Course Name: Web Information Retrieval

PageRank

Qingcai Chen

ICRC HIT (Shenzhen)

Sept.30, 2019

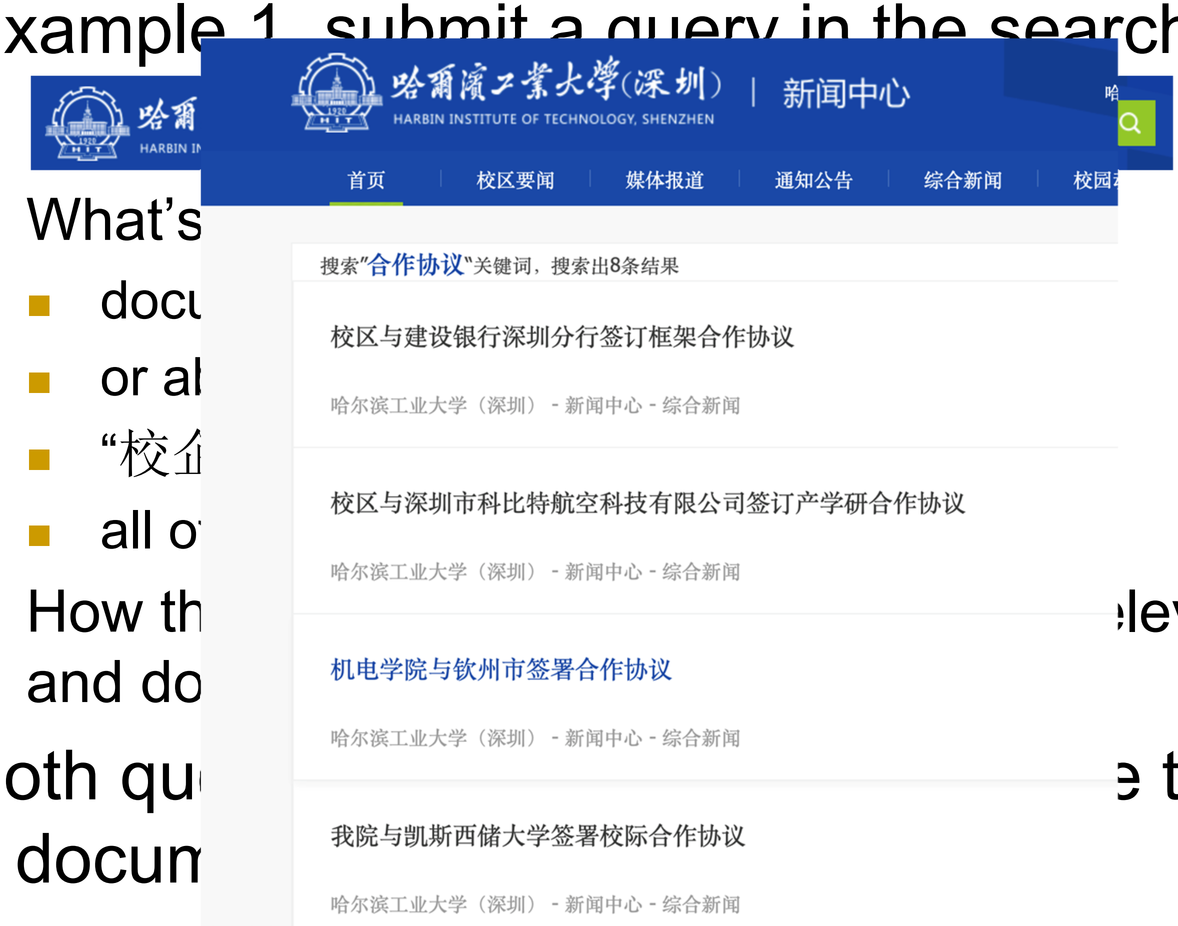
Content

- Background
- Basic Idea
- PageRank Algorithm
- HITS: Hyperlink-Induced Topic Search
- Further Reading

HITS: Hyperlink-Induced Topic Search,
referred from: Pandu Nayak and Prabhakar Raghavan, Lecture 17: Link Analysis

Background

- Example 1. submit a query in the search box


哈爾濱工業大學(深圳) | 新闻中心
HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN
哈爾
HARBIN IN
首页
校区要闻
媒体报道
通知公告
综合新闻
校园

搜索"合作协议"关键词，搜索出8条结果

校区与建设银行深圳分行签订框架合作协议

哈尔滨工业大学（深圳） - 新闻中心 - 综合新闻

校区与深圳市科比特航空科技有限公司签订产学研合作协议

哈尔滨工业大学（深圳） - 新闻中心 - 综合新闻

机电学院与钦州市签署合作协议

哈尔滨工业大学（深圳） - 新闻中心 - 综合新闻

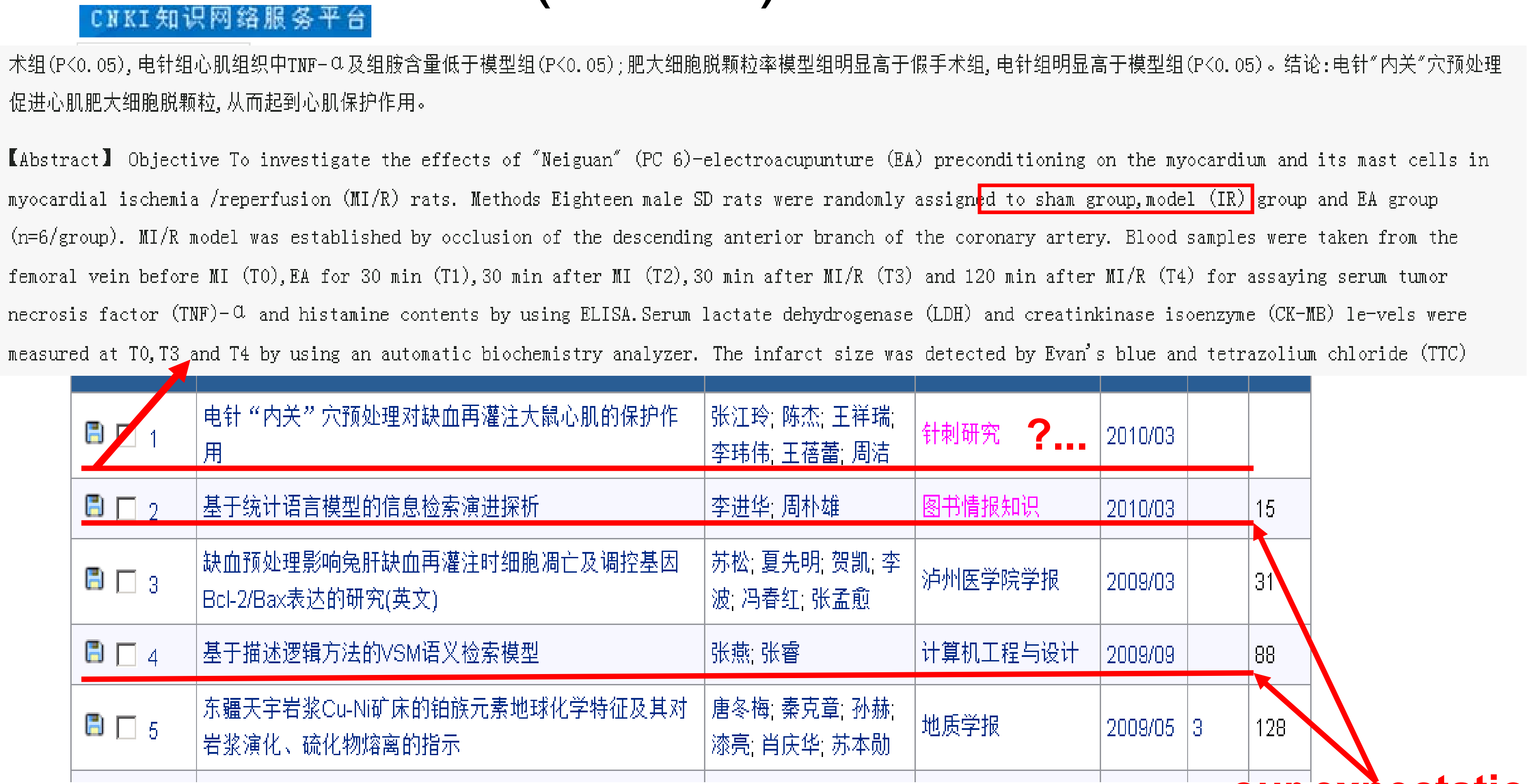
我院与凯斯西储大学签署校际合作协议

哈尔滨工业大学（深圳） - 新闻中心 - 综合新闻

  - What's
    - docu
    - or ab
    - "校企
    - all o
  - How th elevance of query and do
- Both qu e there are just 5 docum

Background

- Example 2. a more complicate query, search in a moderate (million) scale document set



术组($P<0.05$),电针组心肌组织中TNF-α及组胺含量低于模型组($P<0.05$);肥大细胞脱颗粒率模型组明显高于假手术组,电针组明显高于模型组($P<0.05$)。结论:电针"内关"穴预处理促进心肌肥大细胞脱颗粒,从而起到心肌保护作用。

【Abstract】 Objective To investigate the effects of "Neiguan" (PC 6)-electroacupunture (EA) preconditioning on the myocardium and its mast cells in myocardial ischemia /reperfusion (MI/R) rats. Methods Eighteen male SD rats were randomly assigned to sham group,model (IR) group and EA group (n=6/group). MI/R model was established by occlusion of the descending anterior branch of the coronary artery. Blood samples were taken from the femoral vein before MI (T0),EA for 30 min (T1),30 min after MI (T2),30 min after MI/R (T3) and 120 min after MI/R (T4) for assaying serum tumor necrosis factor (TNF)-α and histamine contents by using ELISA.Serum lactate dehydrogenase (LDH) and creatinkinase isoenzyme (CK-MB) le-vels were measured at T0,T3 and T4 by using an automatic biochemistry analyzer. The infarct size was detected by Evan's blue and tetrazolium chloride (TTC)

□ 1	电针"内关"穴预处理对缺血再灌注大鼠心肌的保护作用	张江玲; 陈杰; 王祥瑞; 李玮伟; 王蓓蕾; 周洁	针刺研究 ?...	2010/03		
□ 2	基于统计语言模型的信息检索演进探析	李进华; 周朴雄	图书情报知识	2010/03		15
□ 3	缺血预处理影响兔肝缺血再灌注时细胞凋亡及调控基因Bcl-2/Bax表达的研究(英文)	苏松; 夏先明; 贺凯; 李波; 冯春红; 张孟愈	泸州医学院学报	2009/03		31
□ 4	基于描述逻辑方法的VSM语义检索模型	张燕; 张睿	计算机工程与设计	2009/09		88
□ 5	东疆天宇岩浆Cu-Ni矿床的铂族元素地球化学特征及其对岩浆演化、硫化物熔离的指示	唐冬梅; 秦克章; 孙赫; 漆亮; 肖庆华; 苏本勋	地质学报	2009/05	3	128

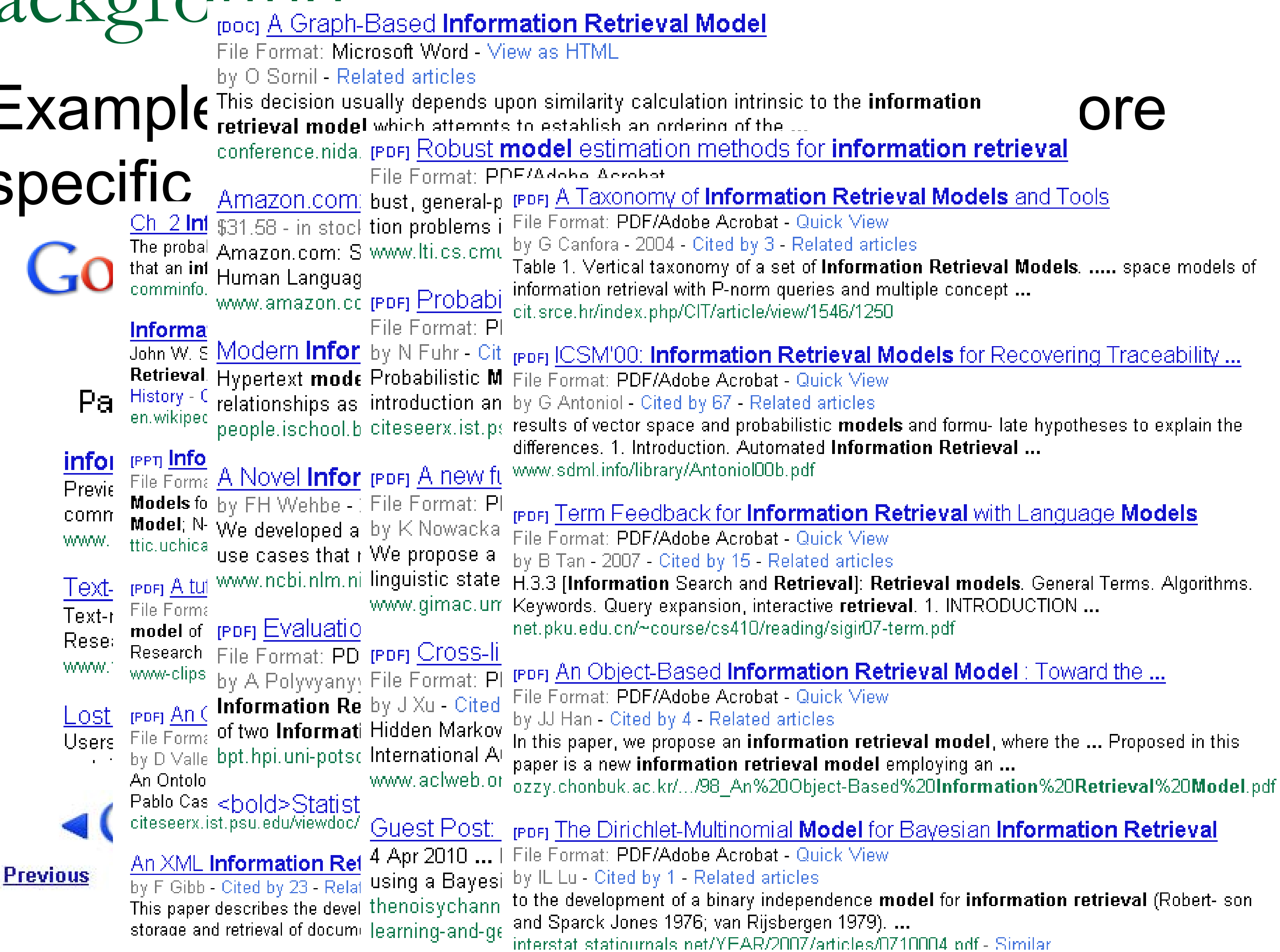
our expectation

Background

- Example 2. a more complicate query, search in a moderate (million) scale document set
  - Information retrieval models are required to find out relevant documents
    - Vector space model
    - Statistical language model
    - Probabilistic model
    - ……
  - Goal of IR models: increase search precision by
    - Understanding user’s information needs
    - Providing techniques for computing relevance (or similarity) of query and documents

Background

- Example ... ore specific



[DOC] A Graph-Based **Information Retrieval Model**
File Format: Microsoft Word - View as HTML
by O Sornil - Related articles
This decision usually depends upon similarity calculation intrinsic to the **information retrieval model** which attempts to establish an ordering of the ...
conference.nida.

[PDF] Robust **model** estimation methods for **information retrieval**
File Format: PDF/Adobe Acrobat
bust, general-p
tion problems i
www.lti.cs.cmu

[PDF] A Taxonomy of **Information Retrieval Models** and Tools
File Format: PDF/Adobe Acrobat - Quick View
by G Canfora - 2004 - Cited by 3 - Related articles
Table 1. Vertical taxonomy of a set of **Information Retrieval Models**. ..... space models of information retrieval with P-norm queries and multiple concept ...
cit.srce.hr/index.php/CIT/article/view/1546/1250

[PDF] ICSM'00: **Information Retrieval Models** for Recovering Traceability ...
File Format: PDF/Adobe Acrobat - Quick View
by G Antoniol - Cited by 67 - Related articles
results of vector space and probabilistic **models** and formu- late hypotheses to explain the differences. 1. Introduction. Automated **Information Retrieval** ...
www.sdml.info/library/Antoniol00b.pdf

[PDF] Term Feedback for **Information Retrieval** with Language **Models**
File Format: PDF/Adobe Acrobat - Quick View
by B Tan - 2007 - Cited by 15 - Related articles
H.3.3 [**Information** Search and **Retrieval**]: **Retrieval models**. General Terms. Algorithms. Keywords. Query expansion, interactive **retrieval**. 1. INTRODUCTION ...
net.pku.edu.cn/~course/cs410/reading/sigir07-term.pdf

[PDF] An Object-Based **Information Retrieval Model** : Toward the ...
File Format: PDF/Adobe Acrobat - Quick View
by JJ Han - Cited by 4 - Related articles
In this paper, we propose an **information retrieval model**, where the ... Proposed in this paper is a new **information retrieval model** employing an ...
ozzy.chonbuk.ac.kr/.../98_An%20Object-Based%20**Information**%20**Retrieval**%20**Model**.pdf

[PDF] The Dirichlet-Multinomial **Model** for Bayesian **Information Retrieval**
File Format: PDF/Adobe Acrobat - Quick View
by IL Lu - Cited by 1 - Related articles
to the development of a binary independence **model** for **information retrieval** (Robert- son and Sparck Jones 1976; van Rijsbergen 1979). ...
interstat.statjournals.net/YEAR/2007/articles/0710004.pdf - Similar

Amazon.com:
$31.58 - in stocl
Amazon.com: S
Human Languag
www.amazon.cc

Modern **Infor**
Hypertext **mode**
relationships as
people.ischool.b

A Novel **Infor**
by FH Wehbe -
We developed a
use cases that
www.ncbi.nlm.ni

[PDF] Evaluatio
File Format: PD
by A Polyvyany
Information Re
of two **Informati**
bpt.hpi.uni-potsc

<bold>Statist

[PDF] Probabi
File Format: Pl
by N Fuhr - Cit
Probabilistic **M**
introduction an
citeseerx.ist.ps

[PDF] A new fu
File Format: Pl
by K Nowacka
We propose a
linguistic state
www.gimac.ur

[PDF] Cross-li
File Format: Pl
by J Xu - Cited
Hidden Markov
International A
www.aclweb.o

Guest Post:
4 Apr 2010 ...
using a Bayesi
thenoisychann
learning-and-ge

Ch 2 **Inf**
The probal
that an **in**
comminfo.

Informa
John W. S
Retrieval
History - C
en.wikiped

[PPT] **Info**
File Forma
Models fo
Model; N-
ttic.uchica

[PDF] A tu
File Forma
model of
Research
www-clips

[PDF] An (
File Forma
by D Valle
An Ontolo
Pablo Cas
citeseerx.ist.psu.edu/viewdoc/

An XML **Information Ret**
by F Gibb - Cited by 23 - Rela
This paper describes the devel
storage and retrieval of docum

Go

Pa

infor
Previe
comm
www.

Text-
Text-r
Rese
www.

Lost
Users

Previous

Background

- Example 3. do the same search (with more specific keywords of Example 2) via Google
  - Now the critical problem becomes:
    - Which kind of documents should be presented on the 1st page?
  - It's not a question about relevant, rather than, it's a question about
    - Which document is more important?

Background

Document measures

Relevance, as conventionally defined, is binary (relevant or not relevant). It is usually estimated by the similarity between the terms in the query and each document: Boolean Model, Vector Space Model, Statistical Language Model, etc.

Importance measures documents by their likelihood of being useful to a variety of users. It is usually estimated by some measure of popularity.

Web search engines rank documents
by a combination of relevance and importance. The goal is to present the user with the most important of the relevant documents.

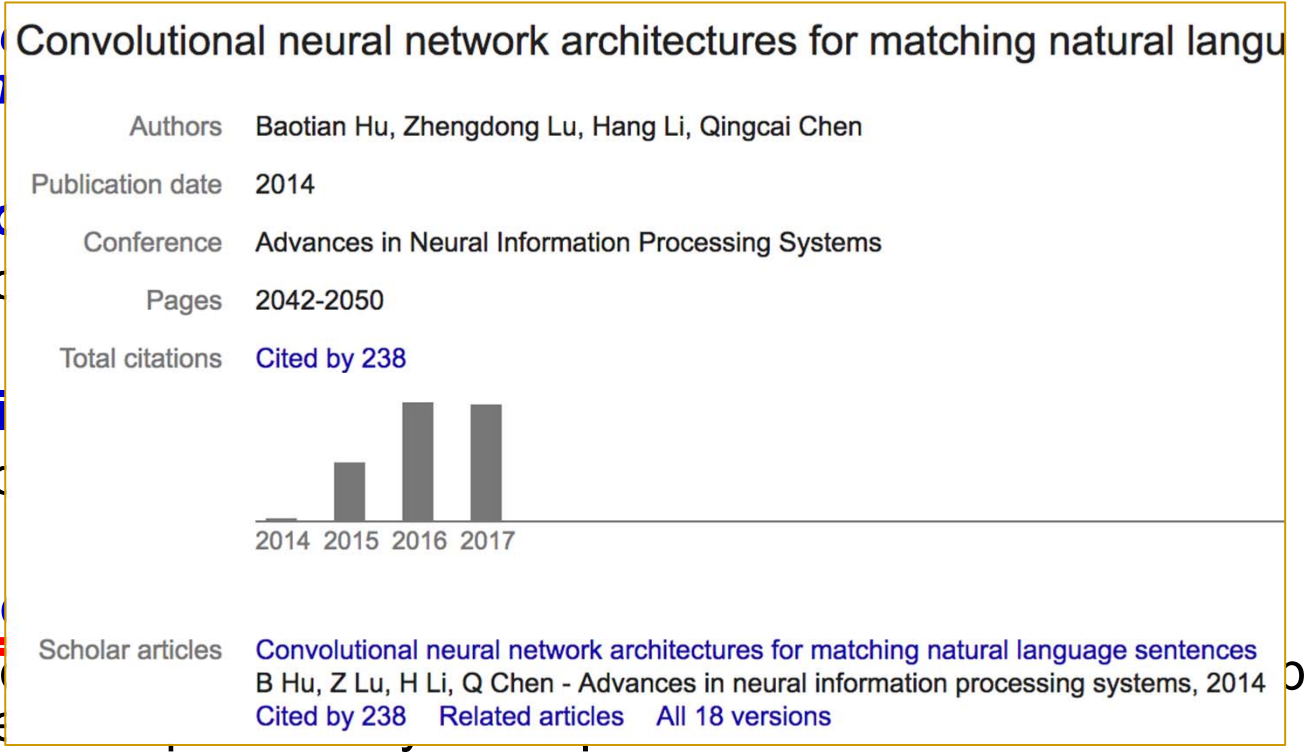
Then, how to compute the importance of a web document?

Basic Idea

- The question about “importance” also comes from the domain “Bibliometrics (文献计量学)”

Basic Idea

- **Bibliometrics (文献计量学):**
  - ***Te***
    sin
- **Biblio**
  - two
- **Co-ci**
  - two
- **Impa**
  - freq been
    cite
  - measure of “importance” of a journal



Convolutional neural network architectures for matching natural langu

Authors	Baotian Hu, Zhengdong Lu, Hang Li, Qingcai Chen
Publication date	2014
Conference	Advances in Neural Information Processing Systems
Pages	2042-2050
Total citations	Cited by 238

Scholar articles Convolutional neural network architectures for matching natural language sentences
B Hu, Z Lu, H Li, Q Chen - Advances in neural information processing systems, 2014
Cited by 238 Related articles All 18 versions

Basic Idea

Basic philosophy implied in Impact Factor:

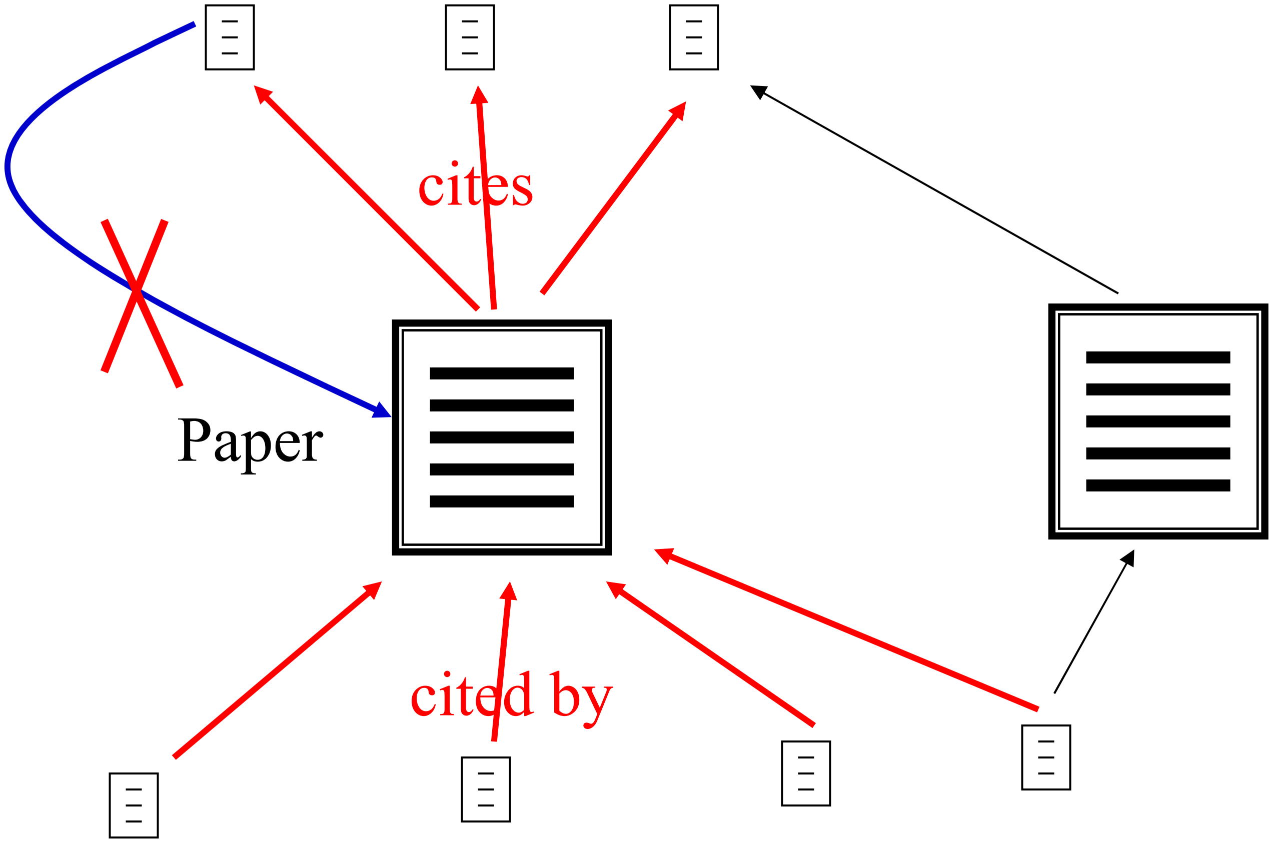
The importance of a journal or a paper is enhanced if more papers cite it.

Just those citations that come from important journals are counted.

Citation Graph - Visualization of Citations

Note that journal citations always refer to earlier work.

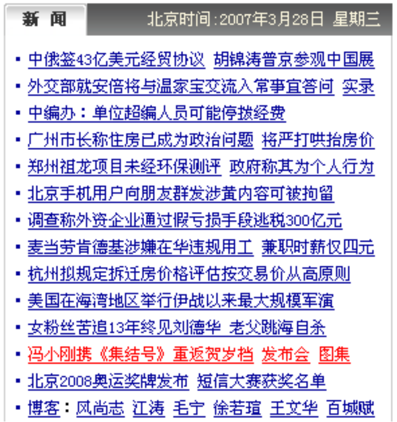
Directed Acyclic Graph(有向无环图, DAG)



Graphical Analysis of Hyperlinks on the Web

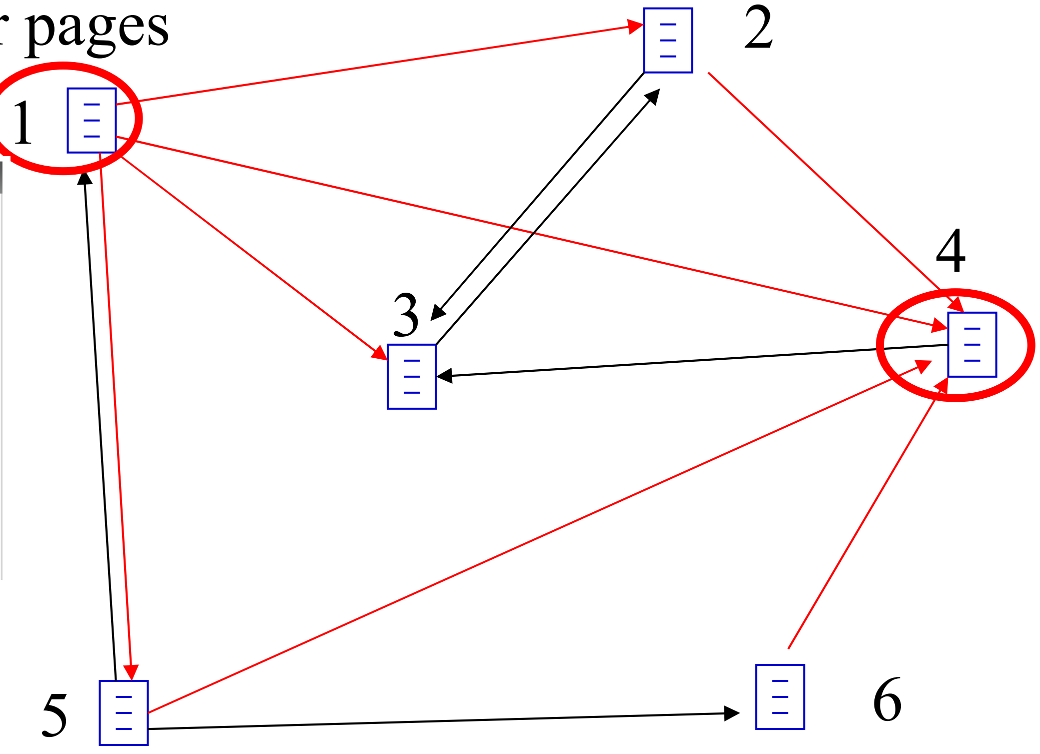
What's the difference between Hyperlink graph and Citation graph?

This page links to many other pages (hub)



新 闻　北京时间：2007年3月28日 星期三

- 中俄签43亿美元经贸协议 胡锦涛普京参观中国展
- 外交部就安倍将与温家宝交流入常事宜答问 实录
- 中编办：单位超编人员可能停拨经费
- 广州市长称住房已成为政治问题 将严打哄抬房价
- 郑州祖龙项目未经环保测评 政府称其为个人行为
- 北京手机用户向朋友群发涉黄内容可被拘留
- 调查称外资企业通过假亏损手段逃税300亿元
- 麦当劳肯德基涉嫌在华违规用工 兼职时薪仅四元
- 杭州拟规定拆迁房价格评估按交易价从高原则
- 美国在海湾地区举行伊战以来最大规模军演
- 女粉丝苦追13年终见刘德华 老父跳海自杀
- 冯小刚携《集结号》重返贺岁档 发布会 图集
- 北京2008奥运奖牌发布 短信大赛获奖名单
- 博客：风尚志 江涛 毛宁 徐若瑄 王文华 百城赋



Many pages link to this page (authority)

Matrix Representation

PageRank Algorithm

Used to estimate importance of documents.

Inspired by the Impact Factor.

Basic Concept:

1. The rank of a web page is higher if many pages link to it.

2. Links from highly ranked pages are given greater weight than links from less highly ranked pages.

Question:

How do we know the rank of a web page that is linking to the web page under consideration?

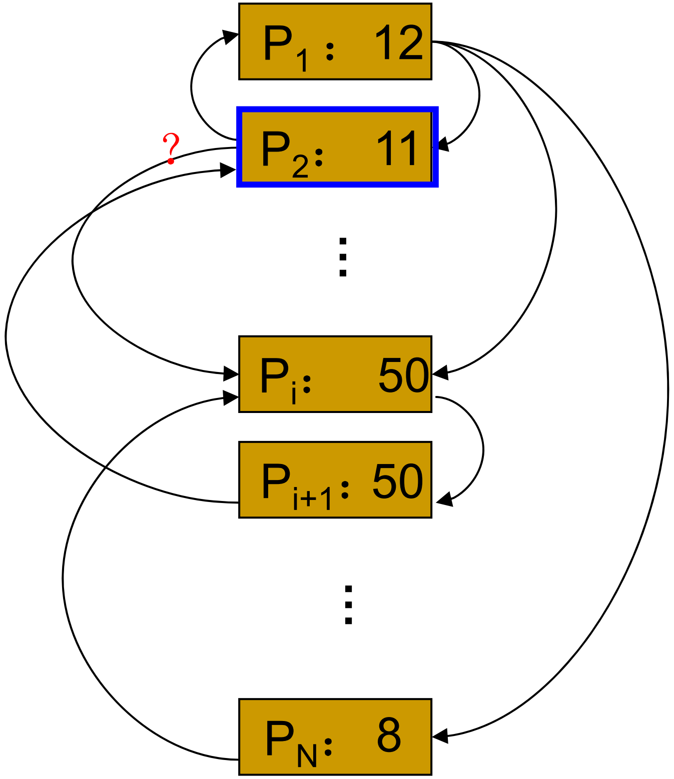
Intuitive Model (no damping)

A user:

1. Starts at a random page on the web
2. Selects a random hyperlink from the current page and jumps to the corresponding page
3. Repeats Step 2 a very large number of times

Pages are ranked according to the relative frequency with which they are visited.

After *T* times (and *T* is large enough), we get



Then we can compute the rank (importance) of each webpage according to its visiting times

Intuitive Model (Probabilistic Model)

- It’s a stochastic process
- To get a precise estimation of visiting times is not practicable
- Estimate the visiting probability of one webpage by the visiting probability of webpages Link In to it, for the example, we get:

$$\Pr(P_i) = \Pr(P_i \mid P_1)\Pr(P_1) + \Pr(P_i \mid P_2)\Pr(P_2) + \Pr(P_i \mid P_N)\Pr(P_N)$$

,

- We just want to get a relative importance (PageRank value), use *w* to replace the probability symbol” Pr”, and get the matrix form:

$$w_i = \mathbf{Pr}_i \bullet \mathbf{w} \quad \text{or} \quad \mathbf{w} = \mathbf{B} \bullet \mathbf{w} \quad \text{for} \quad \mathbf{B} = [\mathbf{Pr}_1, \cdots, \mathbf{Pr}_N]^T$$

$$\mathbf{Pr}_i = [\Pr(P_i \mid P_1), \Pr(P_i \mid P_2), \Pr(P_i \mid P_3), \cdots, \Pr(P_i \mid P_N)]$$

$$\mathbf{w} = [w_1, w_2, w_3, \cdots, w_N]^T \quad w_i \rightarrow \Pr(P_i)$$

$$\Pr(P_i \mid P_j) = \begin{cases} 0, & \text{If there is no hyperlink from } P_j \text{ to } P_i \\ 1/C(P_j), & \text{otherwise} \end{cases}$$

$C(P_j)$ **is the Link Out of webpage *P_j***

Matrix Representation

The Citation Graph

Citing page (from)

		P_1	P_2	P_3	P_4	P_5	P_6	Number
	P_1					1		1
	P_2	1		1				2
Cited page (to)	P_3	1	1		1			3
	P_4	1	1			1	1	4
	P_5	1						1
	P_6					1		1
Number		4	2	1	1	3	1	

P_2 cites P_3

Basic Algorithm: Normalize by Number of Links from Page

Citing page

Cited page	P1	P2	P3	P4	P5	P6
P1					0.33	
P2	0.25		1			
P3	0.25	0.5		1		
P4	0.25	0.5			0.33	1
P5	0.25					
P6					0.33	
Number	4	2	1	1	3	1

= **B**

Normalized link matrix

Basic Algorithm: Weighting of Pages

Initially all pages have weight 1

$$\mathbf{w}_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Recalculate weights

$$\mathbf{w}_1 = \mathbf{B}\mathbf{w}_0 = \begin{bmatrix} 0.33 \\ 1.25 \\ 1.75 \\ 2.08 \\ 0.25 \\ 0.33 \end{bmatrix}$$

Here the first iterating weight for P1 is

$\boldsymbol{w_{11}} = \mathbf{B_1} \bullet \mathbf{w_0}$

$= \mathbf{b_{11} * w_{01} + \ldots + b_{15} * w_{05} + \ldots}$

$= \mathbf{0 * 1 + \ldots + 0.33 * 1 + \ldots}$

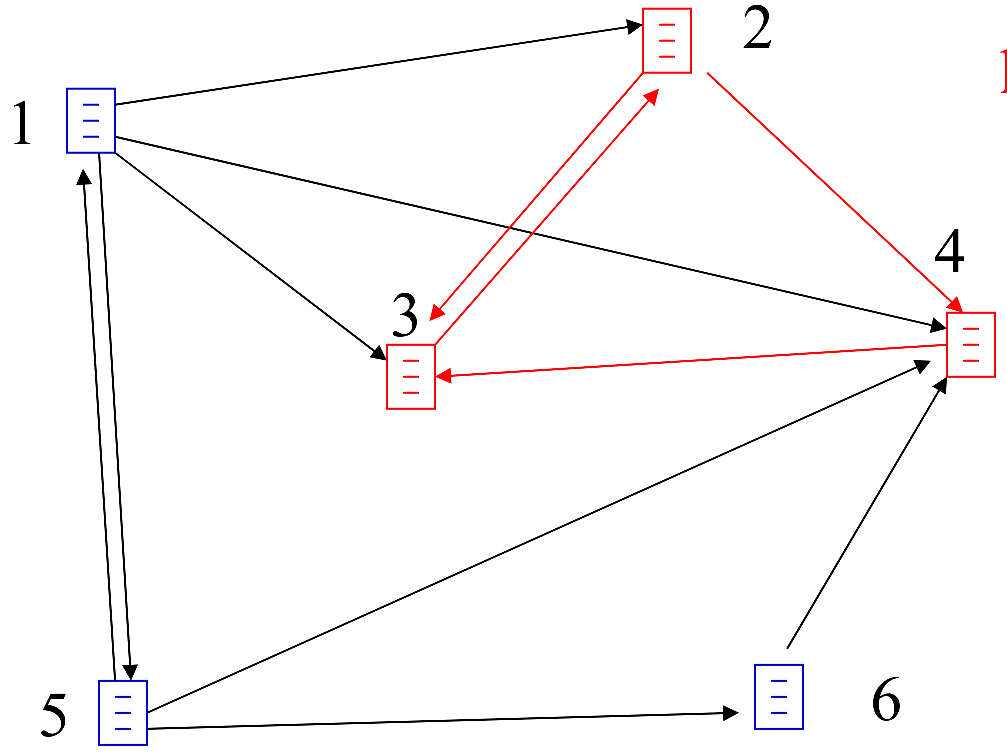
$\mathbf{=0.33}$

Basic Algorithm: Iterate

Iterate: $\mathbf{w}_k = \mathbf{B}\mathbf{w}_{k\text{-}1}$

$$\begin{array}{cccccc}
\mathbf{w}_0 & \mathbf{w}_1 & \mathbf{w}_2 & \mathbf{w}_3 & \text{... converges to ...} & \mathbf{w} \\
\begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} &
\begin{bmatrix}0.33\\1.25\\1.75\\2.08\\0.25\\0.33\end{bmatrix} &
\begin{bmatrix}0.08\\1.83\\2.79\\1.12\\0.08\\0.08\end{bmatrix} &
\begin{bmatrix}0.03\\2.80\\2.06\\1.05\\0.02\\0.03\end{bmatrix} &
\begin{matrix}\text{->}\\\text{->}\\\text{->}\\\text{->}\\\text{->}\\\text{->}\end{matrix} &
\begin{bmatrix}0.00\\2.39\\2.39\\1.19\\0.00\\0.00\end{bmatrix}
\end{array}$$

Graphical Analysis of Hyperlinks on the Web



There is no link out of {2, 3, 4}

Discussion:

How to deal with the issue of link loops?

Google PageRank with Damping

A user:

1. Starts at a random page on the web

2a. With probability *d*, selects any random page and jumps to it

2b. With probability 1-*d*, selects a random hyperlink from the current page and jumps to the corresponding page

3. Repeats Step 2a and 2b a very large number of times

Pages are ranked according to the relative frequency with which they are visited.

Now, can we construct this new model?

The PageRank Iteration

The **basic method** iterates using the **normalized link matrix, B.**

$$\mathbf{w}_k = \mathbf{B}\mathbf{w}_{k\text{-}1}$$

This **w** is the high order eigenvector of **B (i.e., w = Bw)**

PageRank iterates using a damping factor. The method iterates:

$$\mathbf{w}_k = d\mathbf{w}_0 + (1 - d)\mathbf{B}\mathbf{w}_{k\text{-}1}$$

$\mathbf{w}_0$ is a vector with every element equal to 1.
d is a constant found by experiment.

Iterate with Damping

Iterate: $\mathbf{w}_k = d\mathbf{w}_0 + (1\text{-}d)\mathbf{B}\mathbf{w}_{k\text{-}1}$ $(d = 0.3)$

$$\begin{array}{cccccc}
\mathbf{w}_0 & \mathbf{w}_1 & \mathbf{w}_2 & \mathbf{w}_3 & \text{... converges to ...} & \mathbf{w} \\
\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} &
\begin{bmatrix} 0.53 \\ 1.18 \\ 1.53 \\ 1.76 \\ 0.48 \\ 0.53 \end{bmatrix} &
\begin{bmatrix} 0.41 \\ 1.46 \\ 2.03 \\ 1.29 \\ 0.39 \\ 0.41 \end{bmatrix} &
\begin{bmatrix} 0.39 \\ 1.80 \\ 1.78 \\ 1.26 \\ 0.37 \\ 0.39 \end{bmatrix} &
\begin{matrix} \text{->} \\ \text{->} \\ \text{->} \\ \text{->} \\ \text{->} \\ \text{->} \end{matrix} &
\begin{bmatrix} 0.38 \\ 1.68 \\ 1.87 \\ 1.31 \\ 0.37 \\ 0.38 \end{bmatrix}
\end{array}$$

Google: PageRank

The Google PageRank algorithm is usually written with the following notation

If page *A* has pages T_i pointing to it.

- d: damping factor
- $C(T_i)$: number of links out of T_i
- *n:* number of pages pointing to *A*

Iterate until:

$$P(A) = (1 - d) + d\left(\sum_{i=1}^{n} \frac{P(T_i)}{C(T_i)}\right)$$

Information Retrieval Using PageRank

Simple Method

Consider all hits (i.e., all document vectors that share at least one term with the query vector) as equal.

Display the hits ranked by PageRank.

The disadvantage of this method is that it gives no attention to how closely a document matches a query

(i.e. No Relevance)

Combining Term Weighting with Reference Pattern Ranking

Combined Method

1. Find all documents that share a term with the query vector.

2. The similarity, using conventional term weighting, between the **query and document** j is s_j.

3. The rank of **document** j using PageRank or other reference pattern ranking is p_j.

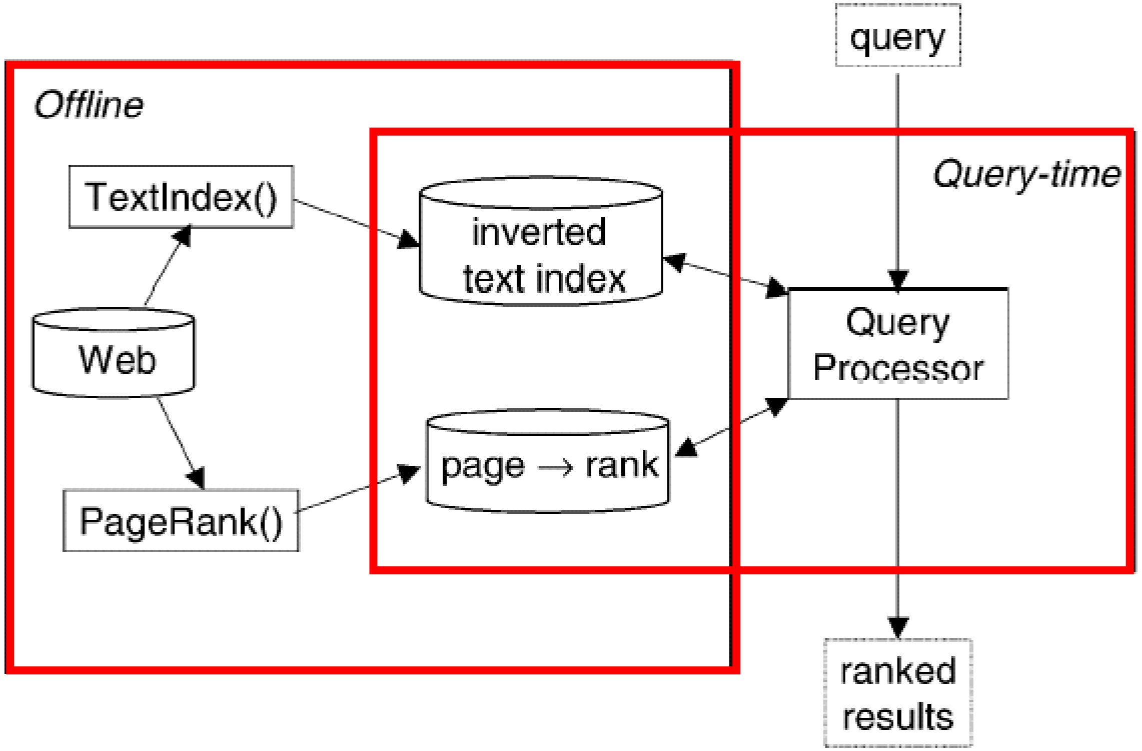
4. Calculate a combined rank $c_j = \lambda s_j + (1- \lambda)p_j$, where λ is a constant.

5. Display the hits ranked by c_j.

This method is used in several commercial systems but the details have not been published.

A standard search engine scheme by using PR

Haveliwala: Topic-sensitive Pagerank: A Context-sensitive Ranking Algorithm for Web Search



Topic sensitive PageRank

Topic sensitive PageRank

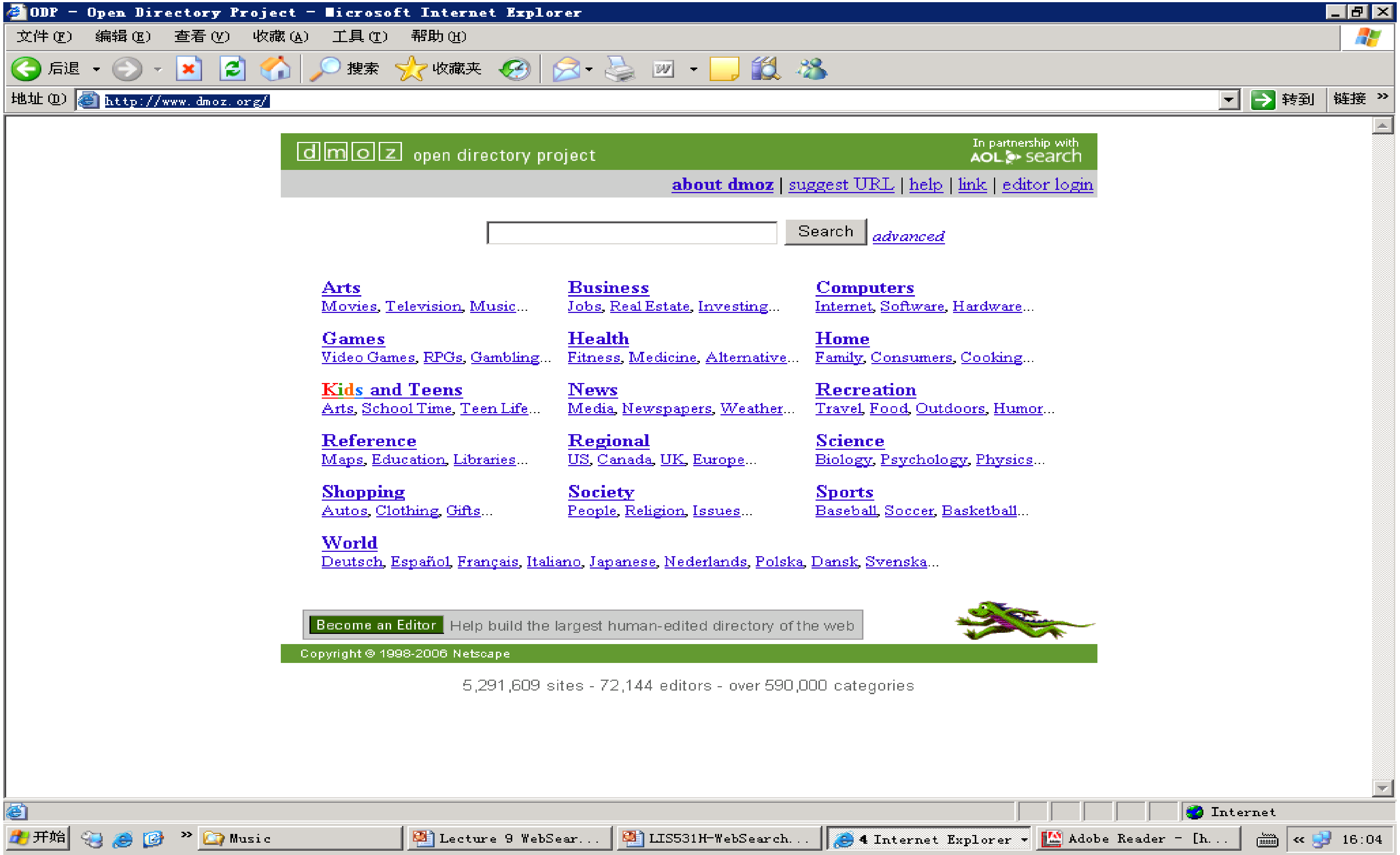
- Basic Idea:
  - Once users random select one page to jump to, they may jump just among the webpages that belong to the same subject (or topic).

Topic Sensitive PageRank

Haveliwala: Topic-sensitive Pagerank: A Context-sensitive Ranking Algorithm for Web Search, 2003

- Assigns *multiple* a-priori “importance” estimates to pages
- One PageRank score per *basis topic*
  - Query specific rank score
  - Make use of context
  - Inexpensive at runtime
- Basis topics come from Yahoo, ODP (Open Directory Project http://www.dmoz.org/) etc.

Open Directory Project



ODP – Open Directory Project – Microsoft Internet Explorer
文件 (F) 编辑 (E) 查看 (V) 收藏 (A) 工具 (T) 帮助 (H)
后退 搜索 收藏夹
地址 (D) http://www.dmoz.org/
转到 链接

dmoz open directory project

In partnership with AOL search

about dmoz | suggest URL | help | link | editor login

Search advanced

Arts
Movies, Television, Music...

Business
Jobs, Real Estate, Investing...

Computers
Internet, Software, Hardware...

Games
Video Games, RPGs, Gambling...

Health
Fitness, Medicine, Alternative...

Home
Family, Consumers, Cooking...

Kids and Teens
Arts, School Time, Teen Life...

News
Media, Newspapers, Weather...

Recreation
Travel, Food, Outdoors, Humor...

Reference
Maps, Education, Libraries...

Regional
US, Canada, UK, Europe...

Science
Biology, Psychology, Physics...

Shopping
Autos, Clothing, Gifts...

Society
People, Religion, Issues...

Sports
Baseball, Soccer, Basketball...

World
Deutsch, Español, Français, Italiano, Japanese, Nederlands, Polska, Dansk, Svenska...

Become an Editor Help build the largest human-edited directory of the web

Copyright © 1998-2006 Netscape

5,291,609 sites - 72,144 editors - over 590,000 categories

Internet
开始 Music Lecture 9 WebSear... LIS531H-WebSearch... 4 Internet Explorer Adobe Reader - [h... 16:04

Matrix form of PageRank

The main difference from our previous damping equation

Basic PageRank (in matrix form) $\mathbf{w}_k = d\mathbf{w}_0 + (1 - d)\mathbf{B}\mathbf{w}_{k-1}$

$$\vec{Rank} = (1-\alpha)(M + D) \times \vec{Rank} + \alpha\vec{p}$$

- Here, if there is a link from page *j* to page *i*, then the matrix entry m_{ij} have the value 1/C(P_j) ,otherwise be 0.
- *p* be the *n*-dimensional column vector representing a uniform probability distribution over all nodes:

$$\vec{p} = \left[\frac{1}{n}\right]_{n\times 1}$$

- $\vec{d}$ be the *n*-dimensional column vector identifying the nodes with link-out *0*:

$$d_i = \begin{cases} 1 & \text{if } \deg(i) = 0, \\ 0 & \text{otherwise.} \end{cases}$$

- and

$$D = \vec{p} \times \vec{d}^T$$

Why we need the matrix *D*?

ODP-Biasing of PageRank

- Let T_j be the set of URLs in the ODP category c_j.
- computing the PageRank vector for topic c_j, in place of the uniform damping vector $\vec{p} = \left[\frac{1}{n}\right]_{n\times 1}$
- i.e. let $\vec{p} = \vec{v_j}$ and

$$v_{ji} = \begin{cases} \frac{1}{|T_j|} & i \in T_j, \\ 0 & i \notin T_j. \end{cases}$$

- The PageRank vector for topic c_j is given by $\vec{PR}(\alpha, \vec{v_j})$ that can be computed by the basic PR algorithm

Query-Time Importance Score

Categorizing the query *q*

- Let q_i' be the i^{th} term in the query (or query context) *q'*. Then given the query *q'*, compute for each c_j the following:

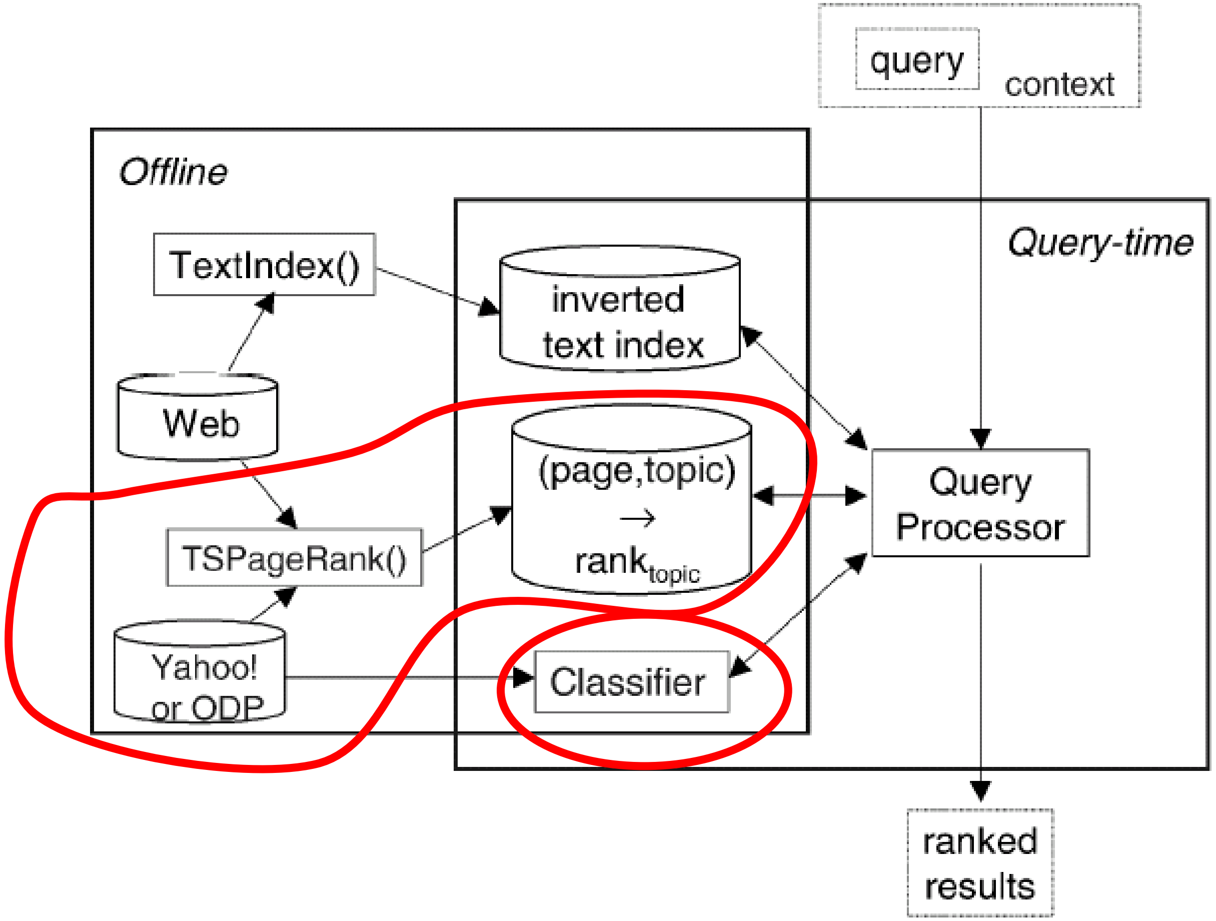
$$P(c_j|q') = \frac{p(c_j) \cdot P(q'|c_j)}{P(q')} \propto P(c_j) \cdot \prod_i P(q_i'|c_j)$$

Keep the same for all $\mathrm{c_j}$

- Let r_{jd} be the rank of document *d* given by the rank vector $\vec{PR}(\alpha, \vec{v}_j)$, then for the Web document *d*, the query-sensitive importance score s_{qd} is computed as follows

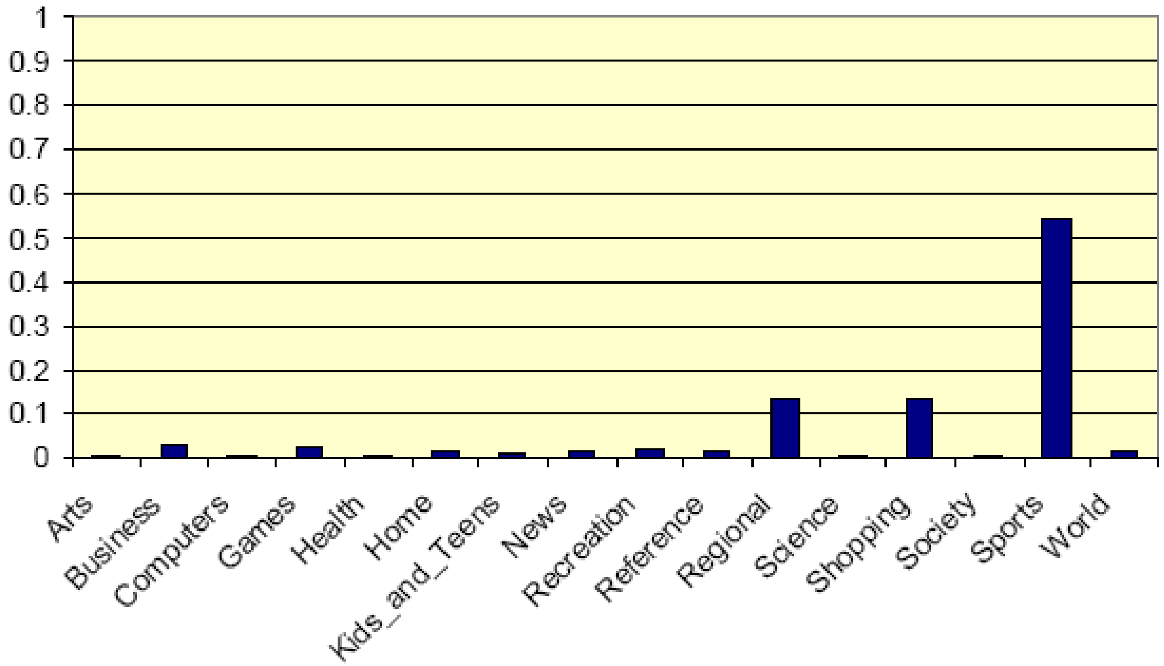
$$s_{qd} = \sum_j P(c_j|q') \cdot r_{jd}$$

SE System scheme with TS PageRank



Example Topic Distribution

- For the query ‘golf’, with no additional context the distribution of topic weights we would use is:



Experiment Results
TSPageRank vs. PageRank

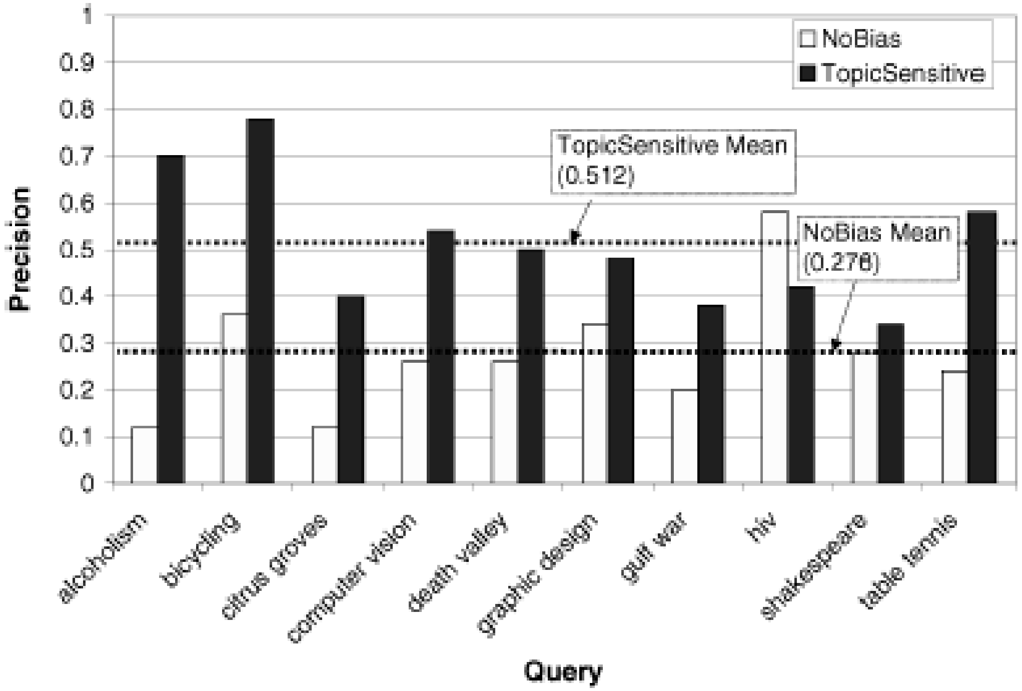


Fig. 3. Precision @ 10 results for our test queries. The average precision over the 10 queries is also shown.

From: Haveliwala, Topic-sensitive Pagerank: A Context-sensitive Ranking Algorithm for Web Search, 2003

Adaptive Search –Searching Context

- Rather than use query *q* itself, use the context of *q* to compute the topic distribution and PageRank values.
- The query context can be:
  - *The query history for the same user*
  - *The user provided text that contains the query terms*

Paper Discussion(Homework)

- Then, why is this paper named “topic-sensitive”?
- What’s the problem this paper was trying to address?
- What’s the role of matrix D?
- Why we need the ODP or other similar project for the computing of multiple page ranks for a given page?

Content

- Background
- Basic Idea
- PageRank Algorithm
- HITS: Hyperlink-Induced Topic Search
- Further Reading

HITS: Hyperlink-Induced Topic Search,

referred from: Pandu Nayak and Prabhakar Raghavan, Lecture 17: Link Analysis

Hyperlink-Induced Topic Search (HITS)

- 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*.

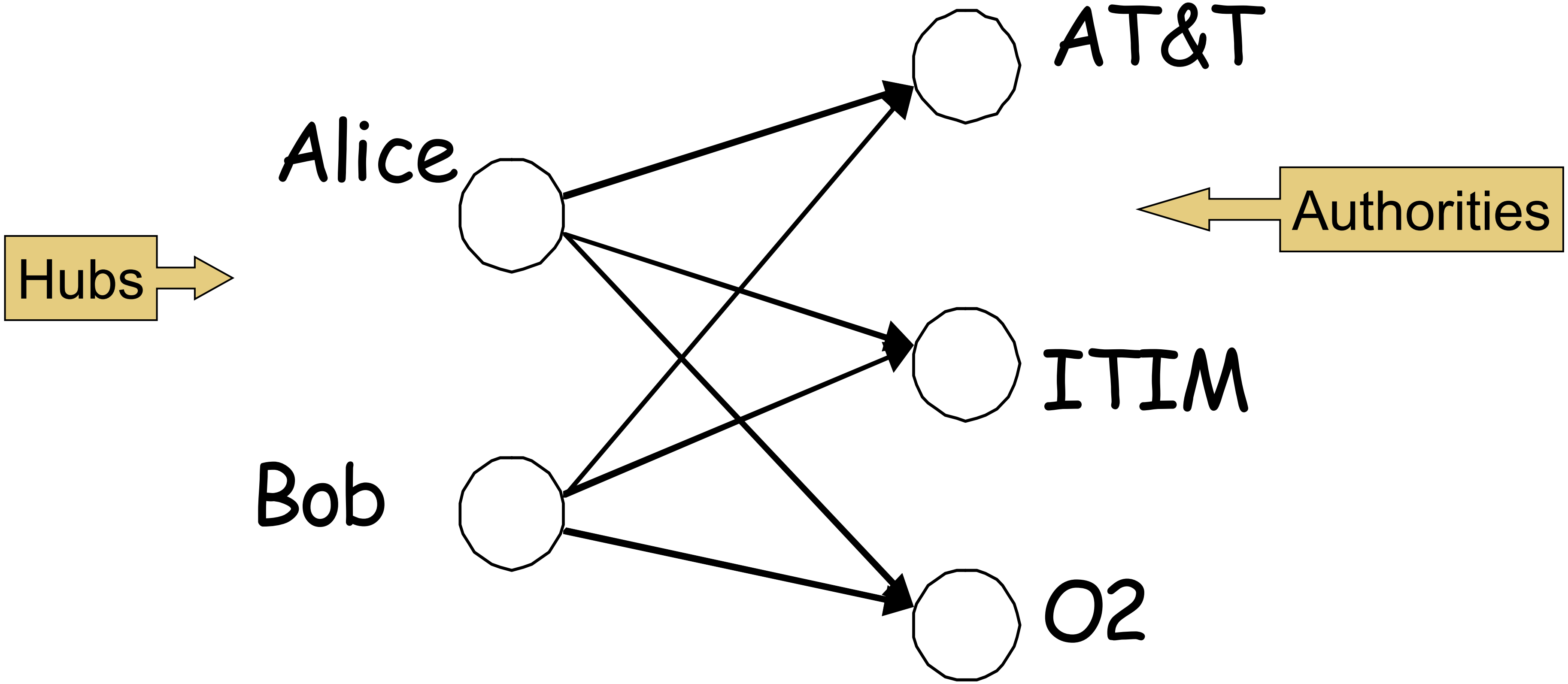
Hubs and Authorities

Basic Assumption

- 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



Mobile telecom companies

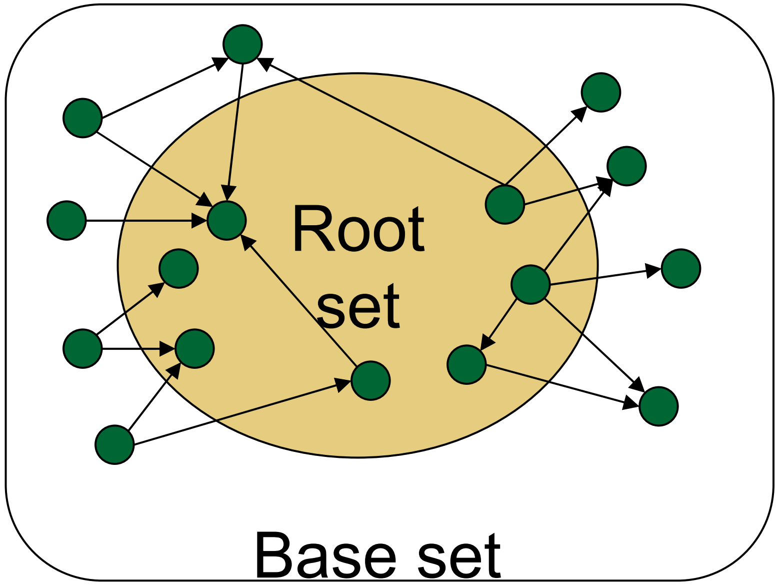
High-level scheme

- 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.

Construct the Base Set



Get in-links (and out-links) from a *connectivity server*

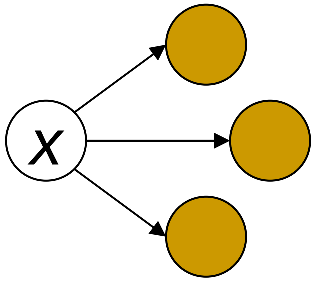
Distilling hubs and authorities

- 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)* ⇐ Key
- After iterations
  - output pages with highest *h()* scores as top hubs
  - highest *a()* scores as top authorities.

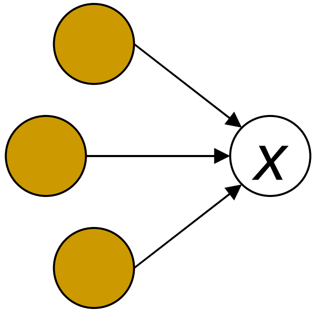
Iterative update

- 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.

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.

Content

- Background
- Basic Idea
- PageRank Algorithm
- HITS: Hyperlink-Induced Topic Search
- Further Reading

HITS: Hyperlink-Induced Topic Search,

referred from: Pandu Nayak and Prabhakar Raghavan, Lecture 17: Link Analysis

Further Reading

- **First publication that detail introduces Google's techniques by it's founders:**
- Brin, S. and Page, L., The Anatomy of a Large-Scale Hypertextual Web Search Engine. WWW 1998, April 14-18, 1998, Brisbane, Australia
- Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd, 'The PageRank Citation Ranking: Bringing Order to the Web', 1998
- Taher H. Haveliwala, 'Efficient Computation of PageRank', Stanford Technical Report, 1999

- **One of the important variations for PageRank:**
- Haveliwala T.H. Topic-sensitive Pagerank: A Context-sensitive Ranking Algorithm for Web Search, IEEE Trans. on Knowledge and Data Engineering, vol.15(4), 2003: 784-796

- **HITS:**
- Kleinberg, Jon (1999). "Authoritative sources in a hyperlinked environment" . Journal of the ACM 46 (5): 604–632.