Introduction to Search Engine Technology
Link Structure Analysis - Continued

Ronny Lempel
Outbrain

Last Lecture - Recap
- HITS: topic-specific algorithm (aka “topic distillation” algorithm)
  - Assigns each page two scores – a hub score and an authority score – with respect to a topic
- PageRank: query independent algorithm
  - Assigns each page a single, global importance score
- Both algorithms reduced to the computation of principal eigenvectors of certain matrices
Today’s Agenda

1. Graph modifications in link analysis algorithms
2. Topic-Sensitive PageRank
3. Stability and monotonicity of PageRank
4. SALSA – HITS with a random-walk twist
5. TKC Effect – qualitative difference between HITS and SALSA

Graph Modifications in Link-Analysis Algorithms

Several works suggested performing modifications, that intuitively should improve precision, to the links of the analyzed collections:

1. Delete irrelevant elements (pages, links) from the collection.
   - Non-informative links
   - Pages that are deemed irrelevant (mostly by dissimilarity of content to the query), and their incident links [Bharat and Henzinger, 1998]
2. Assign varying (positive) link weights to the non-deleted links.
   - Similarity of anchor text to the query [CLEVER]
   - Links incident to pre-defined relevant pages [CLEVER]
   - Multiple links from pages of site A to pages of site B [Bharat and Henzinger, 1998]
   - Some modifications are only applicable to topic distillation algorithms
Topic Sensitive PageRank
[T. Haveliwala, 2002]

- A topic T is defined by a set of on-topic pages $S_T$
- A T-biased PageRank is PageRank where the random jumps (teleportations) land u.a.r. on $S_T$ rather than on any arbitrary Web page
- Recall our interpretation of PageRank from last week, as walking random paths of geometrically distributed lengths between resets
  - Here, a reset returns to some on-topic page

If we assume that pages tend to link to pages with topical affinity, short paths starting at $S_T$ will not stray too far away from on-topic pages
- Hence the PageRanks will be T-biased
- Pages unreachable from $S_T$ will have a T-biased PageRank of 0

Where would be a good place to find sets $S_T$ for certain topics?
- The pages classified under the 16 top-level topics of the Open Directory Project (see next slide)
16 PageRank vectors are computed, \( PR_1, \ldots, PR_{16} \).

Given a query \( q \), its affinity to the 16 topics \( T_1, \ldots, T_{16} \) is computed:

- Based on the probability of generating the query by the language model induced by the set of pages \( S_T \).
- A distribution vector \([\alpha_1, \ldots, \alpha_{16}]\) is computed, where \( \alpha_j \sim \text{Prob}(q \mid \text{language model of } S_T) \).

The PageRank vector that will be used to serve \( q \) is \( PR_q = \sum \alpha_j PR_j \).

The idea of biasing PageRank’s random jumps is also used for personalized PageRank [Jeh and Widom 2003].
L₁-Stability – Definition and Results

- L₁-stability: if the L₁-change of the scores (following a perturbation of a graph) is bounded by a linear function of the scores of the perturbed nodes
  - PageRank is L₁-stable on all graphs
  - Why do we care?

Why Should We Care About Stability?

- Ranking algorithms should be robust – small changes in the input should not dramatically alter the results
- The Web changes constantly: pages and links are constantly being added, deleted and modified
- Link-spammers attempt to manipulate link-based ranking techniques
- Said spammers control small, local portions of the Web and cannot alter the Web’s graph in a radical fashion
  - Stability implies less susceptibility to link spamming
L\(_1\)-Stability of PageRank

[Ng, Zheng and Jordan 2001]

- Let G=(V,E) be a graph, and let Q⊂V
- Let H=(V,E\(^-\)) be a graph where the out-links of q∈Q are *perturbed* in some manner (links either added or deleted)
- Let π\(_G\) and π\(_H\) denote the PageRank vectors of G and H respectively
- Theorem: \(||π\(_G\)-π\(_H\)|||\(_1\) ≤ 2 \* \(d / (1-d)\) \* Σ\(_q∈Q\)π\(_G\)(q)

(1-d is the random jump probability of PageRank)

---

L\(_1\)-Stability of PageRank (cont.)

- Theorem: \(||π\(_G\)-π\(_H\)|||\(_1\) ≤ 2 \* \(d / (1-d)\) \* Σ\(_q∈Q\)π\(_G\)(q)
- Proof: consider two coupled random walks, \(X_g(t)\) and \(Y_h(t)\) that behave as follows:
  - Choose a node v∈V u.a.r. and set \(X_g(0) = Y_h(0) = v\)
  - Both walks will share all decisions whether to perform a random jump or to follow an out-link out of their current node
  - For all times t such that \(X_g(t)=Y_h(t)\) following a random jump, set \(X_g(t) = Y_h(t)\) - i.e. the walks unite after every “reset”
    - Recall that this happens with probability 1-d
  - Whenever \(X_g(t)=Y_h(t)\)∉Q and the walks decide to follow an outlink, they choose the same outlink and so \(X_g(t+1)=Y_h(t+1)\)
  - Whenever either \(X_g(t)=Y_h(t)\)∉Q or \(X_g(t)≠Y_h(t)\), and the walks decide to follow an outlink, each walk selects the outlink independently
    - So probably \(X_g(t+1)=Y_h(t+1)\)
Define $s_t = \Pr[X_G(t) \neq Y_H(t)]$

$s_{t+1} = \Pr[X_G(t+1) \neq Y_H(t+1) \mid \text{random jump after time } t] \cdot (1-d) + \Pr[X_G(t+1) \neq Y_H(t+1) \mid \text{link-step after time } t] \cdot d$

$= 0 \quad // \text{the two walks always unite after random jumps}$

$\leq s_t \cdot d + \Pr[X_G(t)=Y_H(t), X_G(t) \in Q \mid \text{link-step after time } t] \cdot d$

$\leq s_t \cdot d + \Pr[X_G(t) \in Q] \cdot d$

$= d \cdot \left[ s_t + \sum_{q \in Q} \pi_G(q) \right]$

Conclusion: $s_{t+1} \leq d \cdot \left[ s_t + \sum_{q \in Q} \pi_G(q) \right]$

L_1-Stability of PageRank (cont.)

Conclusion: $s_{t+1} \leq d \cdot \left[ s_t + \sum_{q \in Q} \pi_G(q) \right]$

Lemma: $s_t \leq \left[ d / (1-d) \right] \cdot \sum_{q \in Q} \pi_G(q)$

Proof by induction

Recall that $s_0 = 0$, and so trivially $s_1 \leq d \cdot \sum_{q \in Q} \pi_G(q) = d \cdot \sum_{q \in Q} \pi_G(q) / (1-d)$

Step: $s_{t+1} \leq d \cdot \left[ s_t + \sum_{q \in Q} \pi_G(q) \right]$

$\leq d \cdot \left[ \sum_{q \in Q} \pi_G(q) / (1-d) + \sum_{q \in Q} \pi_G(q) \right]$

$\leq \left[ d / (1-d) \right] \cdot \left[ d \cdot \sum_{q \in Q} \pi_G(q) + (1-d) \cdot \sum_{q \in Q} \pi_G(q) \right]$

$= \left[ d / (1-d) \right] \cdot \sum_{q \in Q} \pi_G(q)$

This is an asymptotic bound on the probability that $Y_H$ will stray from $X_G$, and by the Coupling Lemma the L_1-distance of the two limiting distributions is bounded by twice the bound on $s_t$

Hence $||\pi_G - \pi_H||_1 \leq 2 \left[ d / (1-d) \right] \cdot \sum_{q \in Q} \pi_G(q)$, as claimed
**L₁-Stability of PageRank - Intuition**

- \( Y \) only strays from \( X \) when it reaches a node of \( Q \) AND decides to follow an outlink.
- The two walks will unite no later than the next random jump.
- Expected number of steps \( X \) and \( Y \) will spend apart: the expectancy of a Geom(1-\( d \)) random variable, minus one.
  - Hence, \( \frac{d}{1-d} \) steps apart per visit to a node of \( Q \).
  - When considering the probability of hitting a node of \( Q \) in the first place: \( \frac{d}{1-d} \times \sum_{q \in Q} \pi_G(q) \).
- That’s the portion of time (=probability mass) that the two walks spend apart.
- If the time apart is spent by the two walks at disjoint sets of nodes, the \( L_1 \) distance between the distributions will reflect twice that time.

---

**Monotonicity of PageRank**

[Chien, Dwork, Kumar, Simon and Sivakumar 2002]

- Let \( G \) be a graph, and let \( v, u \) be nodes in \( G \) such that \( v \) doesn’t link to \( u \).
- Create \( G' \) by adding \( v \rightarrow u \) to \( G \). Then,
  - \( \text{PageRank}_G(v) < \text{PageRank}_G'(u) \)
  - The ranking of \( u \)’s PageRank in \( G' \) will not be lower than the ranking of \( u \)’s PageRank in \( G \).
    - In particular, for any node \( q \),
      \( \text{PageRank}_G(v) \geq \text{PageRank}_G(q) \Rightarrow \text{PageRank}_{G'}(u) \geq \text{PageRank}_{G'}(q) \)
- The result actually applies not only to PageRank but to general Markov chains (under some Ergodic conditions).
SALSA – Stochastic Approach to Link Structure Analysis

SALSA, like HITS, is a topic-distillation algorithm that aims to assign pages both hub and authority scores.

- SALSA analyzes the same topic-centric graph as HITS, but splits each node into two – a “hub personality” without in-links and an “authority personality” without out-links.
- Examines the resulting bipartite graph.

SALSA (cont.)

Innovation: incorporate stochastic analysis with the authority-hub paradigm.

- Examine two separate random walk Markov chains: an authority chain \( A \), and a hub chain \( H \).
- A single step in each chain is composed of two link traversals on the Web - one link forward, and one link backwards.
- The principal community of each type: the most frequently visited pages in the corresponding Markov Chain.
Formally, the transition probability matrix:

\[ [P_A]_{i,j} = \sum \{k | k \rightarrow i, k \rightarrow j\} \ (i_{in})^{-1} (k_{out})^{-1} \]

- The transition probabilities induce a probability distribution on the authorities (hubs) in the authority (hub) Markov chain.
  - If the chains are not irreducible, the probability depends on the initial distribution (chosen to be uniform).
  - The principal community of authorities (hubs) is defined as the \( k \) most probable pages in the authority (hub) chain.
  - While one can compute the scores by calculating the principal eigenvector of the stochastic transition matrix, a more efficient way exists.
SALSA: Analysis (cont.)

Mathematical Analysis of SALSA leads to the following theorem: SALSA’s authority weights reflect the normalized in-degree of each page, multiplied by the relative size of the page’s component in the authority side of the graph.

\[ a(x) = \frac{3}{3 + 5} \times \frac{4}{4 + 2} = 0.25 \]

SALSA: Proof for Irreducible Authority Chains

- The proof assumes a weighted graph, in which the link \( k \rightarrow j \) has weight \( w(k \rightarrow j) \).
- The examples so far assumed that all links have a weight of 1.
- Define \( W \) as the sum of all links weights.
- Define a distribution vector \( \pi \) by \( \pi_j = \frac{d_{in}(j)}{W} \), where \( d_{in}(j) \) is the sum of weights of \( j \)’s incoming links.
- Similarly, \( d_{out}(k) \) is the sum of weights of \( k \)’s outgoing links.
- It is enough to prove that \( \pi P_A = \pi \), since \( P_A \) has a single stationary eigenvector (Ergodic Theorem).
  - Recall that \( P_A \) is the transition matrix of the authority chain.
  - \( P_A \) is always aperiodic.
### The Tightly-Knit Community Effect

- A tightly-knit community is a small set of densely interconnected pages.
- In a multi-topic authority-connected collection, TKCs often correspond to one narrow aspect of a topic with zealous followers (hub writers).
  - Sometimes, TKCs are the result of topic drift.
The Tightly-Knit Community Effect

- HITS was shown to have a tendency to assign high authority scores to TKCs – this is called the “TKC Effect”
  - May cause results of unambiguous queries to drift
  - May cause results of ambiguous queries to focus on just one, sometimes narrow aspect of the topic
  - A bias toward TKCs may be indicative of susceptibility to link spamming
- SALSA was empirically shown to be less susceptible to the TKC effect
- Next slide – an infinite family of synthetic graphs exhibiting the qualitative difference between the two algorithms

Full connectivity: \( s = \binom{k^2+2k}{k-1} \)

Pair-wise connectivity: \( \binom{k+1}{2}^2 \)

Partial connectivity: \( L = \binom{k+1}{2} \) choose \( k \)
Link Analysis Algorithms - Summary

- Many variants and refinements of both HITS and PageRank have been proposed.
- Other approaches include:
  - Max-flow techniques [Flake et al., SIGKDD 2000]
  - Machine learning and Bayesian techniques
- Examples of applications:
  - Ranking pages (topic specific/global importance/ personalized rankings)
  - Categorization, clustering, finding related pages
  - Identifying virtual communities
- Computational issues:
  - Distributed computations of eigenvectors of massive, sparse matrices
  - Convergence acceleration, approximations
- A wealth of literature