Data mining

Web Data Mining
PageRank, Hubs and Authorities

University of Szeged

Why ranking web pages is useful?

- We are "starving for knowledge"
- It earns Google a bunch of money. How ?

How does the Web looks like?

- Big *strongly* connected *central* component
- Some smaller strongly connected components that attach to the central one through in-, or out-edged
- Direct links between the above mentioned two components
- Isolated components

What is needed for efficient information retrieval?

- We need to know what terms (concepts) are included in documents $\Rightarrow$ indexing
- We need to be able to return the set of documents containing some (possibly multi-word) search queries
  - Lemmatization (especially important for Hungarian and other agglutinative languages)
  - Weighting the within-document importance of words (often called terms), e.g. *tf-idf* weighting (and its variants)
- Ability of ranking those documents that contain some query string according to their expected utility

Ranking of web pages

- How can we measure the importance of a web page?
    - Number of visitors?
    - Using links?
- Define importance (rank) of a node as a recursive function of the importance of those pages which point to it

$$rank(j) = \sum_{i \to j} \frac{rank(i)}{\text{degree}(i)}$$

- What is caused by a node having a high in-degree? What about out-degree?

Stochastic matrices

- M is row stochastic if $\forall m_{i,j} \geq 0$ and $\forall i \in \{1, \ldots, n\} \sum_{j=1}^{n} m_{i,j} = 1$
- Column stochasticity has a similar definition
- What is the meaning of a values of M, i.e. $m_{i,j}$? $\Rightarrow$ M is a matrix describing the (state) transition of a (stochastic) Markov process
- What is the meaning of the product $p_1^\intercal = p_0^\intercal M$?
- How can we interpret the product $p_i^\intercal = p_{i-1}^\intercal M = p_0^\intercal M^i$?

Stationary distribution for stochastic matrices

- Irreducibility: there exists a directed path between any pair of points
- Aperiodicity: $\forall i \exists n' : P(\sigma_n = i | \sigma_0 = i, n > n') > 0$
  $\Rightarrow \exists p^{*\intercal}$ stochastic vector being the stationary distribution of M
- Stationary distribution: $p^{*\intercal} = \lim_{t \to \infty} p_0^\intercal M^t$
  - Slightly differently: such a $p_t^\intercal$ for which $p_t^\intercal \approx p_t^\intercal M$
- Power iteration: keep $p^\intercal$ multiplying by M until convergence
- Convergence can be defined as a function of the changes in $p_t^\intercal$

$$M = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}$$

$$\begin{tikzcd} 2 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] & 4 \arrow[d] \\ 1 \arrow[ur] \arrow[r, leftrightarrow] & 3 \end{tikzcd}$$

Ergodic Markov process – Power iteration

$$M^2 = \begin{pmatrix} \frac{1}{2} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ 0 & \frac{5}{12} & \frac{5}{12} & \frac{1}{6} \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{3}{4} & 0 & 0 & \frac{1}{4} \end{pmatrix}$$

$$M^3 = \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{5}{8} & \frac{1}{12} & \frac{1}{12} & \frac{5}{24} \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ 0 & \frac{3}{8} & \frac{3}{8} & \frac{1}{4} \end{pmatrix}$$

p_0	p_1	p_2	p_3	...	p_6	...	p_9
0.25	0.375	0.313	0.344	...	0.332	...	0.333
0.25	0.208	0.229	0.219	...	0.224	...	0.222
0.25	0.208	0.229	0.219	...	0.224	...	0.222
0.25	0.208	0.229	0.219	...	0.224	...	0.222

Is it surprising that the rank of 3 points happens to be the same for all the iterations?

The Web is not ergodic however – Dead ends

- There might be pages with no outgoing links
- Such pages make the importance traversing from the network to "leak out"
- The simplest such graph
$$M = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & 0 \end{pmatrix} \Rightarrow M^i = \frac{1}{2^{i-1}} M \Rightarrow \forall v, \lim_{i \to \infty} v^\intercal M^i = \vec{0}$$

Resolving dead ends – example

- Remove dead ends until it gets dead ends-free
- By removing nodes, we might generate new dead ends
- Determine the ranks for the nodes in the graph that is left and infer the rank of the removed nodes according to the recursive formula
- Doing so the ranks are no longer guaranteed to sum to 1 (we can do renormalization of the ranks afterwards however)

$$M = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The Web is not ergodic however – Spider traps

- "Traps" in the network "without any exit" that accumulates the importances for its members
- Simplest form: a node with a single self loop (ofc. we can think of larger traps as well)

$$M = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}$$

p_0	p_1	p_2	p_3	...	p_6	...	p_9
0.25	0.125	0.104	0.073	...	0.029	...	0.011
0.25	0.208	0.146	0.108	...	0.042	...	0.016
0.25	0.458	0.604	0.712	...	0.888	...	0.957
0.25	0.208	0.146	0.108	...	0.042	...	0.016

Resolving spider traps – example

- For modeling the behavior of a random surfer include the possibility of performing "teleportation"
- Let the random surfer follow one of the directly accessible neighbors with $\beta(\approx 0.8 - 0.9)$ probability
- With probability $(1-\beta)$ move to *any* of the sites $\Rightarrow$ we can get out of traps that way
- $p^\intercal_{(i+1)} = p^\intercal_{(i)}\beta M + (1-\beta)\frac{\vec{1}}{\text{number of nodes}}$
- We can think of replacing M with $(\beta M + \frac{(1-\beta)}{n}\vec{1}\vec{1}^\intercal)$ with n being the number of websites

Resolving spider traps – example

$$\beta M = \begin{pmatrix} 0 & \frac{4}{15} & \frac{4}{15} & \frac{4}{15} \\ \frac{2}{5} & 0 & 0 & \frac{2}{5} \\ 0 & 0 & \frac{4}{5} & 0 \\ 0 & \frac{2}{5} & \frac{2}{5} & 0 \end{pmatrix}$$

What β was used here?

p_0	p_1	p_2	p_3	...	p_6	...	p_9
0.25	0.150	0.137	0.121	...	0.105	...	0.101
0.25	0.217	0.177	0.157	...	0.134	...	0.130
0.25	0.417	0.510	0.565	...	0.627	...	0.639
0.25	0.217	0.177	0.157	...	0.134	...	0.130

Personalized PageRank

- How objective is the ordering of web pages determined by PageRank?
- For different people different pages count as relevant
- Should a PageRank distribution be determined for every person?
- Even the same person might find different sites as relevant in different scenarios
- Should there be a different PageRank distribution determined for the combination of every person and search scenarios?

Personalized PageRank – "Biased" random walks

- A user is typically interested in documents related to a certain sense/topic (e.g. jaguar related to nature or cars)
- It is possible to predict the kind of topic the user might be interested
  - Browsing history
  - Where the search is conducted (e.g. search bow on a sports site)
  - Users might indicate (implicitly or explicitly) the topic they wish to see results from
- Can provide different PageRank distributions for different search needs

Personalized PageRank – Example

- $p_{(i+1)}^{\intercal} = p_i^{\intercal}\beta M + (1-\beta)\frac{\vec{1_r}}{|\text{relevant sited for some topic}|}$
- $\vec{1_r}$ is special in that it contains 1s for those positions only which correspond to relevant sites

$$\beta M = \begin{pmatrix} 0 & \frac{4}{15} & \frac{4}{15} & \frac{4}{15} \\ \frac{2}{5} & 0 & 0 & \frac{2}{5} \\ \frac{4}{5} & 0 & 0 & 0 \\ 0 & \frac{2}{5} & \frac{2}{5} & 0 \end{pmatrix}$$

Hacking PageRank

- Irrelevant pages might be made seemingly more relevant by forming link farms (i.e. sites the only reason of which is to point to some sites)
- **TrustRank**: Applying Personalized PageRank in a way the the random walker is biased towards trustworthy nodes
  - Trustworthy sites can be determined relying on human labor and they can be detected using some automatism

Hubs and Authorities Algorithm

- A similar approach to PageRank, however, pages are assigned two different scores according to their extent of hubness and authoritiness
- Recursive nature
- Relevant pages with high authority score are those for which many pages with high hubness point to
- The same applies the other way around

Hubs and Authorities formally

- Takes A as input, i.e. the adjacency matrix of web pages
  - Is A a stochastic matrix?
- The i^{th} components of vectors h and a refer to the hub and authority score of the i^{th} site, respectively
- $\mathrm{h} = \xi A\mathrm{a} : \sum_{i=1}^{n} h_i = 1$ or $max(\mathrm{h}) = 1$ and
  $\mathrm{a} = \nu A^{\intercal}\mathrm{h} : \sum_{i=1}^{n} a_i = 1$ or $max(\mathrm{a}) = 1$
- Slightly differently: $\mathrm{h} = \xi\nu AA^{\intercal}\mathrm{h}$ and $\mathrm{a} = \nu\xi A^{\intercal}A\mathrm{a}$
- $AA^{\intercal}$ and $A^{\intercal}A$ can easily get dense $\Rightarrow$ apply asynchronous update

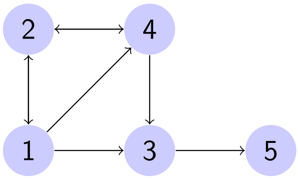
Pseudocode of the asynchronous HITS

Algorithm 1 HITS algorithm

Input: adjacency matrix A
Output: vectors a, h

```
h := \vec{1}
while not converged do
    a = A^T h
    a = a/max(a)
    h = A a
    h = h/max(h)
end while
return a, h
```

HITS algorithm – example



$$A = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

h_0	a_1	h_1	a_2	h_2	a_3	h_3	...	a_{10}	h_{10}
1	0.5	1	0.3	1	0.24	1	...	0.21	1
1	1	0.5	1	0.41	1	0.38	...	1	0.36
1	1	0.17	1	0.03	1	0.007	...	1	0
1	1	0.67	0.9	0.69	0.84	0.71	...	0.79	0.72
1	0.5	0	0.1	0	0.02	0	...	3,5e-07	0