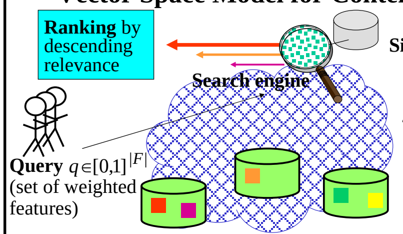
1 Topic-specific Authority Ranking

Vector Space Model for Content Relevance

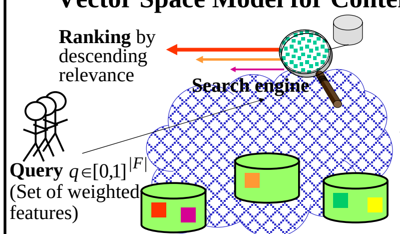


Similarity metric:

$$sim(d_i, q) := \frac{\sum_{j=1}^{|F|} d_{ij} q_j}{\sqrt{\sum_{j=1}^{|F|} d_{ij}^2 \sum_{j=1}^{|F|} q_j^2}}$$

Documents are **feature vectors** $d_i \in [0,1]^{|F|}$

Vector Space Model for Content Relevance



Similarity metric:

$$sim(d_i, q) := \frac{\sum_{j=1}^{|F|} d_{ij} q_j}{\sqrt{\sum_{j=1}^{|F|} d_{ij}^2 \sum_{j=1}^{|F|} q_j^2}}$$

Documents are **feature vectors** $d_i \in [0,1]^{|F|}$

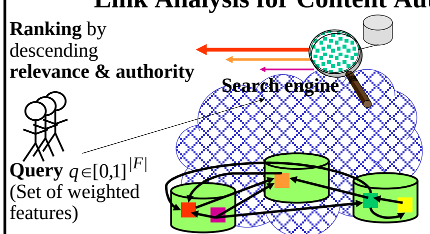
e.g., using:

$$d_{ij} := w_{ij} / \sqrt{\Sigma_k w_{ik}^2}$$

$$w_{ij} := \frac{freq(f_j, d_i)}{\max_k freq(f_k, d_i)} \log \frac{\#docs}{\#docs\ with\ f_i}$$

tf*idf formula

Link Analysis for Content Authority



+ Consider in-degree and out-degree of Web nodes:
Authority Rank (d_i) :=
Stationary visit probability [d_i]
in random walk on the Web

Reconciliation of relevance and authority by ad hoc weighting

1.1 Improving Precision by Authority Scores

Goal:
Higher ranking of URLs with high authority regarding volume, significance, freshness, authenticity of information content
→ improve precision of search results

Approaches (all interpreting the Web as a directed graph G):
- citation or impact rank (q) ~ indegree (q)
- Page rank (by Lawrence Page)
- HITS algorithm (by Jon Kleinberg)

Combining relevance and authority ranking:
- by weighted sum with appropriate coefficients (Google)
- by initial relevance ranking and iterative improvement via authority ranking (HITS)

Page Rank r(q)

given: directed Web graph G=(V,E) with |V|=n and adjacency matrix A: A_{ij} = 1 if (i,j)∈E, 0 otherwise

Idea: $r(q) \sim \sum_{(p,q) \in G} r(p) / outdegree(p)$

Def.: $r(q) = \varepsilon / n + (1-\varepsilon) \sum_{(p,q) \in G} r(p) / outdegree(p)$ with $0 < \varepsilon \le 0.25$

Theorem: With A'_{ij} = 1/outdegree(i) if (i,j)∈E, 0 otherwise:

$$\vec{r} = \frac{\vec{\varepsilon}}{n} + (1-\varepsilon) A' \vec{r} \iff \frac{1}{1-\varepsilon} \vec{r} = \left(\frac{\vec{\varepsilon}}{(1-\varepsilon)n} \vec{1}^T + A' \right) \vec{r}$$

i.e. r is Eigenvector of a modified adjacency matrix

Iterative computation of r(q) (after large Web crawl):
- Initialization: r(q) := 1/n
- Improvement by evaluating recursive equation of definition; typically converges after about 100 iterations

Digression: Markov Chains

A time-discrete finite-state **Markov chain** is a pair (Σ, p) with a state set Σ={s1, ..., sn} and a transition probability function p: Σ×Σ → [0,1] with the property $\sum_j p_{ij} = 1$ for all i where p_{ij} := p(si, sj).

A Markov chain is called **ergodic (stationary)** if for each state sj the limit $\pi_j := \lim_{t\to\infty} p_{ij}^{(t)}$ exists and is independent of si, with $p_{ij}^{(t)} := \sum_k p_{ik}^{(t-1)} p_{kj}$ for t>1 and $p_{ij}^{(t)} := p_{ij}$ for t=1.

For an ergodic finite-state Markov chain, the stationary state probabilities p_j can be computed by solving the linear equation system: $\pi_j = \sum_i \pi_i \, p_{ij}$ *for all j* and $\sum_j \pi_j = 1$

in matrix notation: $\Pi_{(1\times n)} = \Pi_{(1\times n)} \cdot P_{(n\times n)}$ and $\Pi_{(1\times n)} \vec{1}_{(n\times 1)} = 1$

can be approximated by power iteration: $\Pi_{(1\times n)}^{(i)} = \Pi_{(1\times n)}^{(i-1)} \cdot P_{(n\times n)}$

More on Markov Chains

A **stochastic process** is a family of random variables {X(t) | t ∈ T}.
T is called parameter space, and the domain M of X(t) is called state space. T and M can be discrete or continuous.

A stochastic process is called **Markov process** if for every choice of $t_1, ..., t_{n+1}$ from the parameter space and every choice of $x_1, ..., x_{n+1}$ from the state space the following holds:

$$P[\, X(t_{n+1}) = x_{n+1} \,|\, X(t_1) = x_1 \wedge X(t_2) = x_2 \wedge ... \wedge X(t_n) = x_n \,]$$
$$= P[\, X(t_{n+1}) = x_{n+1} \,|\, X(t_n) = x_n \,]$$

A Markov process with discrete state space is called **Markov chain**.
A canonical choice of the state space are the natural numbers.
Notation for Markov chains with discrete parameter space: X_n rather than $X(t_n)$ with n = 0, 1, 2, ...

Properties of Markov Chains with Discrete Parameter Space (1)

The Markov chain Xn with discrete parameter space is

homogeneous if the transition probabilities pij := $P[X_{n+1} = j \mid X_n = i]$ are independent of n

irreducible if every state is reachable from every other state with positive probability:

$$\sum_{n=1}^{\infty} P[\, X_n = j \,|\, X_0 = i \,] > 0 \quad \text{for all i, j}$$

aperiodic if every state i has period 1, where the period of i is the gcd of all (recurrence) values n for which

$$P[\, X_n = i \wedge X_k \neq i \text{ for } k = 1,...,n-1 \,|\, X_0 = i \,] > 0$$

Properties of Markov Chains with Discrete Parameter Space (2)

The Markov chain Xn with discrete parameter space is

positive recurrent if for every state i the recurrence probability is 1 and the mean recurrence time is finite:

$$\sum_{n=1}^{\infty} P[\, X_n = i \wedge X_k \neq i \text{ for } k = 1,...,n-1 \,|\, X_0 = i \,] = 1$$
$$\sum_{n=1}^{\infty} n\, P[\, X_n = i \wedge X_k \neq i \text{ for } k = 1,...,n-1 \,|\, X_0 = i \,] < \infty$$

ergodic if it is homogeneous, irreducible, aperiodic, and positive recurrent.

Results on Markov Chains with Discrete Parameter Space (1)

For the **n-step transition probabilities**

$p_{ij}^{(n)} := P[\, X_n = j \,|\, X_0 = i \,]$ the following holds:

$$p_{ij}^{(n)} = \sum_k p_{ik}^{(n-1)} p_{kj} \quad \text{with} \quad p_{ij}^{(1)} := p_{ik}$$
$$= \sum_k p_{ik}^{(n-l)} p_{kj}^{(l)} \quad \text{for } 1 \le l \le n-1$$

in matrix notation: $P^{(n)} = P^n$

For the **state probabilities after n steps**

$\pi_j^{(n)} := P[\, X_n = j \,]$ the following holds:

$\pi_j^{(n)} = \sum_i \pi_i^{(0)} p_{ij}^{(n)}$ with initial state probabilities $\pi_i^{(0)}$

in matrix notation: $\Pi^{(n)} = \Pi^{(0)} P^{(n)}$ *(Chapman-Kolmogorov equation)*

Results on Markov Chains with Discrete Parameter Space (2)

Every homogeneous, irreducible, aperiodic Markov chain with a finite number of states is positive recurrent and ergodic.

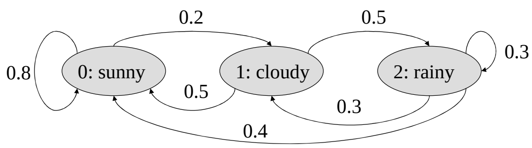
For every ergodic Markov chain there exist **stationary state probabilities** $\pi_j := \lim_{n\to\infty} \pi_j^{(n)}$
These are independent of $\Pi^{(0)}$
and are the solutions of the following system of linear equations:

$$\pi_j = \sum_i \pi_i \, p_{ij} \quad \text{for all } j \qquad \textit{(balance equations)}$$
$$\sum_j \pi_j = 1$$

in matrix notation: $\Pi = \Pi P$ with 1×n row vector Π

$$\Pi \vec{1} = 1$$

Markov Chain Example



$\pi 0 = 0.8\ \pi 0 + 0.5\ \pi 1 + 0.4\ \pi 2$
$\pi 1 = 0.2\ \pi 0 + 0.3\ \pi 2$
$\pi 2 = 0.5\ \pi 1 + 0.3\ \pi 2$
$\pi 0 + \pi 1 + \pi 2 = 1$

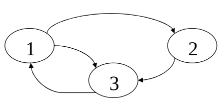
$\Rightarrow \pi 0 = 330/474 \approx 0.696$
$\pi 1 = 84/474 \approx 0.177$
$\pi 2 = 10/79 \approx 0.126$

Page Rank as a Markov Chain Model

Model a **random walk** of a Web surfer as follows:
• follow outgoing hyperlinks with uniform probabilities
• perform „random jump“ with probability ε
→ ergodic Markov chain
The Page rank of a URL is the stationary visiting probability of URL in the above Markov chain.
Further generalizations have been studied
(e.g. random walk with back button etc.)

Drawback of Page-Rank method:
Page Rank is query-independent and orthogonal to relevance

Example: Page Rank Computation



$\varepsilon = 0.2$

$$P = \begin{pmatrix} 0.0 & 0.5 & 0.5 \\ 0.1 & 0.0 & 0.9 \\ 0.9 & 0.1 & 0.0 \end{pmatrix}$$

$$\Pi^{(0)} \approx \begin{pmatrix} 0.333 \\ 0.333 \\ 0.333 \end{pmatrix}^T \Rightarrow \Pi^{(1)} \approx \begin{pmatrix} 0.333 \\ 0.200 \\ 0.466 \end{pmatrix}^T \Rightarrow \Pi^{(2)} \approx \begin{pmatrix} 0.439 \\ 0.212 \\ 0.346 \end{pmatrix}^T \Rightarrow \Pi^{(3)} \approx \begin{pmatrix} 0.332 \\ 0.253 \\ 0.401 \end{pmatrix}^T$$

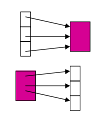
$$\Rightarrow \Pi^{(4)} \approx \begin{pmatrix} 0.385 \\ 0.176 \\ 0.527 \end{pmatrix}^T \Rightarrow \Pi^{(5)} \approx \begin{pmatrix} 0.491 \\ 0.244 \\ 0.350 \end{pmatrix}^T$$

$\pi 1 = 0.1\ \pi 2 + 0.9\ \pi 3$
$\pi 2 = 0.5\ \pi 1 + 0.1\ \pi 3$
$\pi 3 = 0.5\ \pi 1 + 0.9\ \pi 2$
$\pi 1 + \pi 2 + \pi 3 = 1$

$\Rightarrow \pi 1 \approx 0.3776,\ \pi 2 \approx 0.2282,\ \pi 3 \approx 0.3942$

HITS Algorithm: Hyperlink-Induced Topic Search (1)

Idea:
Determine • good content sources: **Authorities** (high indegree)
• good link sources: **Hubs** (high outdegree)



Find • better authorities that have good hubs as predecessors
• better hubs that have good authorities as successors

For Web graph G=(V,E) define for nodes p, q ∈V

authority score $x_q = \sum_{(p,q)\in E} y_p$ and

hub score $y_p = \sum_{(p,q)\in E} x_q$

HITS Algorithm (2)

Authority and hub scores in matrix notation:

$\vec{x} = A^T \vec{y}$ $\quad\quad$ $\vec{y} = A\vec{x}$

Iteration with adjacency matrix A:

$\vec{x} := A^T \vec{y} := A^T A \vec{x}$ $\quad\quad$ $\vec{y} := A\vec{x} := AA^T \vec{y}$

x and y are Eigenvectors of A^TA and AA^T, resp.

Intuitive interpretation:

$M^{(auth)} := A^T A$ is the cocitation matrix: $M^{(auth)}_{ij}$ is the number of nodes that point to both i and j

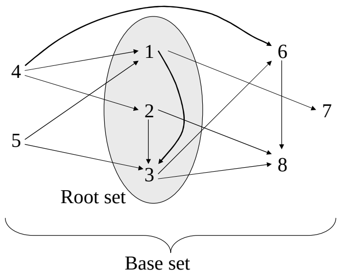
$M^{(hub)} := AA^T$ is the coreference (bibliographic-coupling) matrix $M^{(hub)}_{ij}$ is the number of nodes to which both i and j point

Implementation of the HITS Algorithm

1) Determine sufficient number (e.g. 50-200) of „root pages“ via relevance ranking (e.g. using tf*idf ranking)
2) Add all successors of root pages
3) For each root page add up to d predecessors
4) Compute iteratively the authority and hub scores of this „base set“ (of typically 1000-5000 pages) with initialization $x_q := y_p := 1$ / |base set| and L1 normalization after each iteration
→ converges to principal Eigenvector (Eigenvector with largest Eigenvalue (in the case of multiplicity 1)
5) Return pages in descending order of authority scores (e.g. the 10 largest elements of vector x)

Drawback of HITS algorithm:
relevance ranking within root set is not considered

Example: HITS Algorithm



Improved HITS Algorithm

Potential weakness of the HITS algorithm:
• irritating links (automatically generated links, spam, etc.)
• topic drift (e.g. from „Jaguar car“ to „car“ in general)

Improvement:
• Introduce edge weights:
 0 for links within the same host,
 1/k with k links from k URLs of the same host to 1 URL (xweight)
 1/m with m links from 1 URL to m URLs on the same host (yweight)
• Consider relevance weights w.r.t. query topic (e.g. tf*idf)

→ Iterative computation of

authority score $x_q = \sum_{(p,q) \in E} y_p * topic\ score(p) * xweight(p,q)$

hub score $y_p = \sum_{(p,q) \in E} x_q * topic\ score(q) * yweight(p,q)$

SALSA: Random Walk on Hubs and Authorities

View each node v of the link graph as two nodes v_h and v_a
Construct bipartite undirected graph G‘(V‘,E‘) from link graph G(V,E):
$V' = \{v_h \mid v \in V \text{ and outdegree}(v)>0\} \cup \{v_a \mid v \in V \text{ and indegree}(v)>0\}$
$E' = \{(v_h, w_a) \mid (v,w) \in E\}$

Stochastic hub matrix H: $h_{ij} = \sum_k \frac{1}{\deg ree(i_h)} \frac{1}{\deg ree(k_a)}$

for hubs i, j and k ranging over all nodes with $(i_h, k_a), (k_a, j_h) \in E'$

Stochastic authority matrix A: $a_{ij} = \sum_k \frac{1}{\deg ree(i_a)} \frac{1}{\deg ree(k_h)}$

for authorities i, j and k ranging over all nodes with $(i_a, k_h), (k_h, j_a) \in E'$

The corresponding Markov chains are ergodic on connected component

The stationary solutions for these Markov chains are:
$\pi[v_h]$ ~ outdegree(v) for H and $\pi[v_a]$ ~ indegree(v) for A

1.2 Towards Unified Framework (Ding et al.)

Literature contains plethora of variations on Page-Rank and HITS

Key points are:
• mutual reinforcement between hubs and authorities
• re-scale edge weights (normalization)

Unified notation (for link graph with n nodes):

L	- n×n link matrix, $L_{ij} = 1$ if there is an edge (i,j), 0 else
din	- n×1 vector with din_i = indegree(i), $Din_{n \times n}$ = diag(din)
dout	- n×1 vector with $dout_i$ = outdegree(i), $Dout_{n \times n}$ = diag(dout)
x	- n×1 authority vector
y	- n×1 hub vector
Iop	- operation applied to incoming links
Oop	- operation applied to outgoing links

HITS: x = Iop(y), y=Oop(x) with Iop(y) = $L^T y$, Oop(x) = Lx

Page-Rank: x = Iop(x) with Iop(x) = P^T x with $P^T = L^T Dout^{-1}$
or $P^T = \alpha L^T Dout^{-1} + (1-\alpha)(1/n)\, e\, e^T$

HITS and Page-Rank in the Framework

HITS: x = Iop(y), y=Oop(x) with Iop(y) = $L^T y$, Oop(x) = Lx

Page-Rank: x = Iop(x) with Iop(x) = P^T x with $P^T = L^T Dout^{-1}$
or $P^T = \alpha L^T Dout^{-1} + (1-\alpha)(1/n)\, e\, e^T$

Page-Rank-style computation with mutual reinforcement (SALSA):
x = Iop(y) with Iop(y) = P^T y with $P^T = L^T Dout^{-1}$
y = Oop(x) with Oop(x) = Q x with $Q = L\, Din^{-1}$

and other models of link analysis can be cast into this framework, too

A Familiy of Link Analysis Methods

General scheme: $Iop(\cdot) = Din^{-p} L^T Dout^{-q} (\cdot)$ and $Oop(\cdot) = Iop^T (\cdot)$

Specific instance ***Out-link normalized Rank (Onorm-Rank)***:
$Iop(\cdot) = L^T Dout^{-1/2} (\cdot)$, $Oop(\cdot) = Dout^{-1/2} L (\cdot)$
applied to x and y: x = Iop(y), y = Oop(x)

In-link normalized Rank (Inorm-Rank):
$Iop(\cdot) = Din^{-1/2} L^T (\cdot)$, $Oop(\cdot) = L\, Din^{-1/2} (\cdot)$

Symmetric normalized Rank (Snorm-Rank):
$Iop(\cdot) = Din^{-1/2} L^T Dout^{-1/2} (\cdot)$, $Oop(\cdot) = Dout^{-1/2} L\, Din^{-1/2} (\cdot)$

Some properties of Snorm-Rank:
$x = Iop(y) = Iop(Oop(x)) \rightarrow \lambda x = A^{(S)} x$
with $A^{(S)} = Din^{-1/2} L^T Dout^{-1} L\, Din^{-1/2}$
→ Solution: $\lambda = 1$, $x = din^{1/2}$
and analogously for hub scores: $\lambda y = H^{(S)} y \rightarrow \lambda = 1$, $y = dout^{1/2}$

Experimental Results

Construct neighborhood graph from result of query "star"
Compare authority-scoring ranks

HITS	Onorm-Rank	Page-Rank
1 www.starwars.com	www.starwars.com	www.starwars.com
2 www.lucasarts.com	www.lucasarts.com	www.lucasarts.com
3 www.jediknight.net	www.jediknight.net	www.paramount.com
4 www.sirstevesguide.com	www.paramount.com	www.4starads.com/romar
5 www.paramount.com	www.sirstevesguide.com	www.starpages.net
6 www.surfthe.net/swma/	www.surfthe.net/swma/	www.dailystarnews.com
7 insurrection.startrek.com	insurrection.startrek.com	www.state.mn.us
8 www.startrek.com	www.fanfix.com	www.star-telegram.com
9 www.fanfix.com	shop.starwars.com	www.starbulletin.com
10 www.physics.usyd.edu.au/ .../starwars	www.physics.usyd.edu.au/ .../starwars	www.kansascity.com
		...
		19 www.jediknight.net
		21 insurrection.startrek.cc
		23 www.surfthe.net/swma

Bottom line:
Differences between all kinds of authority ranking methods are fairly minor !

1.3 Topic-specific Page-Rank (Haveliwala 2002)

Given: a (small) set of topics c_k, each with a set T_k of authorities
(taken from a directory such as ODP (www.dmoz.org)
or bookmark collection)

Key idea :
change the Page-Rank random walk by biasing the
random-jump probabilities to the topic authorities T_k:

$\vec{r}_k = \varepsilon\, \vec{p}_k + (1-\varepsilon) A' \vec{r}_k$ with A'_{ij} = 1/outdegree(i) for (i,j)∈E, 0 else
with $(p_k)_j = 1/|T_k|$ for $j \in T_k$, 0 else (instead of $p_j = 1/n$)

Approach:
1) Precompute topic-specific Page-Rank vectors r_k
2) Classify user query q (incl. query context) w.r.t. each topic c_k
→ probability $w_k := P[c_k \mid q]$
3) Total authority score of doc d is $\sum_k w_k\, r_k(d)$

Digression: Naives Bayes Classifier with Bag-of-Words Model

estimate: $P[d \in c_k \mid d \text{ has } \vec{f}\,] \sim P[\,\vec{f} \mid d \in c_k\,]\, P[\,d \in c_k\,]$
with term frequency vector $\vec{f}$

$= \Pi_{i=1}^{m} P[\, f_i \mid d \in c_k\,]\, P[\,d \in c_k\,]$ with feature independence

$= \Pi_{i=1}^{m} \binom{length(d)}{f_i} p_{ik}^{\,f_i} (1-p_{ik})^{length(d)-f_i}\quad p_k$
with binomial distribution of each feature

or: $= \binom{length(d)}{f_1\, f_2 \ldots f_m} p_{1k}^{\,f_1}\, p_{2k}^{\,f_2} \ldots p_{mk}^{\,f_m}\quad p_k$
with multinomial distribution of feature vectors and

with $\binom{n}{k_1\, k_2 \ldots k_m} := \frac{n!}{k_1!\, k_2! \ldots k_m!}$ $\qquad \sum_{i=1}^{m} f_i = length(d)$

Example for Naive Bayes

3 classes: c1 – Algebra, c2 – Calculus, c3 – Stochastics
8 terms, 6 training docs d1, ..., d6: 2 for each class
⇒ p1=2/6, p2=2/6, p3=2/6

	group f1	homomorphism f2	vector f3	integral f4	limit f5	variance f6	probability f7	dice f8
d1:	3	2	0	0	0	0	0	1
d2:	1	2	3	0	0	0	0	0
d3:	0	0	0	3	3	0	0	0
d4:	0	0	1	2	2	0	1	0
d5:	0	0	0	1	1	2	2	0
d6:	1	0	1	0	0	0	2	2

	Algebra k=1	Calculus k=2	Stochastics k=3
p1k	4/12	0	1/12
p2k	4/12	0	0
p3k	3/12	1/12	1/12
p4k	0	5/12	1/12
p5k	0	5/12	1/12
p6k	0	0	2/12
p7k	0	1/12	4/12
p8k	1/12	0	2/12

without smoothing for simple calculation

Example of Naive Bayes (2)

classification of d7: (0 0 1 2 0 0 3 0)

$$P[\,\vec{f} \mid d \in c_k\,]\, P[\,d \in c_k\,] = \binom{length(d)}{f_1\, f_2 \ldots f_m} p_{1k}^{\,f_1}\, p_{2k}^{\,f_2} \ldots p_{mk}^{\,f_m}\quad p_k$$

for k=1 (Algebra): $= \binom{6}{1\;2\;3} \left(\frac{3}{12}\right)^1 0^2\, 0^3\ \frac{2}{6} \qquad = 0$

for k=2 (Calculus): $= \binom{6}{1\;2\;3} \left(\frac{1}{12}\right)^1 \left(\frac{5}{12}\right)^2 \left(\frac{1}{12}\right)^3\ \frac{2}{6} \qquad = 20 * \frac{25}{12^6}$

for k=3 (Stochastics): $= \binom{6}{1\;2\;3} \left(\frac{1}{12}\right)^1 \left(\frac{1}{12}\right)^2 \left(\frac{4}{12}\right)^3\ \frac{2}{6} \qquad = 20 * \frac{64}{12^6}$

Result: assign d7 to class C3 (Stochastics)

Experimental Evaluation: Quality Measures

Setup: based on Stanford WebBase (120 Mio. pages, Jan. 2001)
contains ca. 300 000 out of 3 Mio. ODP pages
considered 16 top-level ODP topics
link graph with 80 Mio. nodes of size 4 GB
on 1.5 GHz dual Athlon with 2.5 GB memory and 500 GB RAID
25 iterations for all 16+1 PR vectors took 20 hours
random-jump prob. ε set to 0.25 (could be topic-specific, too ?)
35 test queries: classical guitar, lyme disease, sushi, etc.

Quality measures: consider top k of two rankings τ1 and τ2 (k=20)

- ***overlap similarity OSim*** (τ1,τ2) = | top(k,τ1) ∩ top(k,τ2) | / k
- ***Kendall's τ measure KSim*** (τ1,τ2) =

$$\frac{|\{(u,v) \mid u,v \in U, u \neq v, \text{ and } \tau 1, \tau 2 \text{ agree on relative order of } u,v\}|}{|U| \cdot (|U|-1)}$$

with U = top(k,τ1) ∪ top(k,τ2)

Experimental Evaluation Results (1)

• Ranking similarities between most similar PR vectors:

	OSim	KSim
(Games, Sports)	0.18	0.13
(No Bias, Regional)	0.18	0.12
(Kids&Teens, Society)	0.18	0.11
(Health, Home)	0.17	0.12
(Health, Kids&Teens)	0.17	0.11

• User-assessed precision at top 10 (# relevant docs / 10) with 5 users:

	No Bias	Topic-sensitive
alcoholism	0.12	0.7
bicycling	0.36	0.78
death valley	0.28	0.5
HIV	0.58	0.41
Shakespeare	0.29	0.33
micro average	0.276	0.512

Experimental Evaluation Results (2)

• Top 3 for query "bicycling"
(classified into sports with 0.52, regional 0.13, health 0.07)

	No Bias	Recreation	Sports
1	www.RailRiders.com	www.gorp.com	www.multisports.com
2	www.waypoint.org	www.GrownupCamps.com	www.BikeRacing.com
3	www.gorp.com	www.outdoor-pursuits.com	www.CycleCanada.com

• Top 5 for query context "blues" (user picks entire page)
(classified into arts with 0.52, shopping 0.12, news 0.08)

	No Bias	Arts	Health
1	news.tucows.com	www.britannia.com	www.baltimorepsych.com
2	www.emusic.com	www.bandhunt.com	www.ncpamd.com/seasonal
3	www.johnholleman.com	www.artistinformation.com	www.ncpamd.com/Women's_
4	www.majorleaguebaseball	www.billboard.com	www.wingofmadness.com
5	www.mp3.com	www.soul-patrol.com	www.countrynurse.com

Efficiency of Page-Rank Computation (1)

Speeding up convergence of the Page-Rank iterations

Solve Eigenvector equation $\lambda x = Ax$
(with dominant Eigenvalue λ_1=1 for ergodic Markov chain)
by power iteration: $x^{(i+1)} = Ax^{(i)}$ until $||x^{(i+1)} - x^{(i)}||_1$ is small enough

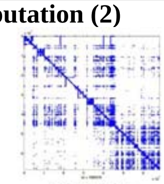
Write start vector $x^{(0)}$ in terms of Eigenvectors $u_1, ..., u_m$:
$x^{(0)} = u_1 + \alpha_2 u_2 + ... + \alpha_m u_m$
$x^{(1)} = Ax^{(0)} = u_1 + \alpha_2 \lambda_2 u_2 + ... + \alpha_m \lambda_m u_m$ with $\lambda_1 - |\lambda_2| = \varepsilon$ (jump prob.)
$x^{(n)} = A^n x^{(0)} = u_1 + \alpha_2 \lambda_2^n u_2 + ... + \alpha_m \lambda_m^n u_m$

Aitken Δ^2 extrapolation:
assume $x^{(k-2)} \approx u_1 + \alpha_2 u_2$ (disregarding all "lesser" EVs)
$\rightarrow x^{(k-1)} \approx u_1 + \alpha_2 \lambda_2 u_2$ and $x^{(k)} \approx u_1 + \alpha_2 \lambda_2^2 u_2$
$\rightarrow$ after step k: solve for u_1 and u_2 and recompute $x^{(k)} := u_1 + \alpha_2 \lambda_2^2 u_2$

can be extended to quadratic extrapolation using first 3 EVs
speeds up convergence by factor of 0.3 to 3

Efficiency of Page-Rank Computation (2)

Exploit block structure of the link graph:
1) partitition link graph by domain names
2) compute local PR vector of pages within each block $\rightarrow$ LPR(i) for page i
3) compute block rank of each block:
   a) block link graph $B_{IJ} = \sum_{i \in I, j \in J} A_{ij} \cdot LPR(i)$
   b) run PR computation on B $\rightarrow$ BR(I) for block I
4) Approximate global PR vector using LPR and BR:
   a) set $x_j^{(0)}$:= LPR(j) · BR(J) where J is the block that contains j
   b) run PR computation on A


(b) Stanford/Berkeley

speeds up convergence by factor of 2 in good "block cases"
unclear how effective it would be on Geocities, AOL, T-Online, etc.

Much adoo about nothing ?
Couldn't we simply initialize the PR vector with indegrees?

Efficiency of Storing Page-Rank Vectors

Memory-efficient encoding of PR vectors
(important for large number of topic-specific vectors)

16 topics * 120 Mio. pages * 4 Bytes would cost 7.3 GB

Key idea:
• map real PR scores to n cells and encode cell no into ceil($\log_2$ n) bits
• approx. PR score of page i is the mean score of the cell that contains i
• should use non-uniform partitioning of score values to form cells

Possible encoding schemes:
• ***Equi-depth partitioning***: choose cell boundaries such that $\sum_{i \in cell\ j} PR(i)$ is the same for each cell
• ***Equi-width partitioning with log values***: first transform all PR values into log PR, then choose equi-width boundaries
• Cell no. could be variable-length encoded (e.g., using Huffman code)

Literature

- Chakrabarti: Chapter 7
- J.M. Kleinberg: Authoritative Sources in a Hyperlinked Environment, Journal of the ACM Vol.46 No.5, 1999
- S Brin, L. Page: The Anatomy of a Large-Scale Hypertextual Web Search Engine, WWW Conference, 1998
- K. Bharat, M. Henzinger: Improved Algorithms for Topic Distillation in a Hyperlinked Environment, SIGIR Conference, 1998
- R. Lempel, S. Moran: SALSA: The Stochastic Approach for Link-Structure Analysis, ACM Transactions on Information Systems Vol. 19 No.2, 2001
- A. Borodin, G.O. Roberts, J.S. Rosenthal, P. Tsaparas: Finding Authorities and Hubs from Link Structures on the World Wide Web, WWW Conference, 2001
- C. Ding, X. He, P. Husbands, H. Zha, H. Simon: PageRank, HITS, and a Unified Framework for Link Analysis,SIAM Int. Conf. on Data Mining, 2003.
- Taher Haveliwala: Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search, IEEE Transactions on Knowledge and Data Engineering to appear in 2003.
- S.D. Kamvar, T.H. Haveliwala, C.D. Manning, G.H. Golub: Extrapolation Methods for Accelerating PageRank Computations, WWW Conference, 2003
- S.D. Kamvar, T.H. Haveliwala, C.D. Manning, G.H. Golub: Exploiting the Block Structure of the Web for Computing PageRank, Stanford Technical Report, 2003