# Information retrieval

Lecture 9

# Recap and today's topics

- Last lecture
  - web search overview
  - pagerank
- Today
  - more sophisticated link analysis
  - using links + content

# Pagerank recap

- Pagerank computation
  - Random walk on the web graph
  - Teleport operation to get unstuck from dead ends
  - $\Rightarrow$ Steady state visit rate for each web page
  - Call this its <u>pagerank</u> score
    - computed from an eigenvector computation (linear system solution)

# Pagerank recap

- Pagerank usage
  - Get pages matching text query
  - Return them in order of pagerank scores
  - This order is query-independent
  - Can combine arithmetically with text-based scores
- Pagerank is a global property
  - Your pagerank score depends on “everybody” else
  - Harder to spam than simple popularity counting

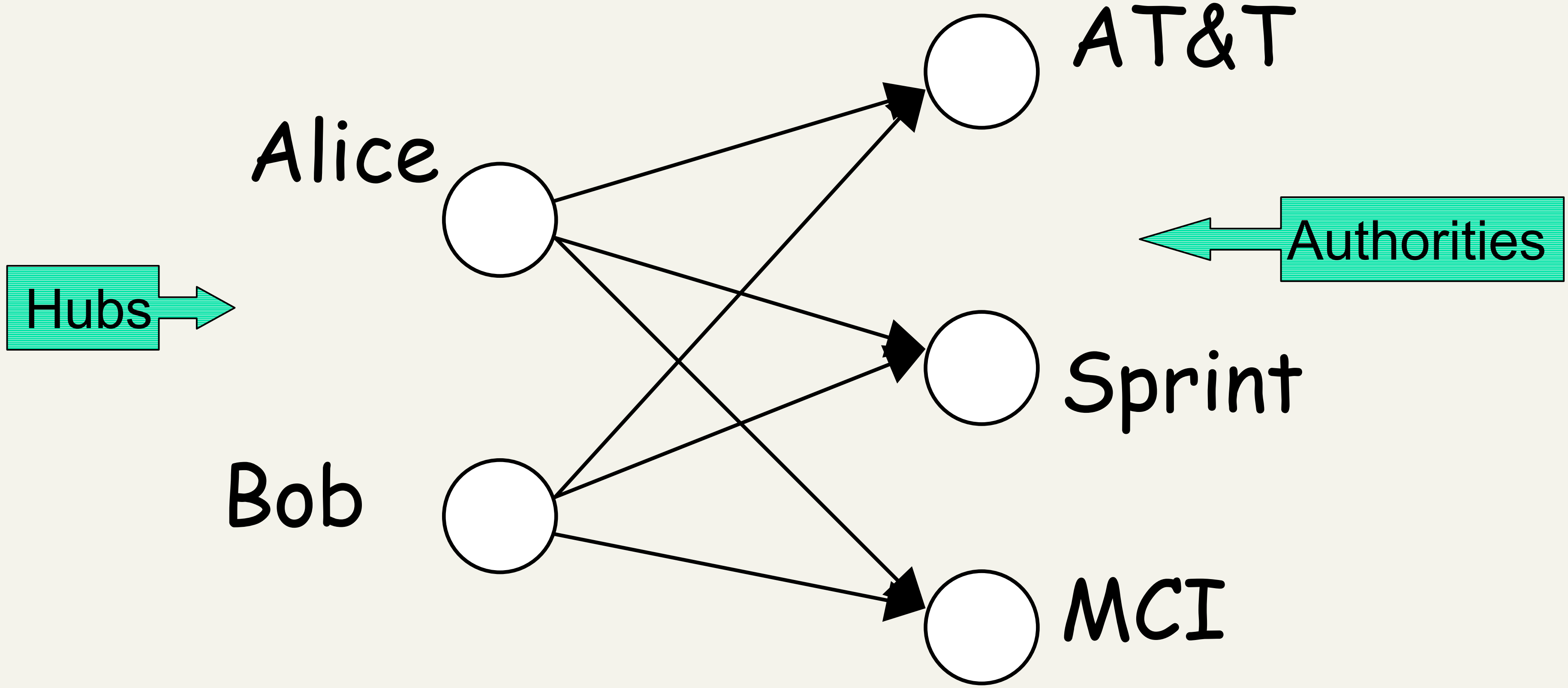
# Hyperlink-Induced Topic Search (HITS) - Klei98

- In response to a query, instead of an ordered list of pages each meeting the query, find <u>two</u> sets of inter-related pages:
  - *Hub pages* are good lists of links on a subject.
    - e.g., “Bob’s list of cancer-related links.”
  - *Authority pages* occur recurrently on good hubs for the subject.
- Best suited for “broad topic” queries rather than for page-finding queries.
- Gets at a broader slice of common *opinion*.

# Hubs and Authorities

- Thus, a good hub page for a topic *points* to many authoritative pages for that topic.
- A good authority page for a topic is *pointed* to by many good hubs for that topic.
- Circular definition - will turn this into an iterative computation.

# The hope




***Long distance telephone companies***

# High-level scheme

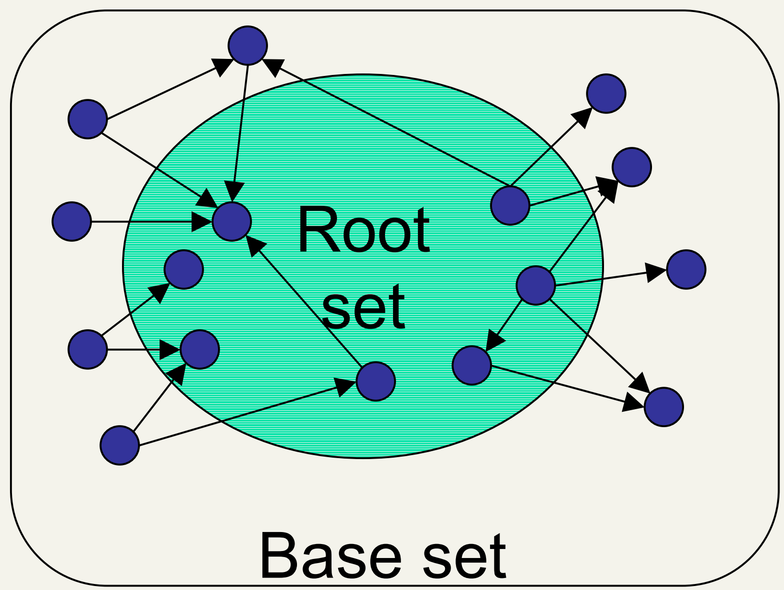
- Extract from the web a <u>base set</u> of pages that *could* be good hubs or authorities.
- From these, identify a small set of top hub and authority pages;

  → iterative algorithm.

# Base set

- Given text query (say *browser*), use a text index to get all pages containing *browser.*
  - Call this the <u>root set</u> 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 <u>base set</u>.

# Visualization



# Assembling the base set

- Root set typically 200-1000 nodes.
- Base set may have up to 5000 nodes.
- How do you find the base set nodes?
  - Follow out-links by parsing root set pages.
  - Get in-links (and out-links) from a *connectivity server.*
  - (Actually, suffices to text-index strings of the form *href="URL"* to get in-links to *URL*.)

# Distilling hubs and authorities

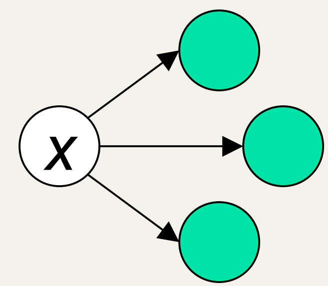
- Compute, for each page $x$ in the base set, a <u>hub score</u> $h(x)$ and an <u>authority score</u> $a(x)$.
- Initialize: for all $x$, $h(x) \leftarrow 1$; $a(x) \leftarrow 1$;
- Iteratively update all $h(x)$, $a(x)$;
- After iterations
  - output pages with highest $h()$ scores as top hubs
  - highest $a()$ scores as top authorities.

Key

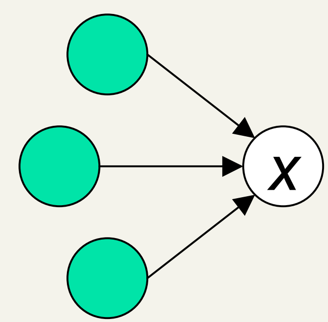
# Iterative update

- 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!
  - proof of this comes later.
- We only require the <u>relative orders</u> of the *h()* and *a()* scores - not their absolute values.
- In practice, ~5 iterations get you close to stability.

# Japan Elementary Schools

## Hubs

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

## Authorities

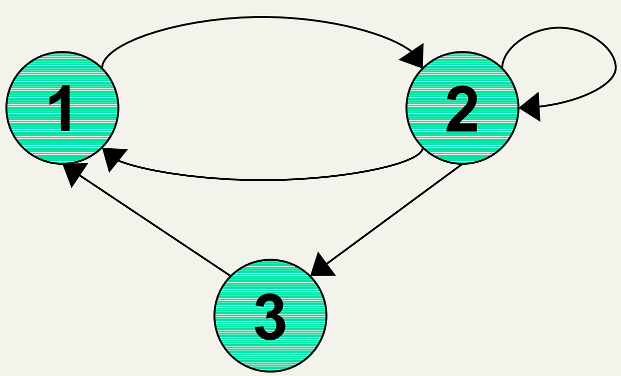
- The American School in Japan
- The Link Page
- ‰ª□è□s—§ˆä“c□¬Šw□Zƒz□[ƒ□ƒy□[ƒW
- Kids' Space
- ˆÀ□é□s—§ˆÀ□é□¼•”□¬Šw□Z
- ‹{□é‹³ˆç‘åŠw•□‘®□¬Šw□Z
- KEIMEI GAKUEN Home Page ( Japanese )
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- □_“Þ□ìŒ§□E‰¡•l□s—§’†□ì□¼□¬Šw□Z‚Ìƒy
- http://www...p/~m_maru/index.html
- 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 <u>after</u> base set assembled
  - iterative scoring is query-independent.
- Iterative computation <u>after</u> text index retrieval - significant overhead.

# 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

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

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

# Rewrite in matrix form

- $h=Aa$.
- $a=A^th$.

Recall $A^t$ is the transpose of A.

Substituting, $\mathbf{h}=\mathbf{AA^th}$ and $\mathbf{a}=\mathbf{A^tAa}$.

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

# Tag/position heuristics

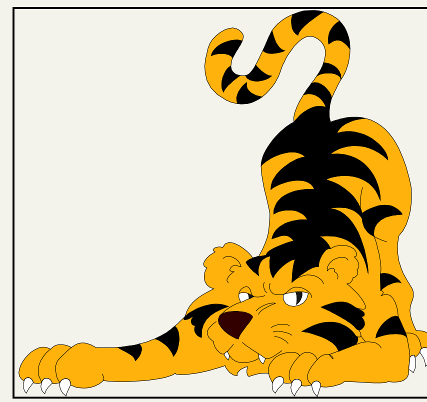
- Increase weights of terms
  - in titles
  - in tags
  - near the beginning of the doc, its chapters and sections

# Anchor text (first used *WWW Worm* - McBryan [Mcbr94])

Tiger image

Here is a great picture of a tiger

Cool tiger webpage



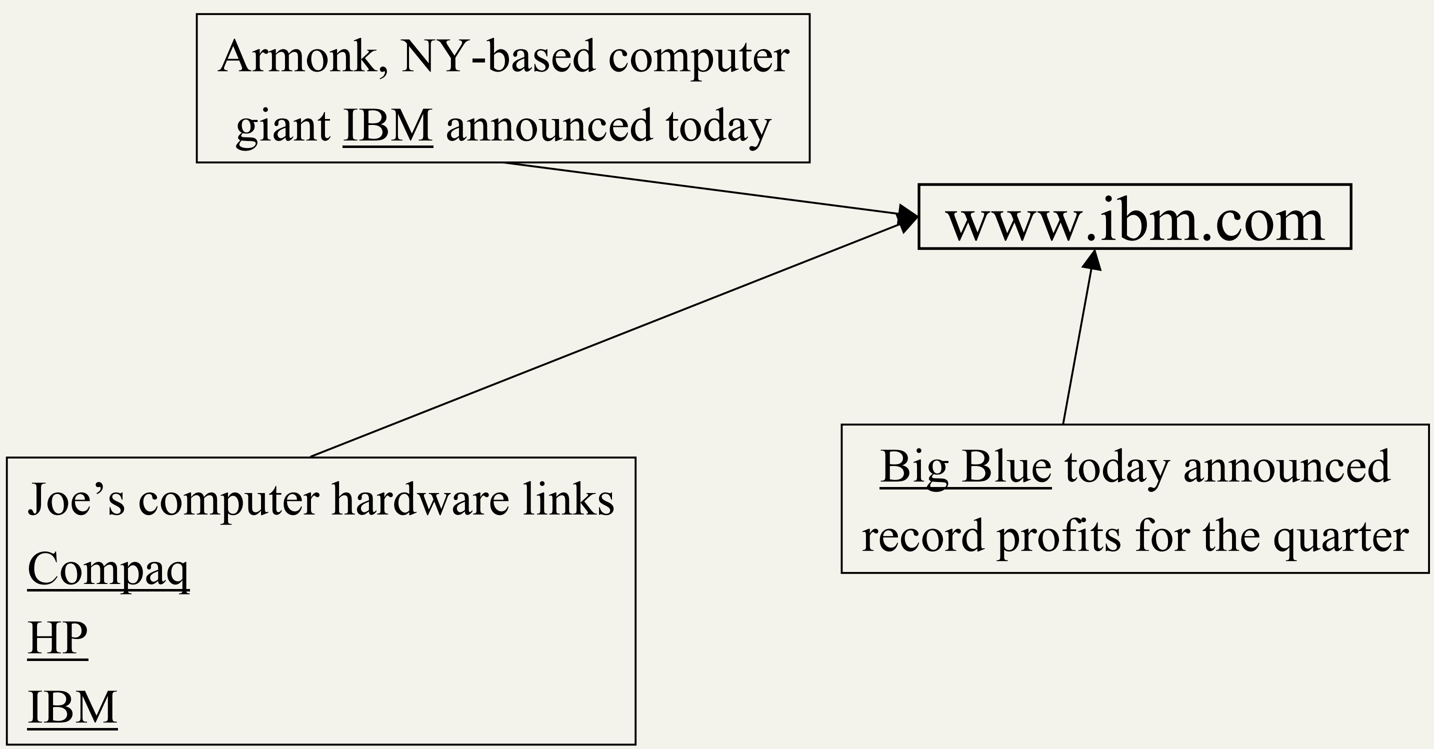
The text in the vicinity of a hyperlink is descriptive of the page it points to.

# Two uses of anchor text

- When indexing a page, also index the anchor text of links pointing to it.
  - Retrieve a page when query matches its anchor text.
- To weight links in the hubs/authorities algorithm.
- Anchor text usually taken to be a window of 6-8 words around a link anchor.

# Indexing anchor text

- When indexing a document *D*, include anchor text from links pointing to *D*.



# Indexing anchor text

- Can sometimes have unexpected side effects
  - *e.g., evil empire*.
- Can index anchor text with less weight.

# Weighting links

- In hub/authority link analysis, can match anchor text to query, then weight link.

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

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

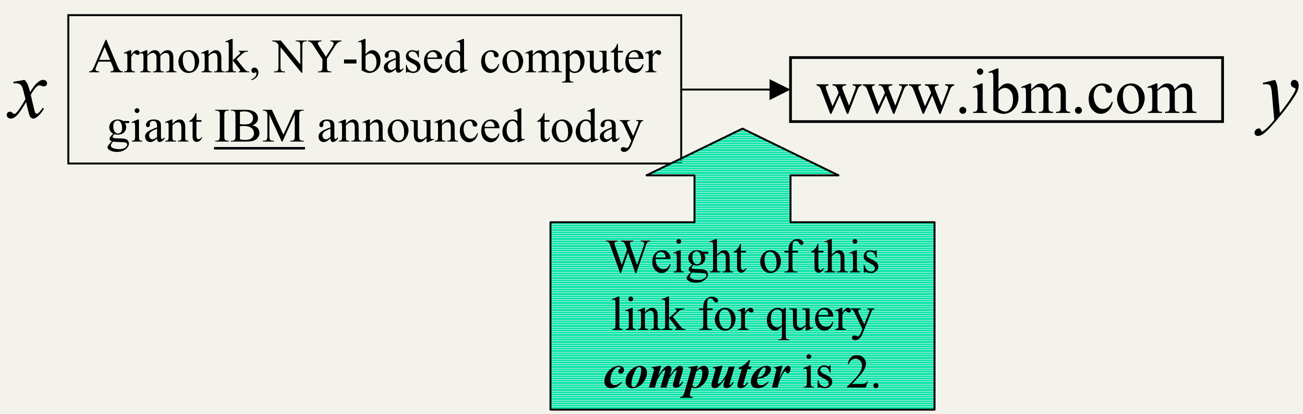
$$\Rightarrow$$

$$h(x) = \sum_{x \mapsto y} w(x,y) \cdot a(y)$$

$$a(x) = \sum_{y \mapsto x} w(x,y) \cdot h(y)$$

# Weighting links

- What is $w(x,y)$?
- Should increase with the number of query terms in anchor text.
  - E.g.: 1+ number of query terms.



# Weighted hub/authority computation

- Recall basic algorithm:
  - Iteratively update all *h(x), a(x)*;
  - After iteration, output pages with
    - highest *h()* scores as top hubs
    - highest *a()* scores as top authorities.
- Now use weights in iteration.
- Raises scores of pages with “heavy” links.

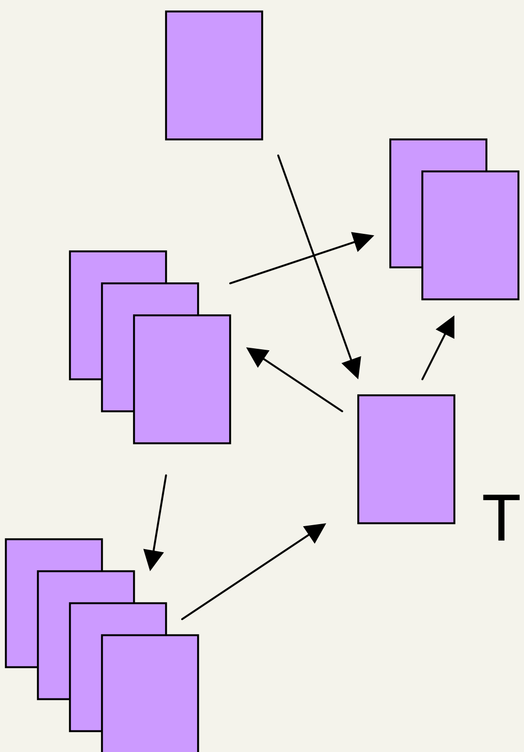
Do we still have convergence of scores? To what?

# Anchor Text

- Other applications
  - Weighting/filtering links in the graph
    - HITS [Chak98], Hilltop [Bhar01]
  - Generating page descriptions from anchor text [Amit98, Amit00]

# Web sites, not pages

- Lots of pages in a site give varying aspects of information on the same topic.



Treat portions of web-sites as a single entity for score computations.

# Link neighborhoods

- Links on a page tend to point to the same topics as neighboring links.
  - Break pages down into *pagelets* (say separate by tags)
    - compute a hub/authority score for each pagelet.

# Link neighborhoods - example

**Ron Fagin's links**

- Logic links
  - Moshe Vardi's logic page
  - International logic symposium
  - Paper on modal logic
- ….
- My favorite football team
  - The 49ers
  - Why the Raiders suck
  - Steve's homepage
  - The NFL homepage

# Comparison

## Pagerank

Pros

- Hard to spam
- Computes quality signal for <u>all</u> pages

Cons

- Non-trivial to compute
- Not query specific
- Doesn't work on small graphs

Proven to be effective for general purpose ranking

## HITS & Variants

Pros

- Easy to compute, real-time execution is hard [Bhar98b, Stat00]
- Query specific
- Works on small graphs

Cons

- Local graph structure can be manufactured (spam!)
- Provides a signal <u>only</u> when there's direct connectivity (e.g., home pages)

Well suited for supervised directory construction

# Topic Specific Pagerank [Have02]

- Conceptually, we use a random surfer who teleports, with say 10% probability, using the following rule:
  - Selects a category (say, one of the 16 top level ODP categories) based on a query & user -specific distribution over the categories
  - Teleport to a page uniformly at random within the chosen category
- Sounds hard to implement: can’t compute PageRank at query time!

# Topic Specific Pagerank [Have02]

- Implementation
  - offline:Compute pagerank distributions wrt to *individual* categories

    Query independent model as before

    Each page has multiple pagerank scores – one for each ODP category, with teleportation only to that category
  - online: Distribution of weights over categories computed by query context classification

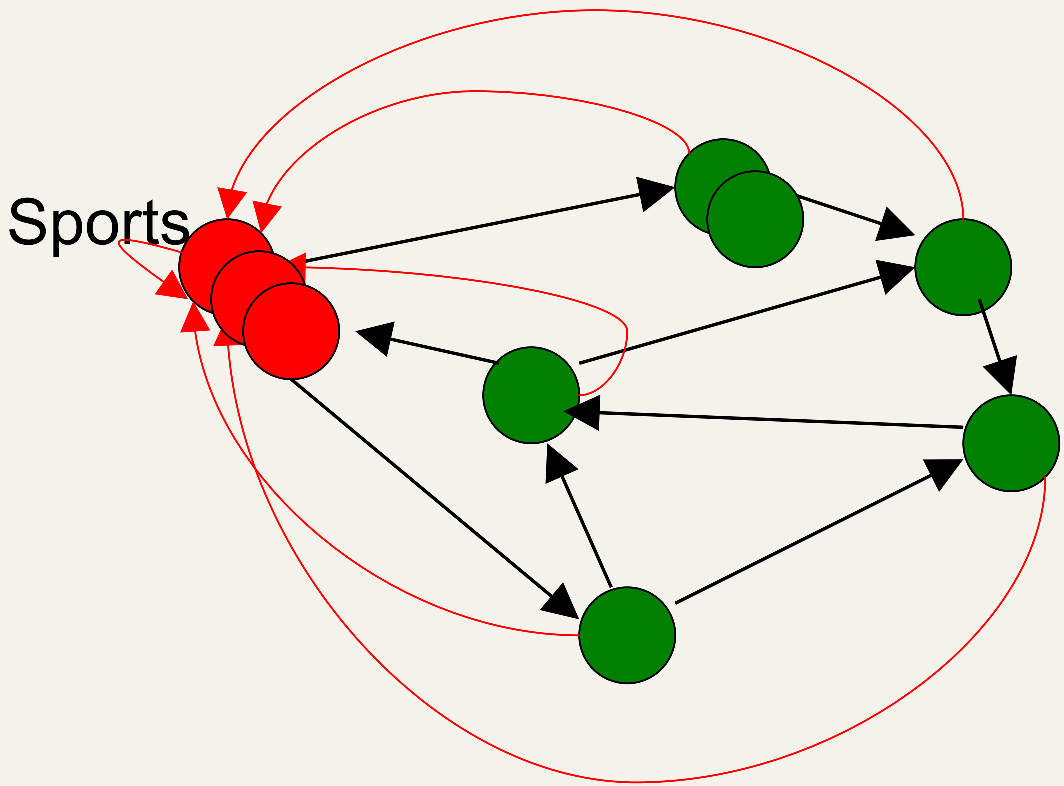
    Generate a dynamic pagerank score for each page - weighted sum of category-specific pageranks

# Influencing PageRank ("Personalization")

- Input:
  - Web graph $W$
  - influence vector v

    v : (page $\rightarrow$ degree of influence)
- Output:
  - Rank vector r: (page $\rightarrow$ page importance wrt v )
- r = PR($W$, v)

# Non-uniform Teleportation




Teleport with 10% probability to a Sports page

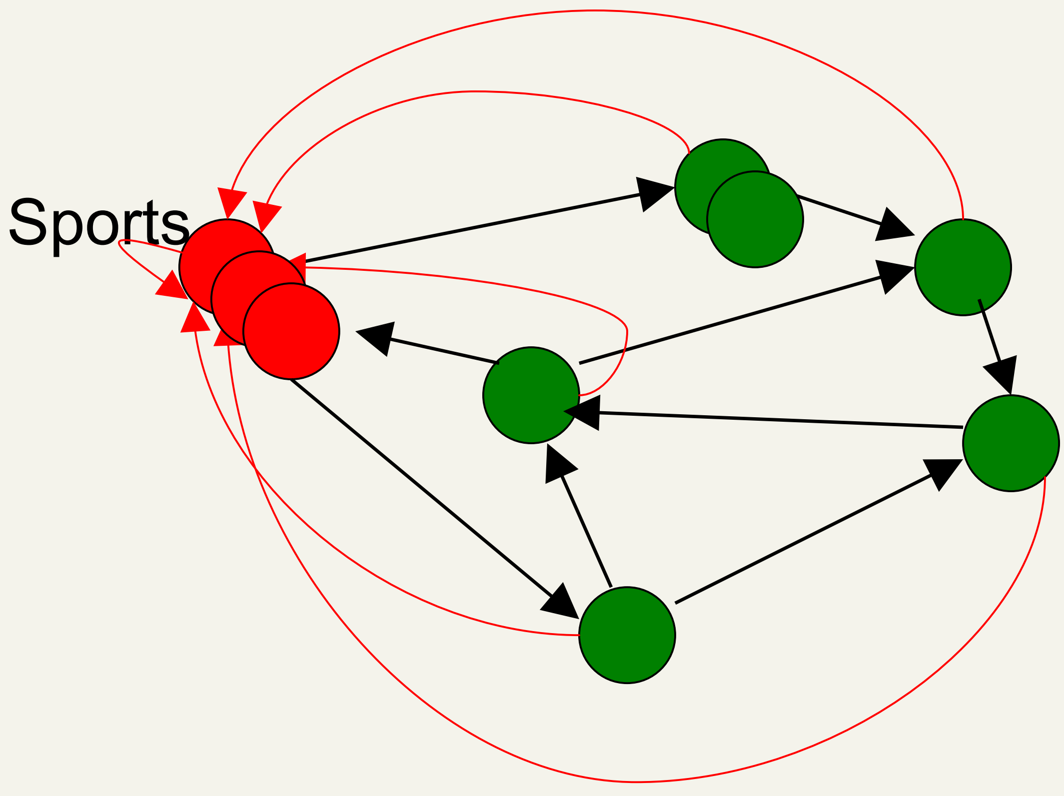
# Interpretation of Composite Score

- For a set of personalization vectors $\{v_j\}$

$$\sum_j [w_j \cdot PR(W, v_j)] = PR(W, \sum_j [w_j \cdot v_j])$$

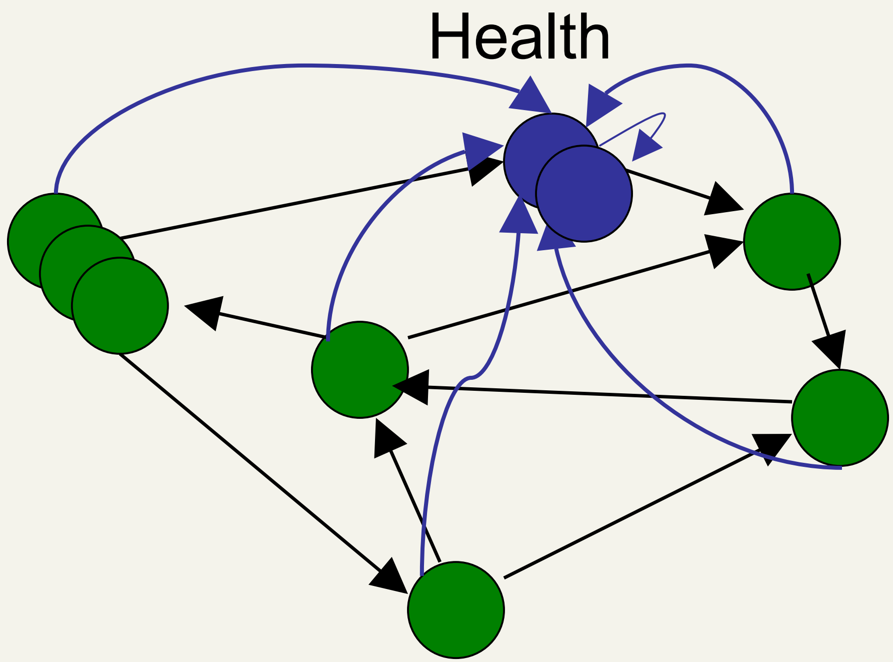
- Weighted sum of rank vectors itself forms a valid rank vector, because PR() is linear wrt $v_j$

# Interpretation



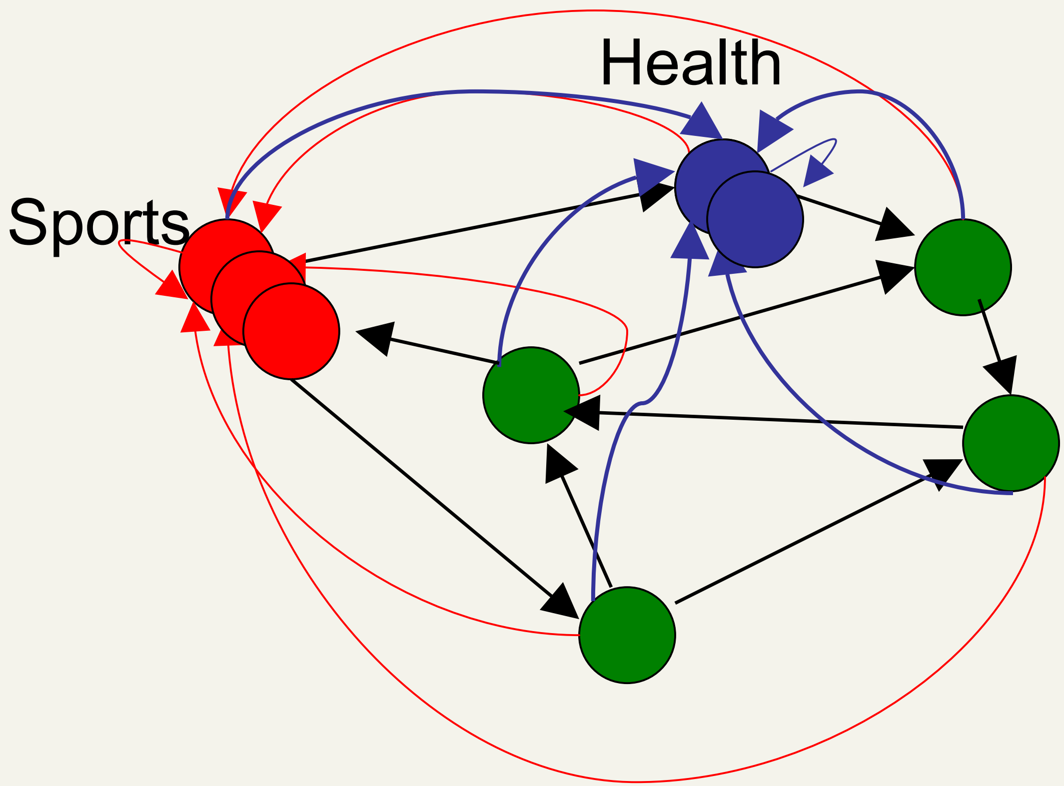

10% Sports teleportation

# Interpretation




10% Health teleportation

# Interpretation




pr = (0.9 $PR_{sports}$ + 0.1 $PR_{health}$) gives you:
9% sports teleportation, 1% health teleportation

# Web vs. hypertext search

- The WWW is full of free-spirited opinion, annotation, authority conferral
- Most other forms of hypertext are far more structured
  - enterprise intranets are regimented and templated
  - very little free-form community formation
  - web-derived link ranking doesn’t quite work

# Next up

- Behavior-based ranking
- Crawling
- Spam detection
- Mirror detection
- Web search infrastructure