PageRank

Debapriyo Majumdar
Data Mining – Fall 2014
Indian Statistical Institute Kolkata

October 27, 2014

Search in the traditional way

Assumption: If term T is has a good "score" in document D, then D is about T

Query:
amir khan movie

amir khan
pk
movie

movie
amir khan

amir khan salman khan
buy this! pay here ☺
movie amir khan pk
shahrukh khan sachin
tendulkar pk pk pk

Devil: wants to
sell illegal stuff

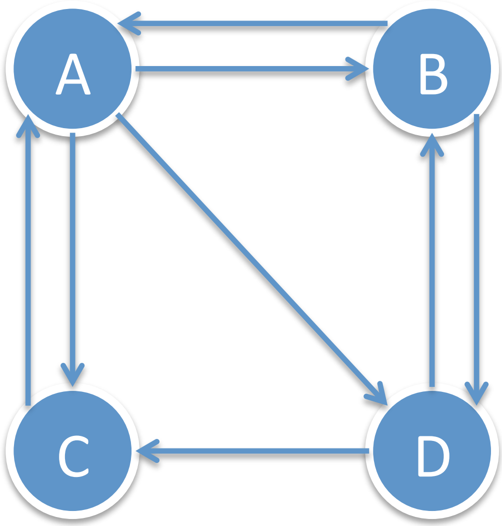
Term spam

PageRank

- Motivation
  - Users of the web are largely reasonable people
  - They put (more) links to useful pages
- PageRank
  - Named after Larry Page (co-founder of Google Inc.)
  - Patented by Stanford University, later bought by Google
- Approach
  - Importance (PageRank) of a webpage is influenced by the number and quality of links into the page
  - Search results ranked by term matching as well as PageRank
  - Intuition – Random web surfer model: a random surfer follows links and surfs the web. More likely to end up at more important pages
- Advantage: term spam cannot ensure in-links into those pages
- Many variations of PageRank

The random surfer model

A tiny web



Example courtesy: book by Leskovec, Rajaraman and Ullman

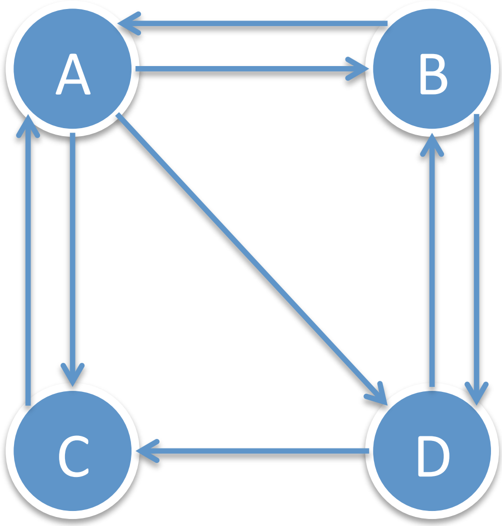
$$M = \begin{array}{c} \begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \\ \begin{matrix} \mathrm{A} & \mathrm{B} & \mathrm{C} & \mathrm{D} \end{matrix} \end{array}$$

- Web graph, links are directed edges
  - Assume equal weights in this example
  - If a surfer starts at A, with probability 1/3 each, may go to B, C, or D
  - If a surfer starts at B, with probability 1/2 each may go to A or D
  - Can define a transition matrix
- Markov process:
  - Future state solely based on present

$$M_{ij} = \mathrm{P}[i \rightarrow j \text{ in next step} \mid \text{presently in } i]$$

The random surfer model

A tiny web



Example courtesy: book by Leskovec, Rajaraman and Ullman

- Random surfer: initially at any position, with equal probability $1/n$
- Distribution (column) vector $v = (1/n, \ldots, 1/n)$
- Probability distribution for her location after one step?
- Distribution vector: Mv
- How about two steps? $M^2 v$

- Initially at A (1/4), A → A : not possible
- Initially at B (1/4), B → A (1/2), overall prob =1/8
- Initially at C (1/4), C → A (1), overall prob = ¼
- Initially at D (1/4), no route to A in one step

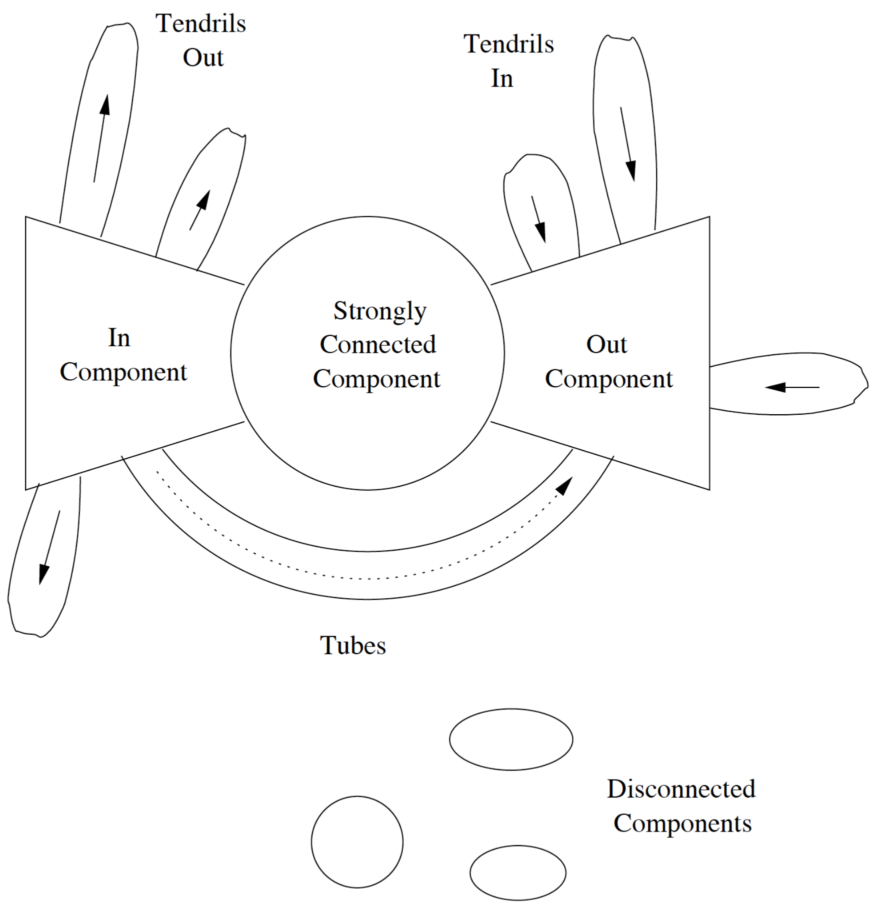
$$M = \begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \quad v = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \quad Mv = \begin{bmatrix} 0 + 1/8 + 1/4 + 0 = 9/24 \\ 1/12 + 0 + 0 + 1/8 = 5/24 \\ 1/12 + 0 + 0 + 1/8 = 5/24 \\ 1/12 + 1/8 + 0 + 0 = 5/24 \end{bmatrix}$$

Perron – Frobenius theorem

- The probability distribution converges to a limiting distribution (when $Mv = v$) if
  - The graph is strongly connected (possible to get from any node to any other node)
  - No dead ends (each node has some outgoing edge)
- The limiting v is an eigenvector of M with eigenvalue 1
- Note: M is (left) stochastic (each column sum is 1)
  - Hence 1 is the largest eigenvalue
  - Then v is the principal eigenvector of M
- Method for computing the limiting distribution (PageRank!)

```
Initialize v = (1/n, … , 1/n)
while (Mv − v > ε) {
      v = Mv
}
```

Structure of the web

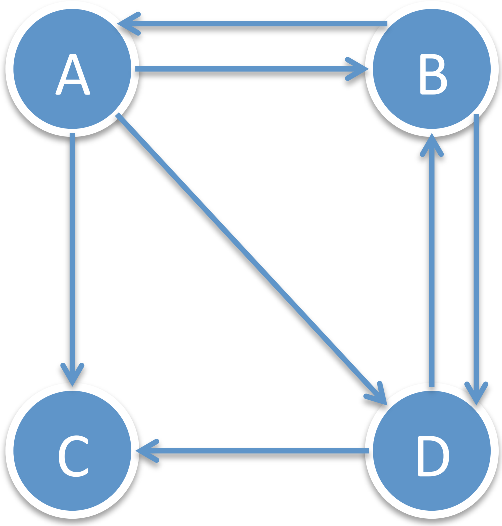


Picture courtesy: book by Leskovec, Rajaraman and Ullman

- The web is *not* strongly connected ☹
- An early study of the web showed
  - One large strongly connected component
  - Several other components
- Requires modification to PageRank approach
- Two main problems
  1. Dead ends: a page with no outlink
  2. Spider traps: group of pages, outlinks only within themselves

Dead ends

A tiny web



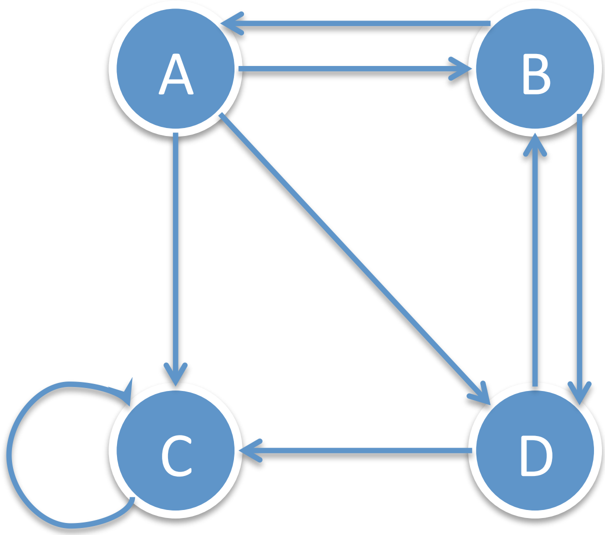
Example courtesy: book by Leskovec, Rajaraman and Ullman

- Let's make C a dead end
- *M* is not stochastic anymore, rather *substochastic*
  - The 3rd column sum = 0 (not 1)
- Now the iteration $v := Mv$ takes all probabilities to zero

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \quad v = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \quad \begin{array}{c} Mv \\ \begin{bmatrix} 3/24 \\ 5/24 \\ 5/24 \\ 5/24 \end{bmatrix} \end{array}, \begin{array}{c} M^2v \\ \begin{bmatrix} 5/48 \\ 7/48 \\ 7/48 \\ 7/48 \end{bmatrix} \end{array} \quad \cdots \longrightarrow \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Spider traps

A tiny web



Example courtesy: book by Leskovec, Rajaraman and Ullman

- Let C be a one node spider trap
- Now the iteration $v := Mv$ takes all probabilities to zero except the spider trap
- The spider trap gets all the PageRank

$$M = \begin{bmatrix} 0 & 1/2 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 1 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \quad v = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} \quad \overset{Mv}{\begin{bmatrix} 3/24 \\ 5/24 \\ 11/24 \\ 5/24 \end{bmatrix}}, \overset{M^2 v}{\begin{bmatrix} 5/48 \\ 7/48 \\ 29/48 \\ 7/48 \end{bmatrix}} \cdots \rightarrow \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Taxation

- Approach to handle dead-ends and spider traps
- Taxation
  - The surfer may leave the web with some probability
  - A new surfer may start at any node with some probability
- Idealized PageRank: iterate $v_k = Mv_{k\text{-}1}$
- PageRank with taxation

$$v_k = \beta M v_{k-1} + (1-\beta)\frac{\mathbf{e}}{n}$$

where β is a constant, usually between 0.8 and 0.9

$\mathbf{e} = (1, \ldots, 1)$

with probability β continue to an outlink

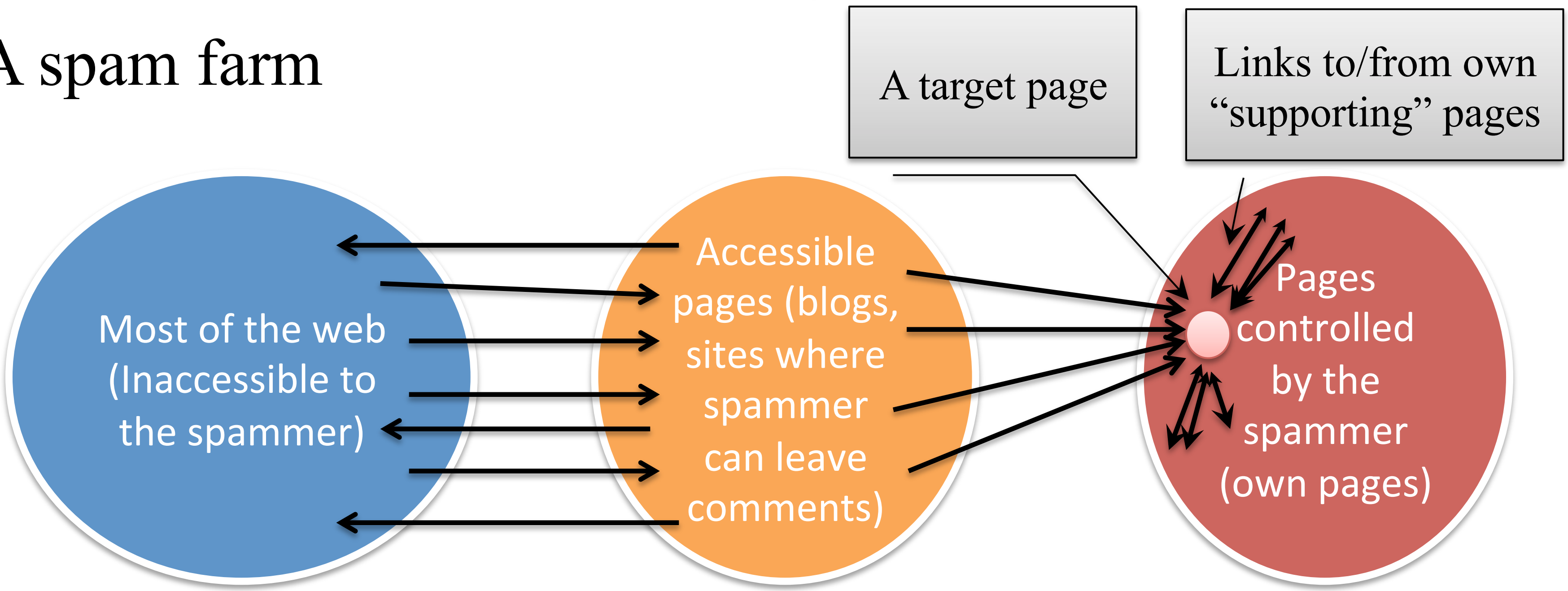
with probability (1-β) teleport (leave and join at another node)

Link spam

Google took care of the term spam, but …

The devil **still** wants to sell illegal stuff

A spam farm



Analysis of a spam farm

- Setting
  - Total #of pages in the web = n
  - Target page T, with m supporting pages
  - Let x be the PageRank contributed by accessible pages (sum of all PageRank of accessible pages times β)
  - How much y = PageRank of the target page can be?
- PageRank of every supporting page

$$\frac{\beta y}{m} + \frac{1-\beta}{n}$$

Contribution from the target page with PageRank y

Share of PageRank among all pages in the web

Analysis of a spam farm (continued)

- Three sources contribute to PageRank
  1. Contribution from accessible pages = x
  2. Contribution from supporting pages = $\beta\left(\frac{\beta y}{m}+\frac{1-\beta}{n}\right)$
  3. The n-th share of the fraction $(1-\beta)/n$ [negligible]
- So, we have

$$y = x + \beta m\left(\frac{\beta y}{m}+\frac{1-\beta}{n}\right)$$

$$= x + \beta^2 y + \beta(1-\beta)\frac{m}{n}$$

- Solving for y, we get

$$y = \frac{x}{1-\beta^2} + \frac{\beta}{(1+\beta)} \times \frac{m}{n}$$

If β = 0.85, then
$\boldsymbol{y = 3.6 \times x + 0.46 \times m/n}$

- External contribution up by 3.6 times, plus 46% of the fraction of the PageRank from the web

TrustRank and Spam Mass

- A set S of trustworthy pages where the spammers cannot place links
  - Wikipedia (after moderation), university pages, …
- Compute TrustRank

$$v_k = \beta M v_{k-1} + (1-\beta)\frac{\mathbf{e}_S}{|S|}$$

- The random surfers are introduced only at trusted pages
- Spam mass = PageRank – TrustRank
- High spam mass ➔ likely to be spam

References

- Mining of Massive Datasets: Leskovec, Rajaraman and Ullman