relevant 相关的
Converge 收敛

Sparse matrix 稀疏矩阵

1. dead end

no out

2. spider trap

只在一个group里，不能出group

3. teleport : trap 和 dead end 执行策略.

∠

Some problems with Page Rank:

① 受特定主题影响: Topic-Specific PageRank
② 单一衡量: Hubs-and-Authorities 枢纽和权威
③ 垃圾链接: Trust Rank
Spam

Topic-Specific PageRank Allow be answered based user's interests
准备一个相关set, teleport对象从此处里找

计算方式: 只需改变A. $A_{ij} = \begin{cases} \beta M_{ij} + (1-\beta)/|S| & \text{if } i \in S \\ \beta M_{ij} + 0 & \text{else} \end{cases}$

关于选择topic: 可以让user从菜单选
可以识别查询主题
可以使用查询上下文.

Proximity in Graphs 图中接近度
短路径并不一定好.

Spam: web pages that are the result of any deliberate action to boost a web page's position in search engine results, incommensurate with page's real value.

早期 Page search: crawel the web, index pages 包含查询关键字.

早期 Page Rank:
- 关键字出现的次数.
- 关键字出现的位置, e.g.: title

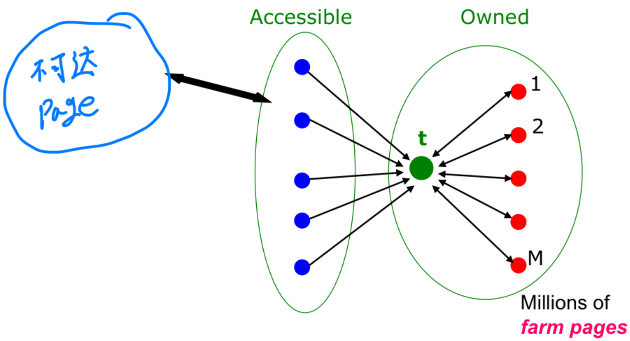
锚文本

对 early spam:
- 用 anchor text 或 link 周围文本而不是 link 自己说的文本.
- PageRank 可以帮助过滤掉 spam.

Round 2 - Spam farm: concentrate PageRank on a single page.

↳ 实现: ① link spam: create link structures that boost pageRank of a particular page

从 accessible pages (如博客评论区) 尽可能多创建指向 target page 的 link.



经典组织图.

N pages on the web

M pages spammer owned

x: PageRank by accessible

y: PageRank of t.

Rank of each farm page: $\frac{\beta y}{M} + \frac{1-\beta}{N}$

而 $y = x + \beta M\left[\frac{\beta y}{M} + \frac{1-\beta}{N}\right] + \frac{1-\beta}{N}$

$= x + \beta^2 y + \frac{\beta(1-\beta)M}{N} + \frac{1-\beta}{N}$ too small so ignore ✗

So: $y = \frac{x}{1-\beta^2} + c\frac{M}{N}$ 其中 $c = \frac{\beta}{1+\beta}$ (can make M large to make PageRank large)

Combating term spam:

- analyze text using statistical methods
- similar to email spam filtering.
- Detecting approximate duplicate pages.

Combating link spam:

Detection and blacklisting of structures that look like spam farms.

TrustRank: topic-specific PageRank with a teleport set of trusted pages

原理：近似隔离 approximate isolation

good pages rarely to point to be pages

每个页面信任度在0-1，信任度是加法，来自于它的in-links传递的信任度之和。

eg：P的trust是t_p，P has a set of out-links O_p. seed设为1.

P传递给它指向的q的trust为 $\frac{\beta t_p}{|O_p|}$

if teleport set 都是 trusted pages，则 TrustRank = PageRank

信任削减

- Trust attenuation：路径越长，越削减.
- Trust splitting：split越多，传递的trust越少.

→ 信任传播模型. → solution 1

How to pick a seed set of k pages:

① PageRank: 选k个PageRank最高的. 因为现实中很难让bad page有高PageRank

② Use trusted domains. eg: .edu .mil .gov.

Solution2: Spam Mass Estimate

r_p = Page P的 PageRank

r_p^+ = PageRank of P只从trusted pages teleport into 的.

from spam pages $r_p^- = r_p - r_p^+$

Spam mass of $p = \frac{r_p^-}{r_p^+}$

Hubs and Authorities:

HITS: Hypertext-Induced Topic Selection

评估 pages or documents 重要程度的方法.

每个page 有2个scores

① as hub (中心 → expert): 指向的authority的 value sum 和

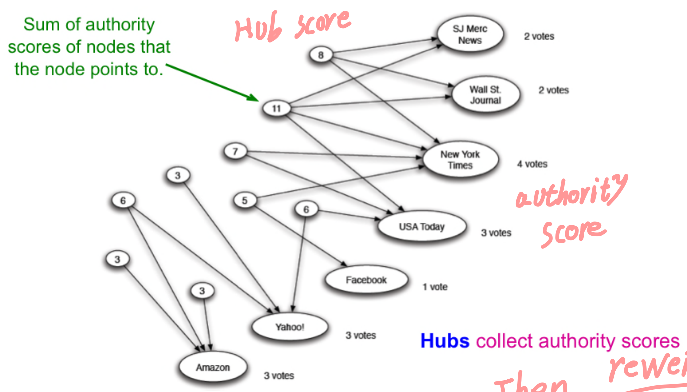
② as authority (→ content): 来自 hub 的 vote总和

Authority: Pages 包含有用信息

like 报纸 homepage

hub: 指向authority 的Pages

like 报纸列表

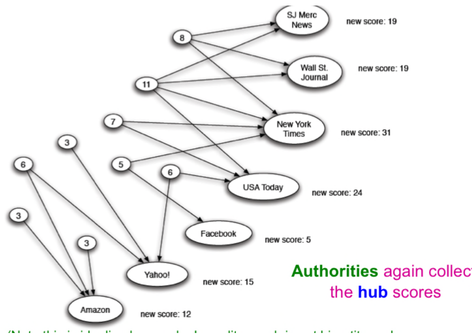


Hub score

authority score

Hubs collect authority scores

Then reweight



HITS 算法流程:

初始化: $a_j^{(0)} = \frac{1}{\sqrt{N}}$ $\quad h_j^{(0)} = \frac{1}{\sqrt{N}}$

$a_i^{(t+1)} = \sum_{j\to i} h_j^{(t)}$ then $h_i^{(t+1)} = \sum_{i\to j} a_j^{(t)}$

就是变成 $a_i^{(t+1)} = \frac{a_i^{(t)}}{\sqrt{\sum_i (a_i^{(t)})^2}}$

the 归一化: $\sum_i (a_i^{(t+1)})^2 = 1$, $\sum_j (h_j^{(t+1)})^2 = 1$

until 收敛.

HITS 矩阵计算:

邻接矩阵 $A_{N\times N}$, 其中 $\begin{cases} A_{ij}=1 & \text{if } i\to j \\ A_{ij}=0 & \text{else} \end{cases}$

故 $h_i = \sum_{i\to j} a_j$ 可以写成 $h_i = \sum_j A_{ij}\cdot a_j$ 即 $h = A\cdot a$

同理 $a_i = \sum_{j\to i} h_j$ 可以写成 $a_i = \sum_j A_{ji}\cdot h_j$ 即 $a = A^T\cdot h$

故

st1: set $a_i = h_i = \frac{1}{\sqrt{n}}$

st2: Repeat $\begin{cases} h = A\cdot a \\ a = A^T\cdot h \\ a\text{与}h\text{ 归一化} \end{cases}$ $\xrightarrow{\text{until}}$ $\begin{cases} \sum_i (h_i^{(t)} - h_i^{(t-1)})^2 < \varepsilon \\ \sum_i (a_i^{(t)} - a_i^{(t-1)})^2 < \varepsilon \end{cases}$

Then $a = A^T \cdot (\underbrace{A \cdot a}_{\text{new } h})$ 故更新成 $\begin{cases} a = A^T(Aa) = (A^TA)a \\ h = A(A^Th) = (AA^T)h \end{cases}$

Summary: PageRank 与 HITS 想解决的问题都是 u-v 的 in-link 价值. (from u to v 中)

$\begin{cases} \text{PageRank: 取决于 links to } u. \\ \text{HITS: 取决于 links out of } u. \end{cases}$