned ranking (Lecture 19) heavily used
- Pagerank still very useful for things like crawl policy

HITS

HITS

Δύο βασικές διαφορές

- Κάθε σελίδα έχει δύο βαθμούς:
  - ένα βαθμό κύρους (authority rank) και
  - ένα κομβικό βαθμό (hub rank)
- Οι βαθμοί είναι θεματικοί

HITS: βασική ιδέα

Ας πούμε ότι ψάχνουμε για εφημερίδες (keyword: newspaper)

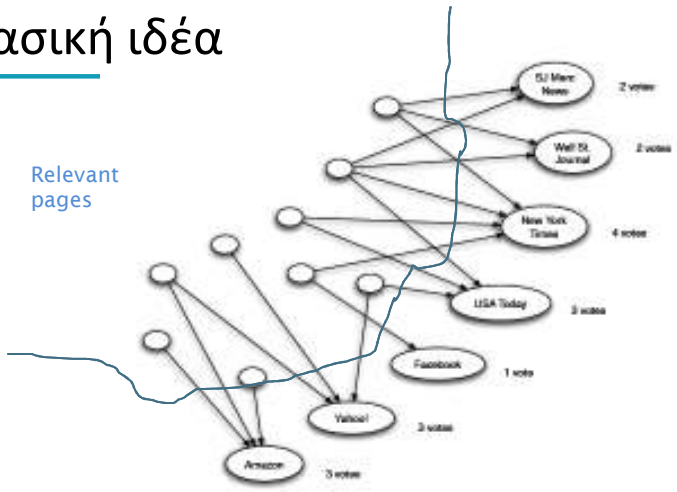
First collect a large number of pages that are *relevant* to the topic (e.g., using text-only IR)

The pages may not be actual web sites of a "newspaper" but we can use them to access the **"authority"** of a page on the topic

How?

- Relevant pages "vote" through their links
- Authority = #in-links from relevant pages

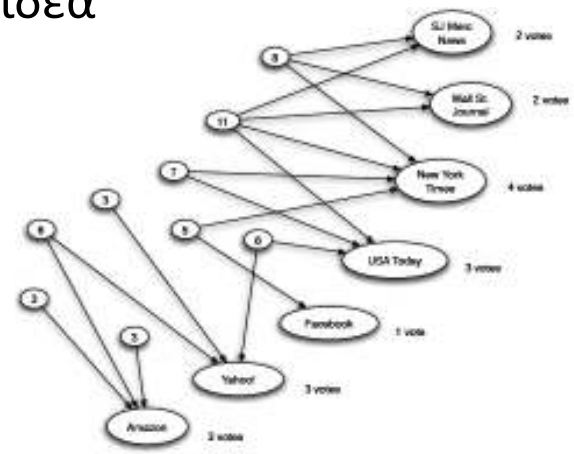
HITS: βασική ιδέα



"quality" of relevant pages?
Best – pages that compile lists of resources relevant to the topic (hubs)

HITS: βασική ιδέα

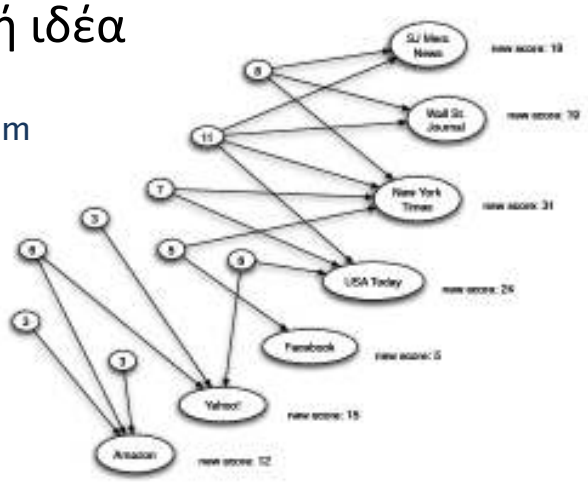
the principle of repeated improvement



Hubs:
Voted for many of the pages
value of a hub = sum of the votes received by all pages it voted for

HITS: βασική ιδέα

✓ Vote (link) from a good "hub" should count more!



Think about restaurant recommendations!!

HITS: βασική ιδέα

the principle of repeated improvement

Why stop here?

Recompute the score of the hubs using the improved scores of the authorities!

Repeat .. Recompute the score of the authories using the improved scores of the hubs!

…

HITS

Interesting pages fall into two classes:

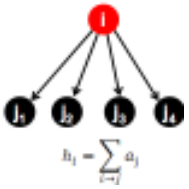
- Authorities: pages containing useful information (the prominent, highly endorsed answers to the queries)
  - Newspaper home pages
  - Course home pages
  - Home pages of auto manufacturers
- Hubs: pages that link to authorities (highly value lists)
  - List of newspapers
  - Course bulletin
  - List of US auto manufacturers

✓ A good hub links to many good authorities
✓ A good authority is linked from many good hubs

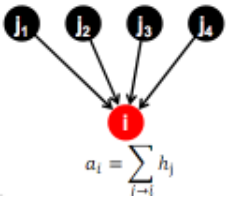
HITS: Algorithm

Each page p, has two scores

- A **hub score** (h) quality as an expert
  - Total sum of authority scores that it points to

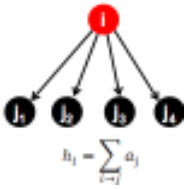


- An **authority score** (a) quality as content
  - Total sum of hub scores that point to it

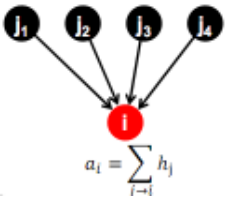


HITS: Algorithm

Authority Update Rule: For each page i, update a(i) to be the sum of the hub scores of all pages that point to it.



Hub Update Rule: For each page i, update h(i) to be the sum of the authority scores of all pages that it points to.



HITS: Algorithm

- Start with all hub scores and all authority scores equal to 1.
- Perform a sequence of k hub-authority updates. For each node:
  - First, apply the Authority Update Rule to the current set of scores.
  - Then, apply the Hub Update Rule to the resulting set of scores.
- At the end, hub and authority scores may be very large.

Normalize: divide each authority score by the sum of all authority scores, and each hub score by the sum of all hub scores.

61

High-level scheme

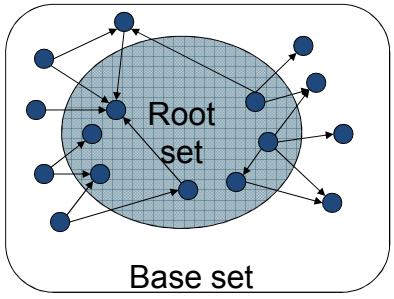
- Extract from the web a base set of pages that *could* be good hubs or authorities.
- From these, identify a small set of top hub and authority pages;

  →iterative algorithm.

Base set

- Given text query (say ***browser***), use a text index to get all pages containing ***browser.***
  - Call this the root set of pages.
- Add in any page that either
  - points to a page in the root set, or
  - is pointed to by a page in the root set.
- Call this the base set.

Visualization



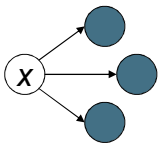
Distilling hubs and authorities

- Compute, for each page *x* in the base set, a hub score *h(x)* and an authority score *a(x).*
- Initialize: for all *x, h(x)←1; a(x) ←1;*
- Iteratively update all *h(x), a(x);* Key
- After iterations
  - output pages with highest *h()* scores as top hubs
  - highest *a()* scores as top authorities.

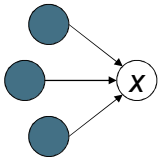
Iterative update

- Repeat the following updates, for all *x*:

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$



$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$



Scaling

- To prevent the *h()* and *a()* values from getting too big, can scale down after each iteration.
- Scaling factor doesn't really matter:
  - we only care about the *relative* values of the scores.

How many iterations?

- Claim: relative values of scores will converge after a few iterations:
  - in fact, suitably scaled, *h()* and *a()* scores settle into a steady state!
- In practice, ~5 iterations get you close to stability.

Japan Elementary Schools

Hubs

- schools
- LINK Page-13
- “ú–{‚ÌŠwZ
- a‰„¬ŠwZƒzƒyƒW
- 100 Schools Home Pages (English)
- K-12 from Japan 10/...rnet and Education)
- http://www...iglobe.ne.jp/~IKESAN
- ‚l‚f‚j¬ŠwZ‚U”N‚P‘g•¨Œê
- ÒŠ—'¬— § ÒŠ—“Œ¬ŠwZ
- Koulutus ja oppilaitokset
- TOYODA HOMEPAGE
- Education
- Cay's Homepage(Japanese)
- –y”ì¬ŠwZ‚ÌƒzƒyƒW
- UNIVERSITY
- ‰J—³¬ŠwZ DRAGON97-TOP
- Â‰ª¬ŠwZ‚T”N‚P‘gƒzƒyƒW
- ¶µ° é¼ÂÁ© ¥á¥Ë¥â¡¼ ¥á¥Ë¥â¡¼

Authorities

- The American School in Japan
- The Link Page
- ‰ªèš— § ˆä“c¬ŠwZƒzƒyƒW
- Kids' Space
- ˆÀés— § ˆÀé¼•”¬ŠwZ
- ‹{é‹³‘ç‘âŠw•'®¬ŠwZ
- KEIMEI GAKUEN Home Page (Japanese)
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- _“ÞìŒ § E‰¡•ls— § '†ì¼¬ŠwZ‚Ìƒy
- http://www...p/~m_maru/index.html
- fukui haruyama-es HomePage
- Torisu primary school
- goo
- Yakumo Elementary,Hokkaido,Japan
- FUZOKU Home Page
- Kamishibun Elementary School...

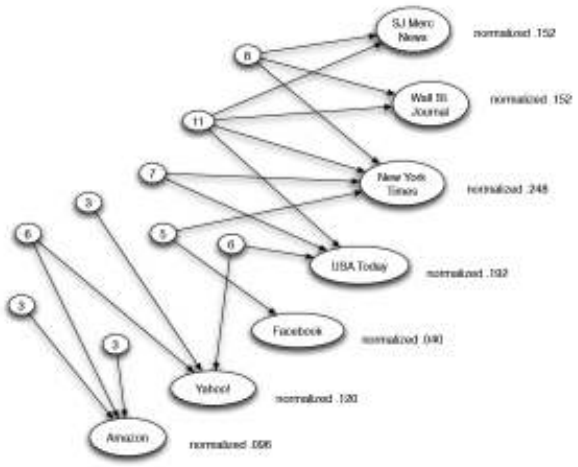
Things to note

- Pulled together good pages regardless of language of page content.
- Use *only* link analysis after base set assembled
  - iterative scoring is query-independent.
- Iterative computation after text index retrieval - significant overhead.

Issues

- Topic Drift
  - Off-topic pages can cause off-topic "authorities" to be returned
    - E.g., the neighborhood graph can be about a "super topic"
- Mutually Reinforcing Affiliates
  - Affiliated pages/sites can boost each others' scores
    - Linkage between affiliated pages is not a useful signal

HITS: Algorithm



72

HITS: Διανυσματική Αναπαράσταση

Adjacency matrix

Aij = 1 if i-j
0, otherwise

Two vectors
a: with authority ranks
h: with hup ranks

Rewrite in matrix form

- **h**=**Aa**.
- **a**=**A**^{t}**h**.

Recall A^t is the transpose of A.

Substituting, **h**=**AA**^{t}**h** and **a**=**A**^{t}**Aa**.

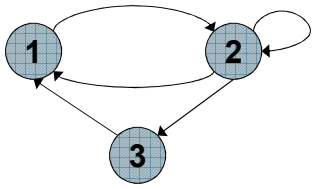
Thus, **h** is an eigenvector of **AA**^{t} and **a** is an eigenvector of **A**^{t}**A**.

Further, our algorithm is a particular, known algorithm for computing eigenvectors: the *power iteration* method.

Guaranteed to converge.

Proof of convergence

- $n \times n$ adjacency matrix **A**:
  - each of the n pages in the base set has a row and column in the matrix.
  - Entry $A_{ij} = 1$ if page i links to page j, else = 0.



	1	2	3
1	0	1	0
2	1	1	1
3	1	0	0

Hub/authority vectors

- View the hub scores *h()* and the authority scores *a()* as vectors with n components.
- Recall the iterative updates

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$

$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$

HITS: Διανυσματική Αναπαράσταση

- HITS converges to a single stable point
- Notation:
  - Vector $a = (a_1 \ldots, a_n)$, $h = (h_1 \ldots, h_n)$
  - Adjacency matrix $\boldsymbol{A}$ (n x n): $A_{ij} = 1$ if $i \rightarrow j$
- Then $h_i = \sum_{i \rightarrow j} a_j$
  can be rewriten as $h_i = \sum_j A_{ij} \cdot a_j$
- So: $\boldsymbol{h} = \boldsymbol{A} \cdot \boldsymbol{a}$
- And likewise: $\boldsymbol{a} = \boldsymbol{A}^T \cdot \boldsymbol{h}$

HITS: Διανυσματική Αναπαράσταση

- HITS algorithm in vector notation:
  - Set: $a_i = h_i = \frac{1}{\sqrt{n}}$

  Repeat until convergence:
  - $h = A \cdot a$
  - $a = A^T \cdot h$
  - Normalize a and h
- Then: $\boldsymbol{a} = \boldsymbol{A}^T \cdot (\underbrace{\boldsymbol{A} \cdot \boldsymbol{a}}_{\text{new } h})$ (new a)
- Thus, in $2k$ steps:
  $a = (A^T \cdot A)^k \cdot a$
  $h = (A \cdot A^T)^k \cdot h$

Convergence criterion:

$$\sum_i \left(h_i^{(t)} - h_i^{(t-1)}\right)^2 < \varepsilon$$

$$\sum_i \left(a_i^{(t)} - a_i^{(t-1)}\right)^2 < \varepsilon$$

a is updated (in 2 steps):
$a = A^T(A\,a) = (A^T A)\,a$

h is updated (in 2 steps):
$h = A\,(A^T h) = (A\,A^T)\,h$

Repeated matrix powering

HITS: Spectral Analysis

- **Definition:**
  - Let $R \cdot x = \lambda \cdot x$
    for some scalar λ, vector x, matrix R
  - Then $\boldsymbol{x}$ **is an eigenvector**, and λ **is its eigenvalue**
- **Fact:**
  - If R is symmetric ($R_{ij} = R_{ji}$)
    (in our case $R = A^T \cdot A$ and $\mathrm{R} = A \cdot A^T$ are symmetric)
  - Then R has n orthogonal unit eigenvectors
    $w_1 \ldots w_n$ that form a basis (coordinate system) with
    eigenvalues $\lambda_1 \ldots \lambda_n \quad (|\lambda_i| \geq |\lambda_{i+1}|)$

HITS: Spectral Analysis

$Rx = \lambda x$

- Let's write x in coordinate system $w_1 \ldots w_n$
  $x = \lambda_i \alpha_i w_i$
  - x has coordinates $(\alpha_1, \ldots, \alpha_n)$
- **Suppose:** $\lambda_1 \ldots \lambda_n \quad (|\lambda_1| \geq \ldots \geq |\lambda_n|)$
- **Then:** $R^k x = \lambda^k x = \sum_i \lambda_i^k \alpha_i w_i$
- **As** $k \to \infty$, if we normalize
  $R^k x \to \lambda_1 \alpha_1 w_1$
  (contribution of all other coordinates $\to 0$)
  $\lim_{k \to \infty} \frac{\lambda_1^k}{\lambda_2^k} = \left(\frac{\lambda_1}{\lambda_2}\right)^k \to \infty$
- So, **authority** a is eigenvector of $\mathrm{R} = A^T A$
  associated with largest eigenvalue λ_1
- Similarly: **hub** h is eigenvector of $\mathrm{R} = A\, A^T$

HITS: Spectral Analysis

$Rx = \lambda x$

- Let's write x in coordinate system $w_1 \dots w_n$
  $x = \lambda_i \alpha_i w_i$
  - x has coordinates $(\alpha_1, \dots, \alpha_n)$
- Suppose: $\lambda_1 \dots \lambda_n \quad (|\lambda_1| \geq \dots \geq |\lambda_n|)$
- **Then:** $R^k x = \lambda^k x = \sum_i \lambda_i^k \alpha_i w_i$
- **As** $k \to \infty$, if we normalize
  $R^k x \to \lambda_1 \alpha_1 w_1$
  (contribution of all other coordinates $\to 0$)
  $\lim_{k\to\infty} \frac{\lambda_1^k}{\lambda_2^k} = \left(\frac{\lambda_1}{\lambda_2}\right)^k \to \infty$
- So, **authority** a is eigenvector of $\mathrm{R} = A^T A$ associated with largest eigenvalue λ_1
- Similarly: **hub** h is eigenvector of $\mathrm{R} = A A^T$

81

PageRank vs HITS

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.
- PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.
  - We could also apply HITS to the entire web and PageRank to a small base set.
- Claim: On the web, a good hub almost always is also a good authority.
- The actual difference between PageRank ranking and HITS ranking is therefore not as large as one might expect.

82

Περίληψη

- Anchor text: What exactly are links on the web and why are they important for IR?
- PageRank: the original algorithm that was used for link-based ranking on the web
- Hubs & Authorities: an alternative link-based ranking algorithm

Google's official description of PageRank: *PageRank reflects our view of the importance of web pages by considering more than 500 million variables and 2 billion terms. Pages that believe are important pages receive a higher PageRank and are more likely to appear at the top of the search results.*

ΤΕΛΟΣ 10ου Μαθήματος

Ερωτήσεις?

Χρησιμοποιήθηκε κάποιο υλικό από:
- ✓*Pandu Nayak and Prabhakar Raghavan, CS276:Information Retrieval and Web Search (Stanford)*
- ✓*Hinrich Schütze and Christina Lioma, Stuttgart IIR class*