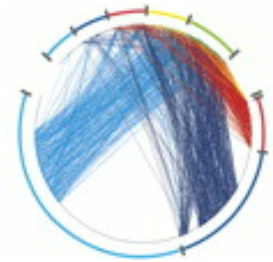


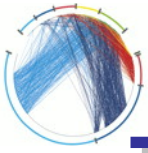
Theoretical analysis of Link Analysis Ranking

Panayiotis Tsaparas
University of Helsinki
HIIT-BRU



Link Analysis Ranking

- Link Analysis Ranking (LAR) algorithm:
 - Given a (directed) graph G , determine the importance of the nodes in the graph using the information of the edges (links) between the nodes.
- Intuition:
 - A link from node p to node q denotes endorsement. Node p considers node q an authority on a subject
 - mine the graph of recommendations, assign an **authority value** to every page
- Applications:
 - Assess the importance of Web pages using link information.
 - Recommendation systems



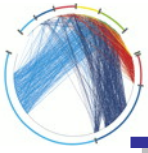
Why theoretical analysis of Link Analysis Ranking?

- Plethora of LAR algorithms: we need a formal way to compare and analyze them
- Need to define properties that are useful
 - stability of the algorithm
- Axiomatic characterization of LAR algorithms
 - extension of social choice theory to recommendation systems



Link Analysis Ranking algorithm

- A LAR algorithm is a function that maps a graph to a real vector
$$A:G_n \rightarrow \mathbf{R}^n$$
- G_n : class of graphs of size n
- **LAR vector w** : the output $A(G)$ of an algorithm A on a graph G
 - w_i : the authority weight of node i



Popular LAR algorithms

- InDegree algorithm
 - $w_i = \text{in-degree}(i)$
- PageRank algorithm [BP98]
 - perform a random walk on G with random resets (with probability $1-\alpha$)
 - w = stationary distribution of the random walk
- HITS algorithm [K98]
 - compute the left (hub) and right (authority) singular vectors of the adjacency matrix W
 - w = right singular vector

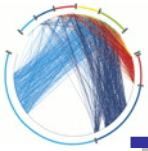


Properties of Interest

- Stability
 - small changes in the graph should cause small changes in the output of the algorithm
- Similarity
 - the output of two algorithms are close

Under what conditions (for which classes of graphs) is an algorithm stable, or are two algorithms similar?

- Axiomatic characterizations



Distance between LAR vectors

- Geometric distance: how close are the **numerical weights** of vectors w_1, w_2 ?

$$d_2(w_1, w_2) = \sqrt{\sum |w_1[i] - w_2[i]|^2}$$

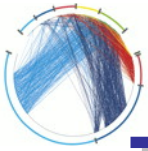
- Assumption: Weights are normalized under norm L_2
 - normalization makes a difference



Distance between LAR vectors

- Rank distance: how close are the **ordinal rankings** induced by the vectors w_1, w_2 ?
 - Kendal's τ distance

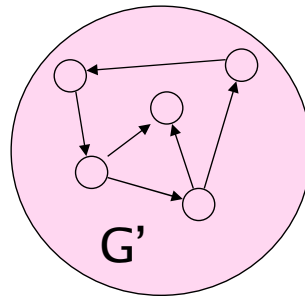
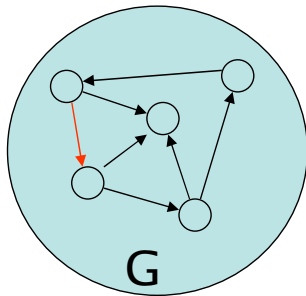
$$d_r(w_1, w_2) = \frac{\text{pairs ranked in a different order}}{\text{total number of distinct pairs}}$$



Stability: graph distance

- Definition: **Link distance** between graphs $G=(P,E)$ and $G'=(P,E')$

$$d_\ell(G,G') = |E \cup E'| - |E \cap E'|$$



$$d_\ell(G,G') = 2$$



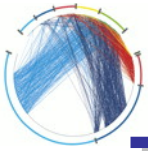
Stability

- $C_k(G)$: set of graphs G' such that $d_\ell(G,G') \leq k$
- Definition: Algorithm **A** is **stable** if for any fixed k

$$\max_{G \in G_n} \max_{G' \in C_k(G)} d_2(A(G), A(G')) = o(1)$$

- Definition: Algorithm **A** is **rank stable** if for any fixed k

$$\max_G \max_{G' \in C_k(G)} d_r(A(G), A(G')) = o(1)$$



Stability: Results

- InDegree is (rank) stable on G_n [BRRT05]
- HITS, PageRank, are (rank) unstable on G_n

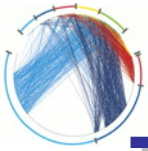


Perturbations of PageRank

- Perturbations to unimportant nodes have small effect on the PageRank values [NZJ01][BGS03]

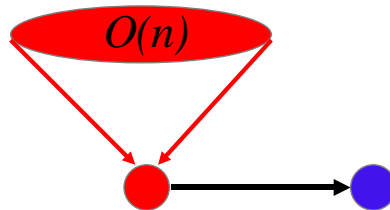
$$d_1(A(G), A(G')) \leq \frac{2\alpha}{1-2\alpha} \sum_{i \in P} A(G)[i]$$

- Lee and Borodin 2003: PageRank is stable
 - HITS remains unstable



Instability of PageRank

- PageRank is unstable



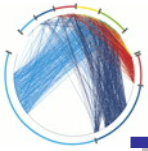
- PageRank is rank unstable [Lempel Moran 2003]
- Open question: Can we derive conditions for the stability of PageRank in the general case?



Singular Value Decomposition

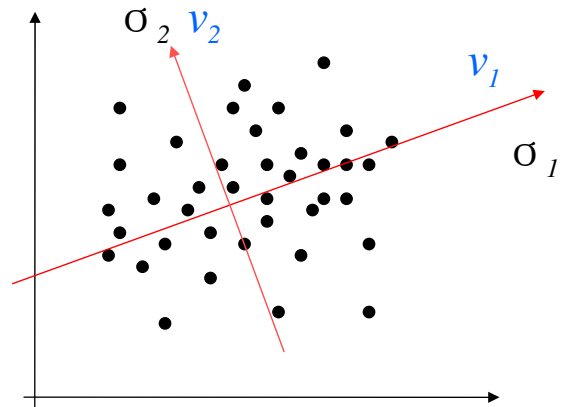
$$A = U \Sigma V^T = \begin{matrix} \vec{u}_1 & \vec{u}_2 & \dots & \vec{u}_r \\ [n \times r] & [r \times r] & [r \times n] \end{matrix} \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_r \end{bmatrix} \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \\ \vdots \\ \vec{v}_r \end{bmatrix}$$

- r : rank of matrix A
- $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$: singular values (square roots of eig-val AA^T , $A^T A$)
- u_1, u_2, \dots, u_r : left singular vectors (eig-vectors of AA^T)
- v_1, v_2, \dots, v_r : right singular vectors (eig-vectors of $A^T A$)
- $A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \dots + \sigma_r u_r v_r^T$



Singular Value Decomposition

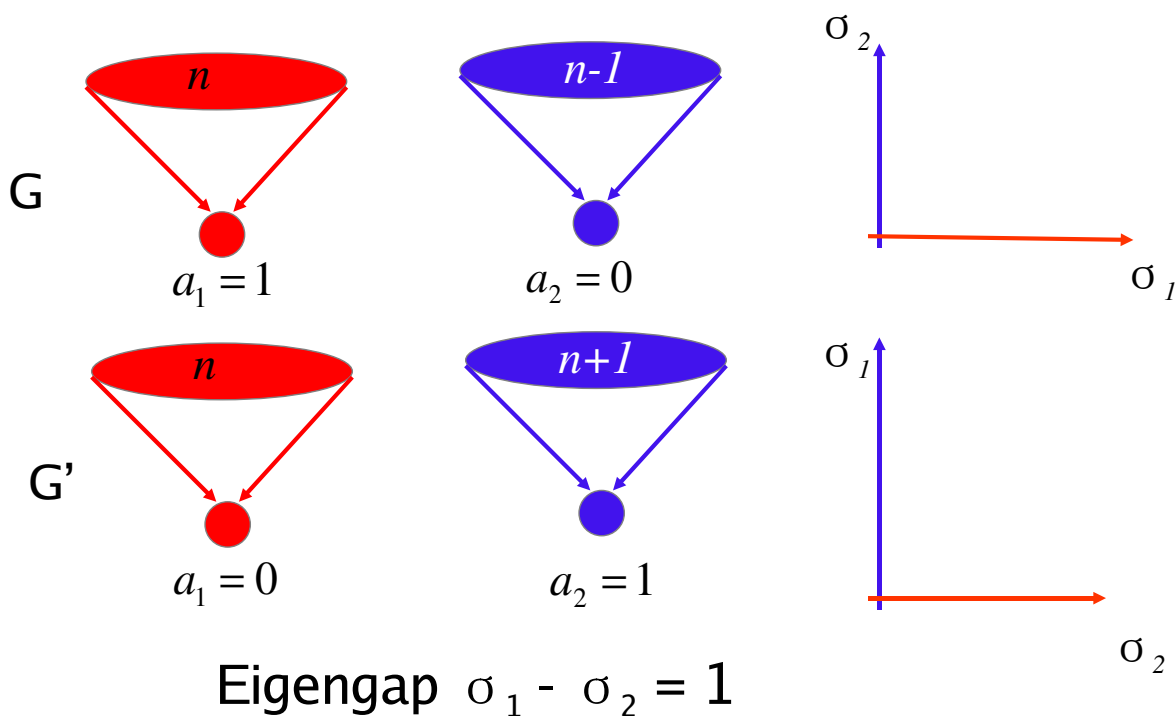
- **Linear trend \mathbf{v}** in matrix \mathbf{A} :
 - the tendency of the row vectors of \mathbf{A} to align with vector \mathbf{v}
 - strength of the linear trend: $\mathbf{A}\mathbf{v}$
- SVD discovers the linear trends in the data
- $\mathbf{u}_i\mathbf{v}_i^T$: the i -th strongest linear trend
- σ_i : the strength of the i -th strongest linear trend

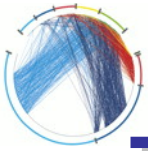


- HITS ranks according to the **strongest linear trend \mathbf{v}_i** in the authority space



Instability of HITS





Stability of HITS

- **Theorem:** HITS is stable if

$$\sigma_1(W) - \sigma_2(W) = \omega(1)$$

- The two strongest linear trends are well separated

- [Ng, Zheng, Jordan 2001]: HITS is stable if

$$\alpha_1^2 - \alpha_2^2 = \omega(\sqrt{d_{\text{out}}})$$



Similarity

- Definition: Two algorithms A_1, A_2 are **similar** if

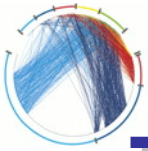
$$\max_{G \in G_n} d_2(A_1(G), A_2(G)) = o(1)$$

- Definition: Two algorithms A_1, A_2 are **rank similar** if

$$\max_{G \in G_n} d_r(A_1(G), A_2(G)) = o(1)$$

- Definition: Two algorithms A_1, A_2 are **rank equivalent** if

$$\max_{G \in G_n} d_r(A_1(G), A_2(G)) = 0$$



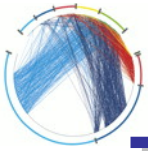
Similarity: Results

- No pairwise combination of InDegree, HITS, PageRank algorithms is similar, or rank similar on the class of all possible graphs G_n
- Can we get better results if we restrict ourselves to smaller classes of graphs?
 - We focus on similarity of HITS and InDegree algorithms [DLT05]



Product Graphs

- Latent authority and hub vectors a, h
 - h_i = probability of node i being a good hub
 - a_j = probability of node j being a good authority
- Generate a link $i \rightarrow j$ with probability $h_i a_j$
$$W[i, j] = \begin{cases} 1 & \text{with probability } h_i a_j \\ 0 & \text{with probability } 1 - h_i a_j \end{cases}$$
 - Azar, Fiat, Karlin, McSherry, Saia 2001, Michail, Papadimitriou 2002, Chung, Lu, Vu 2002
- The class of product graphs G_n^p
 - (a.k.a. “graphs with given expected degree sequences”)



Product Graphs

$$W = M + R$$

- **M**: rank-1 matrix $h a^T$

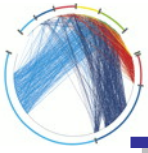
$$M = h a^T = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix} \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} = \begin{bmatrix} h_1 a_1 & h_1 a_2 & \cdots & h_1 a_n \\ h_2 a_1 & h_2 a_2 & \cdots & h_2 a_n \\ \vdots & \vdots & \ddots & \vdots \\ h_n a_1 & h_n a_2 & \cdots & h_n a_n \end{bmatrix}$$

- **R**: rounding matrix $R_{[i,j]} = \begin{cases} 1 - h_i a_j & \text{with probability } h_i a_j \\ -h_i a_j & \text{with probability } 1 - h_i a_j \end{cases}$



Product Graphs

- Idea[AFK+01]: View the product graph $W=M+R$ as a perturbation of the rank-1 matrix **M** by the matrix **R**
- HITS and InDegree are identical on rank-1 matrix **M**
- How do the outputs change after perturbing **M** by **R** ?



HITS and InDegree on Product Graphs

- **Theorem:** HITS and InDegree are similar with high probability on the class of product graphs, G_n^p subject to **some assumptions**

Assumption 1: $\alpha_1(M) = \|\mathbf{h}\|_2 \|\mathbf{a}\|_2 = \omega(\sqrt{n})$

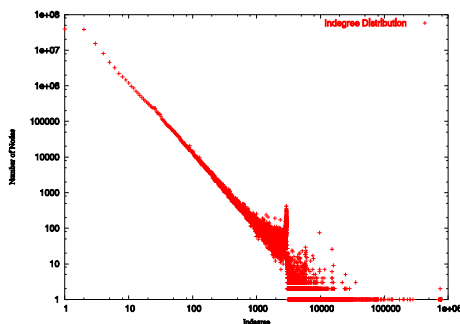
Assumption 2: Let $H = \sum h_i$ then $H\|\mathbf{a}\|_2 = \omega(\sqrt{n \log n})$

- **Assumptions 1 and 2** are general enough to include graphs with (expected) degrees that follow power law distribution with $\alpha > 3$

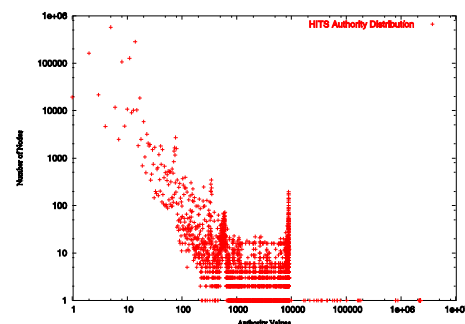


Experiments with real web graphs

- Dataset: The Stanford WebBase project
- Correlation coefficient between authority and in-degree vector: **0.93**

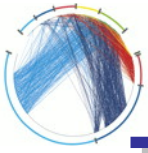


in-degree distribution



HITS authority values distribution

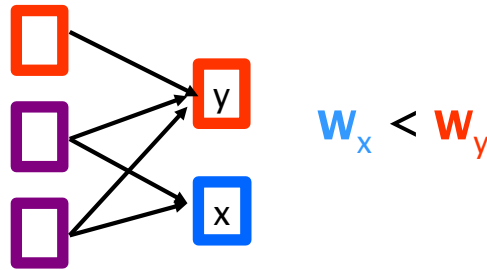
- Correlation coefficient between hub and out-degree vectors: **0.05**



Monotonicity

- Monotonicity: Algorithm A is **strictly monotone** if for any nodes x and y

$$B_N(x) \subset B_N(y) \Leftrightarrow A(G)[x] < A(G)[y]$$

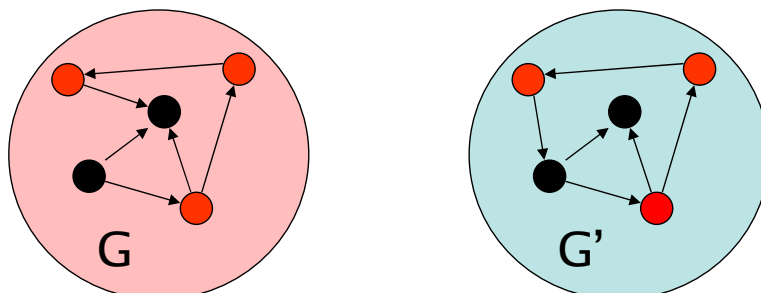


Locality

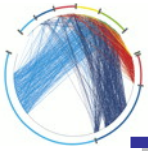
- Locality: An algorithm A is **strictly rank local** if, for every pair of graphs $G=(P,E)$ and $G'=(P,E')$, and for every pair of nodes x and y , if $B_G(x)=B_{G'}(x)$ and $B_G(y)=B_{G'}(y)$ then

$$A(G)[x] < A(G)[y] \Leftrightarrow A(G')[x] < A(G')[y]$$

- the relative order of the nodes remains the same if their back links are not affected



- The InDegree algorithm is strictly rank local



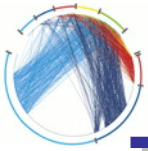
Label Independence

- Label Independence: An algorithm is **label independent** if a permutation of the labels of the nodes yields the same permutation of the weights
 - the weights assigned by the algorithm do not depend on the labels of the nodes



Axiomatic characterization of the InDegree algorithm

- Theorem: Any algorithm that is **strictly rank local**, **strictly monotone** and **label independent** is **rank equivalent** to the InDegree algorithm
- All three properties are needed



Other work

- An axiomatic characterization of PageRank algorithm
 - “Ranking Systems: The PageRank axioms”
Alon Altman, Moshe Tenneholtz, ACM
Conference on Electronic Commerce, 2005



Open questions

- What is the **necessary** condition for the stability of the HITS algorithm?
 - can the results of [NZJ01] be proven for 0/1 matrices?
- Can we say anything about other LAR algorithms on product graphs?
 - e.g. PageRank
- Can we prove anything when we consider **rank distance**?
- Can we define other properties?
 - e.g., is **spam sensitivity** different from stability?

Thank you!

