Authoritative Sources in a Hyperlinked Enviroment

Dimitris Sakavalas

June 15, 2010

Outline

www-search

- Discover pages relevant to a given query string
- Discover the most authoritative pages
  (subjective, requires human evaluation)

www-search

- Discover pages relevant to a given query string
- Discover the most authoritative pages
  (subjective, requires human evaluation)

Queries

- Specific queries $\longrightarrow$ *Scarcity Problem*
- Broad-topic queries $\longrightarrow$ *Abundance Problem*
- Similar-page queries

In search of a definition of authority

In search of a definition of authority

- No endogenous measure of a page to assess authority (ex. *Text-based* ranking functions for queries: “Harvard”, “search engines”, “automobile manufacturers”, images)

In search of a definition of authority

- No endogenous measure of a page to assess authority (ex. *Text-based* ranking functions for queries: "Harvard", "search engines", "automobile manufacturers", images)
- Solution: *Analysis of the link structure*

In search of a definition of authority

- No endogenous measure of a page to assess authority (ex. *Text-based* ranking functions for queries: "Harvard", "search engines", "automobile manufacturers", images)
- Solution: *Analysis of the link structure*

Pitfalls:

- Advertising links
- Navigational links
- Universal Popularity links

In search of a definition of authority

- No endogenous measure of a page to assess authority (ex. *Text-based* ranking functions for queries: "Harvard", "search engines", "automobile manufacturers", images)
- Solution: *Analysis of the link structure*

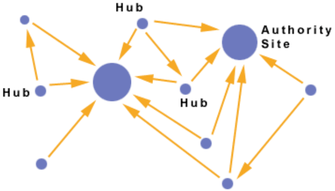
Pitfalls:

- Advertising links
- Navigational links
- Universal Popularity links

Question. How reliable is page p?

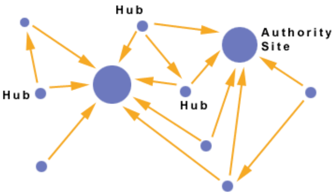
Hubs

Pages that link to many Authorities



Hubs

Pages that link to many Authorities



Outline of the algorithm

1. Create a focused subgraph of the www from the output of a text-based engine (Small collection of pages likely to contain the most authoritative pages)
2. Identify *hubs* and *authorities*

Construction of a Focused Subgraph of the www

Directed graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ of the www.

V the collection of hyperlinked pages of the www

- V: (nodes correspond to the pages)
- $(p, q) \in E \Leftrightarrow$ There is a link from p to q

$\mathbf{G}[\mathbf{W}]$. Subgraph of G induced on $W \subseteq V$

Construction of a Focused Subgraph of the www

Directed graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ of the www.
V the collection of hyperlinked pages of the www

- V: (nodes correspond to the pages)
- $(p, q) \in E \Leftrightarrow$ There is a link from p to q

G[W]. Subgraph of G induced on $W \subseteq V$

Goal

Construct $G_\sigma = G[S_\sigma]$ such that

1. $|S_\sigma|$ is relatively small $\longrightarrow$ (computational cost)
2. S_σ rich in relevant pages $\longrightarrow$ (search quality)
3. S_σ contains most of the *strongest authorities*

Construction of a Focused Subgraph of the www

Directed graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ of the www.
V the collection of hyperlinked pages of the www

- V: (nodes correspond to the pages)
- $(p, q) \in E \Leftrightarrow$ There is a link from p to q

G[W]. Subgraph of G induced on $W \subseteq V$

Goal

Construct $G_\sigma = G[S_\sigma]$ such that

1. $|S_\sigma|$ is relatively small $\rightarrow$ (computational cost)
2. S_σ rich in relevant pages $\rightarrow$ (search quality)
3. S_σ contains most of the *strongest authorities*

Observation

S_σ *rich in relevant pages* $\rightarrow$ *many links to authorities*

Construction of a Focused Subgraph of the www

Notation

- $\Gamma^{+}(p)$: the set of all pages p points to
- $\Gamma^{-}(p)$: the set of all pages pointing to p
- $\mathcal{E}$: a *text-based search engine*

Construction of a Focused Subgraph of the www

Notation

- $\Gamma^{+}(p)$: the set of all pages p points to
- $\Gamma^{-}(p)$: the set of all pages pointing to p
- $\mathcal{E}$: a *text-based search engine*

1st Step

Root set $R_\sigma = \{t$ highest ranked pages for the query σ from $\mathcal{E}\}$

- R_σ satisfies $(i), (ii)$.
- Generally doesn't satisfy (iii) and is *structureless* (few links)

Notation

- $\Gamma^+(p)$: the set of all pages p points to
- $\Gamma^-(p)$: the set of all pages pointing to p
- $\mathcal{E}$: a *text-based search engine*

1st Step

Root set $R_\sigma = \{t$ highest ranked pages for the query σ from $\mathcal{E}\}$

- R_σ satisfies $(i), (ii)$.
- Generally doesn't satisfy (iii) and is *structureless* (few links)

Observation

If $a \notin R_\sigma$, strong authority, there is a high probability that $\exists p \in R_\sigma$ such that $a \in \Gamma^+(p)$

2nd Step

Idea:Expand R_σ into the Base set S_σ by adding strong authorities and relevant pages

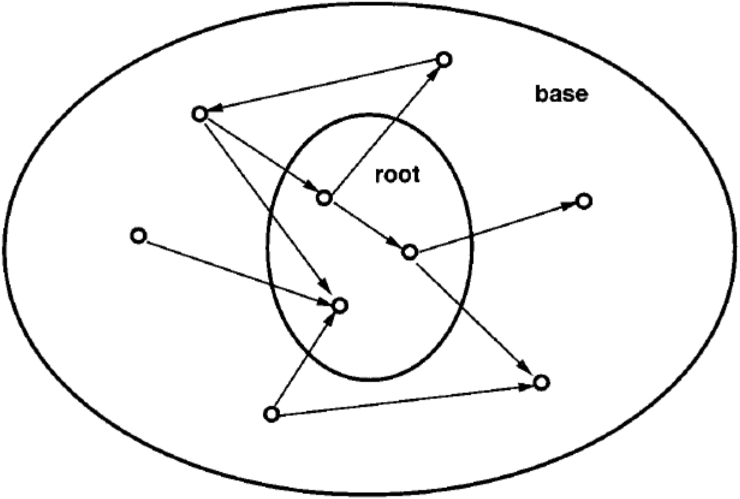
2nd Step

Idea:Expand R_σ into the Base set S_σ by adding strong authorities and relevant pages

Subgraph$(\sigma, \mathcal{E}, t, d)$

$t, d \in \mathbb{N}$
Set $S_\sigma := R_\sigma$
For each page $p \in R_\sigma$
  Add all pages in $\Gamma^+(p)$ to S_σ
  If $|\Gamma^-(p)| \leq d$ then,
    Add all pages in $\Gamma^-(p)$ to S_σ
  Else
    Add an arbitary set of d pages from $\Gamma^-(p)$ to S_σ
End.

Expanding R_σ into the base set S_σ



Heuristics impoving S_σ

Links

- $Transverse$: link between pages with different domain name
- $Intrinsic$: link between pages with the same domain name (navigational)

Heuristics impoving S_σ

Links

- $Transverse$: link between pages with different domain name
- $Intrinsic$: link between pages with the same domain name (navigational)

3rd Step:Heuristics

1. Delete all $Intrinsic$ links from the graph G_σ
2. Allow up to a small number m pages from a single domain to point to any given page p

Heuristics impoving S_σ

Links

- *Transverse*: link between pages with different domain name
- *Intrinsic*: link between pages with the same domain name (navigational)

3rd Step:Heuristics

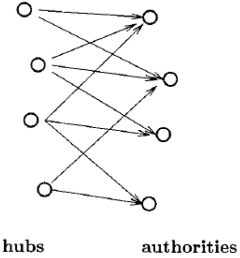
1. Delete all *Intrinsic* links from the graph G_σ
2. Allow up to a small number m pages from a single domain to point to any given page p

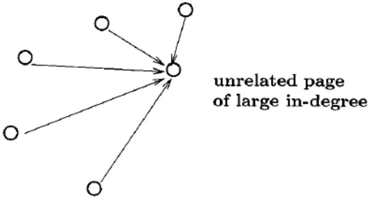
Finally we obtain a small subgraph G_σ , relatively focused on the query σ, containing many relevant pages and strong authorities.

Purely In-degree ordering of Authorities

Purely In-degree ordering of Authorities

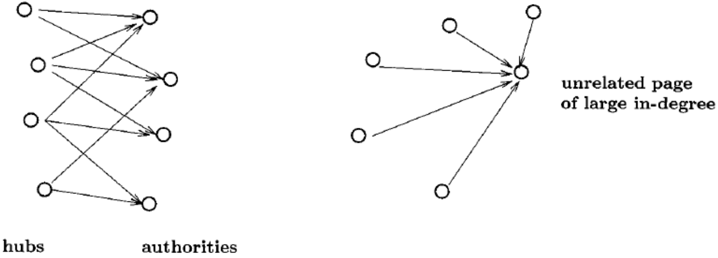
Problem. "Universally popular" pages, large in-degree regardless of the underlying query topic (ex.Amazon Books)





Purely In-degree ordering of Authorities

Problem. “Universally popular” pages, large in-degree regardless of the underlying query topic (ex.Amazon Books)



Observation

Authorities: *high in-degree, considerable overlap in the sets of pages that point to them(hubs)*

Observation (mutually reinforcing relationship)

- **Good Hub**: *a page that points to many good authorities*
- **Good Authority**: *a page that is pointed by many good hubs*

Observation (mutually reinforcing relationship)

- **Good Hub**: *a page that points to many good authorities*
- **Good Authority**: *a page that is pointed by many good hubs*

Definitions

$\forall p \in S_\sigma = \{1, 2, ..., n\}$ assign:

- authority weight $x^{<p>} \geq 0$
- hub weight $y^{<p>} \geq 0$

Observation (mutually reinforcing relationship)

- **Good Hub**: *a page that points to many good authorities*
- **Good Authority**: *a page that is pointed by many good hubs*

Definitions

$\forall p \in S_\sigma = \{1, 2, ..., n\}$ assign:

- authority weight $x^{<p>} \geq 0$
- hub weight $y^{<p>} \geq 0$
- corresponding vectors $x = \begin{bmatrix} x^{<1>} \\ x^{<2>} \\ \vdots \\ x^{<n>} \end{bmatrix}$ and $y = \begin{bmatrix} y^{<1>} \\ y^{<2>} \\ \vdots \\ y^{<n>} \end{bmatrix}$

  Normalized such that $\|x\| = \|y\| = 1$

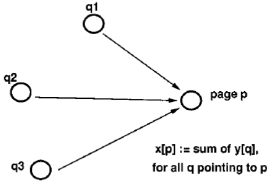
  example. For norm $\| \cdot \|_1$, $\sum_{i=1}^{n} x^{<i>} = \sum_{i=1}^{n} y^{<i>} = 1$

Hubs and Authorities

Definitions

- $\mathcal{I}(x, y)$ (x-weight update):

$$\forall x^{<p>}, \quad x^{<p>} \leftarrow \sum_{q:(q,p)\in E} y^{<q>}$$

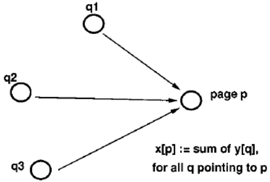


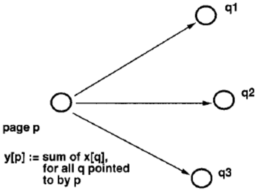
- $\mathcal{I}(x, y)$ (x-weight update):

$$\forall x^{<p>}, \;\; x^{<p>} \leftarrow \sum_{q:(q,p)\in E} y^{<q>}$$

- $\mathcal{O}(x, y)$ (y-weight update):

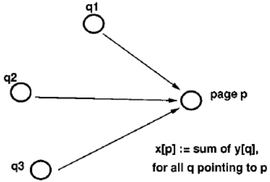
$$\forall y^{<p>}, \;\; y^{<p>} \leftarrow \sum_{q:(p,q)\in E} x^{<q>},$$





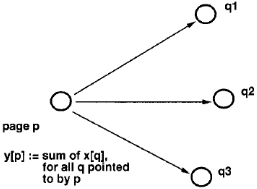
- $\color{blue}{\mathcal{I}(x,y)}$ (x-weight update):

$$\forall x^{<p>},\quad x^{<p>} \leftarrow \sum_{q:(q,p)\in E} y^{<q>}$$



- $\color{blue}{\mathcal{O}(x,y)}$ (y-weight update):

$$\forall y^{<p>},\quad y^{<p>} \leftarrow \sum_{q:(p,q)\in E} x^{<q>},$$



Idea

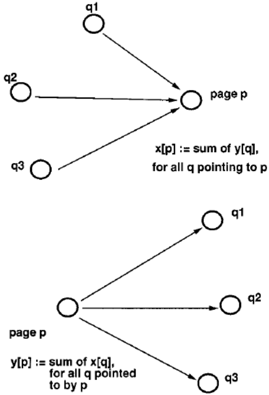
Find the disired *"equilibrium"* values for the weights by applying $\mathcal{I}, \mathcal{O}$ operations iterativelly.

- $\mathcal{I}(x,y)$ (x-weight update):

$$\forall x^{<p>},\ \ x^{<p>} \leftarrow \sum_{q:(q,p)\in E} y^{<q>}$$

- $\mathcal{O}(x,y)$ (y-weight update):

$$\forall y^{<p>},\ \ y^{<p>} \leftarrow \sum_{q:(p,q)\in E} x^{<q>},$$



Idea

Find the disired *"equilibrium"* values for the weights by applying $\mathcal{I},\mathcal{O}$ operations iterativelly.

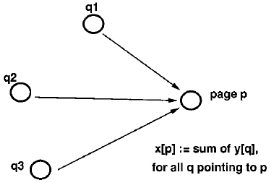
Question. Does this *"equilibrium"* exist?

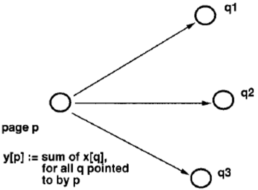
- $\mathcal{I}(x,y)$ (x-weight update):

$$\forall x^{<p>}, \quad x^{<p>} \leftarrow \sum_{q:(q,p)\in E} y^{<q>}$$

- $\mathcal{O}(x,y)$ (y-weight update):

$$\forall y^{<p>}, \quad y^{<p>} \leftarrow \sum_{q:(p,q)\in E} x^{<q>},$$





Idea

Find the disired *"equilibrium"* values for the weights by applying $\mathcal{I}, \mathcal{O}$ operations iterativelly.

Question. Does this *"equilibrium"* exist? $\rightarrow$YES (linear algebra)

Iterative algorithm

- Iterate(G_σ, k)

$z := (1, 1, ..., 1) \in \mathbb{R}^n$
Set $x_0 := z$
Set $y_0 := z$
For $i = 1, 2, ..., k$
  Apply the $\mathcal{I}(x_{i-1}, y_{i-1})$ operation to obtain new $x-$weights x_i'
  Apply the $\mathcal{O}(x_i', y_{i-1})$ operation to obtain new $y-$weights y_i'
  Normalize x_i', obtaining x_i
  Normalize y_i', obtaining y_i
End
Return (x_k, y_k)

Iterative algorithm

- Iterate(G_σ, k)

$z := (1, 1, ..., 1) \in \mathbb{R}^n$
Set $x_0 := z$
Set $y_0 := z$
For $i = 1, 2, ..., k$
  Apply the $\mathcal{I}(x_{i-1}, y_{i-1})$ operation to obtain new $x-$weights x_i'
  Apply the $\mathcal{O}(x_i', y_{i-1})$ operation to obtain new $y-$weights y_i'
  Normalize x_i', obtaining x_i
  Normalize y_i', obtaining y_i
End
Return (x_k, y_k)

- Filter(G_σ, k, c)

(x_k, y_k) :=Iterate(G_σ, k)
Report the pages with the c largest coordinates in x_k as authorities
Report the pages with the c largest coordinates in y_k as hubs

Notation

Matrix $M \in \mathbb{C}^{n \times n}$

- $Eigenvalues\ \lambda_i(M)$
- $Eigenvectors\ \omega_i(M)$
- $eigenspace\ V_\lambda$ subspace of $\mathbb{C}^n$
- $multiplicity\ of\ \lambda$: $m_\lambda = dim(V_\lambda)$

Elements of Linear Algebra

Notation

Matrix $M \in \mathbb{C}^{n \times n}$

- $Eigenvalues\ \lambda_i(M)$
- $Eigenvectors\ \omega_i(M)$
- $eigenspace\ V_\lambda$ subspace of $\mathbb{C}^n$
- $multiplicity\ of\ \lambda$: $m_\lambda = dim(V_\lambda)$

Definitions

- $Positive\ matrix\ M$: If $M_{i,j} > 0, \forall i, j$
- $Primitive\ matrix\ M$: If $\exists m \in \mathbb{N}, M^m$: positive

Elements of Linear Algebra

Let $M \in \mathbb{R}^{n\times n}$ *symmetric, nonnegative matrix* $(M_{i,j} \geq 0, \forall i, j)$

Theorems

1. M has at most n distinct eigenvalues and $\sum_{m_{\lambda_i}} = n$

Let $M \in \mathbb{R}^{n \times n}$ *symmetric, nonnegative matrix* $(M_{i,j} \geq 0, \forall i, j)$

Theorems

1. M has at most n distinct eigenvalues and $\sum_{m_{\lambda_i}} = n$
2. M has *only real eigenvalues* λ_i
   Denote them $|\lambda_1(M)| \geq |\lambda_2(M)| \geq ... \geq |\lambda_n(M)|$

Let $M \in \mathbb{R}^{n \times n}$ *symmetric, nonnegative matrix* $(M_{i,j} \geq 0, \forall i, j)$

Theorems

1. M has at most n distinct eigenvalues and $\sum_{m_{\lambda_i}} = n$
2. M has *only real eigenvalues* λ_i
   Denote them $|\lambda_1(M)| \geq |\lambda_2(M)| \geq ... \geq |\lambda_n(M)|$
3. $(Perron\text{-}Frobenius)$ If M is primitive then:
   i. Largest eigenvalue $\lambda_1(M) > 0$ and $m_\lambda = 1$
   ii. $\lambda_1 > |\lambda_i| \forall i \neq 1$
   iii. λ_1 has a corresponding eigenvector $\omega_1(M)$ with all entries positive (the principal eigenvector)
   Consider $\omega_1(M)$ normalized, $\|\omega_1(M)\| = 1$

Let $M \in \mathbb{R}^{n \times n}$ *symmetric, nonnegative matrix* $(M_{i,j} \geq 0, \forall i, j)$

Theorems

1. M has at most n distinct eigenvalues and $\sum_{m_{\lambda_i}} = n$
2. M has *only real eigenvalues* λ_i
   Denote them $|\lambda_1(M)| \geq |\lambda_2(M)| \geq ... \geq |\lambda_n(M)|$
3. (*Perron-Frobenius*) If M is primitive then:
   i. Largest eigenvalue $\lambda_1(M) > 0$ and $m_\lambda = 1$
   ii. $\lambda_1 > |\lambda_i| \forall i \neq 1$
   iii. λ_1 has a corresponding eigenvector $\omega_1(M)$ with all entries positive (the principal eigenvector)
   Consider $\omega_1(M)$ normalized, $\|\omega_1(M)\| = 1$
4. If M is primitive and vector v not orthogonal to the principal eigenvector $\omega_1(M)$, let v_k be the unit vector in the direction of $M^k v$. Then $v_k \xrightarrow{k \to \infty} \omega_1(M)$

Observation

Note that we can consider $\omega_1(M)$ *as a vector of the orthonormal base of* V_{λ_1}. *Moreover* $dim V_{\lambda_1} = m_{\lambda_1} = 1$ *and all entries in* $\omega_1(M)$ *are positive. Hence the principal eigenvector* $\omega_1(M)$ *is unique.*

Connection of Iterative algorithm to linear Algebra

Let A be the adjacency matrix of G_σ

Observation

Operations $\mathcal{I}(x, y)$ *and* $\mathcal{O}(x, y)$ *can be written as*

$$x \leftarrow A^\top y \text{ and } y \leftarrow Ax \text{ respectivelly}$$

Connection of Iterative algorithm to linear Algebra

Let A be the adjacency matrix of G_σ

Observation

Operations $\mathcal{I}(x, y)$ *and* $\mathcal{O}(x, y)$ *can be written as*

$$x \leftarrow A^\top y \text{ and } y \leftarrow Ax \text{ respectivelly}$$

Iterate Algorithm

The algorithm initializes $x_0 = y_0 = z = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$. In the and $i - th$ step of thalgorithm we have the following steps:

- $\mathcal{I}(x_i, y_i - 1) \dashrightarrow x_i = A^\top y_{i-1}$
- $\mathcal{O}(x_i, y_i) \dashrightarrow y_i = Ax_i$

Connection of Iterative algorithm to linear Algebra

Let A be the adjacency matrix of G_σ

Observation

Operations $\mathcal{I}(x, y)$ *and* $\mathcal{O}(x, y)$ *can be written as*

$$x \leftarrow A^\top y \text{ and } y \leftarrow Ax \text{ respectivelly}$$

Iterate Algorithm

The algorithm initializes $x_0 = y_0 = z = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$. In the and $i - th$ step of thalgorithm we have the following steps:

- $\mathcal{I}(x_i, y_i - 1) \dashrightarrow x_i = A^\top y_{i-1}$
- $\mathcal{O}(x_i, y_i) \dashrightarrow y_i = Ax_i$

$$\Rightarrow \left.\begin{array}{l} x_i = A^\top A x_{i-1} = (A^\top A)^i \cdot x_0 \\ y_i = AA^\top y_{i-1} = (AA^\top)^i \cdot y_0 \end{array}\right\} \Rightarrow \begin{array}{l} x_i = (A^\top A)^i \cdot z \\ y_i = (AA^\top)^i \cdot z \end{array} \quad (*)$$

Convergence of the Iterative Algorithm

Observation

- *We consider* x_i, y_i *normalized in each step as in Iterate algorithm. So in every step* $\|x_i\| = \|y_i\| = 1$
- *For simplicity, matrices* $AA^\top$ *and* $A^\top A$ *are considered primitive*
- *Obviously* $z \cdot c \neq 0, \forall c \in \mathbb{R}^n_{\geq 0}$

Convergence of the Iterative Algorithm

Observation

- *We consider* x_i, y_i *normalized in each step as in Iterate algorithm. So in every step* $\|x_i\| = \|y_i\| = 1$
- *For simplicity, matrices* $AA^\top$ *and* $A^\top A$ *are considered primitive*
- *Obviously* $z \cdot c \neq 0, \forall c \in \mathbb{R}^n_{\geq 0}$

Theorem

The sequences $\{x_i\}$ *and* $\{y_i\}$ *produced by Iterate Algorithm converge to limits* x^*, y^* *respectively*

Observation

- *We consider* x_i, y_i *normalized in each step as in Iterate algorithm. So in every step* $\|x_i\| = \|y_i\| = 1$
- *For simplicity, matrices* $AA^\top$ *and* $A^\top A$ *are considered primitive*
- *Obviously* $z \cdot c \neq 0, \forall c \in \mathbb{R}^n_{\geq 0}$

Theorem

The sequences $\{x_i\}$ *and* $\{y_i\}$ *produced by Iterate Algorithm converge to limits* x^*, y^* *respectively*

Proof.

$A^\top A$ primitive, z not orthogonal to the principal eigenvector $\omega_1(A^\top A) = x^*$, x_i the unit vector in the direction $(A^\top A)^i \cdot z$ Hence from *Theorem 4* $\Rightarrow \{x_i\} \rightarrow x^*$, simirarly $\{y_i\} \rightarrow y^* = \omega_1(AA^\top)$

□

Conclusion

The process of iterated $\mathcal{I}/\mathcal{O}$ operations will converge. The desired equilibrium values for the x, y weights, are the values x^*, y^*.

Conclusion

The process of iterated $\mathcal{I}/\mathcal{O}$ operations will converge. The desired equilibrium values for the x, y weights, are the values x^*, y^*.

Observation

In practice the convergence of $Iterate$ is quite rapid, so one can compute x, y weights by starting from any initial vectors x_0, y_0, and performing a fixed number of $\mathcal{I}, \mathcal{O}$ operations

Conclusion

The process of iterated $\mathcal{I}/\mathcal{O}$ operations will converge. The desired equilibrium values for the x, y weights, are the values x^*, y^*.

Observation

In practice the convergence of $Iterate$ is quite rapid, so one can compute x, y weights by starting from any initial vectors x_0, y_0, and performing a fixed number of $\mathcal{I}, \mathcal{O}$ operations

Hyperlink-Induced Topic Search (HITS) algorithm (Ask.com)

Similar-Page Queries

Query. Find pages similar to p

Idea

- Initiate search with the page p instead of a query string σ
- Use the link structure to infer "similarity among pages"
- Similar pages to p $\rightarrow$ Strongest authorities in local region of p

Similar-Page Queries

Query. Find pages similar to p

Idea

- Initiate search with the page p instead of a query string σ
- Use the link structure to infer "similarity among pages"
- Similar pages to $p \rightarrow$ Strongest authorities in local region of p

Adaptation of the broad-topic query method

1. Find t pages pointing to $p \longrightarrow$ Assemble *root set* R_p
2. Expand R_p to *base set* S_p as before
3. Search for authorities and hubs in the focused subgraph G_p

Multiple Sets of Hubs and Authorities

Situation. For a query string σ, relevant pages grouped into $clusters$. Reasons

- Different meanings of σ. ex. "`jaguar`"
- String σ arises as term in multiple technical communities. ex. "`randomized algorithms`"
- String σ refers to a highly polarized issue. ex."`abortion`"

Situation. For a query string σ, relevant pages grouped into $clusters$. Reasons

- Different meanings of σ. ex. "`jaguar`"
- String σ arises as term in multiple technical communities. ex. "`randomized algorithms`"
- String σ refers to a highly polarized issue. ex."`abortion`"

Problem. Identify the $clusters$

Multiple Sets of Hubs and Authorities

Situation. For a query string σ, relevant pages grouped into $clusters$. Reasons

- Different meanings of σ. ex. "`jaguar`"
- String σ arises as term in multiple technical communities. ex. "`randomized algorithms`"
- String σ refers to a highly polarized issue. ex."`abortion`"

Problem. Identify the $clusters$

Idea

- Clusters represented in focused subgraph G_σ as densely linked subgraphs(collection of hubs and authorities)
- Extract densely linked collections of hubs and authorities through *non-principal* eigenvectors of $AA^\top$ and $A^\top A$

Non-Principal Eigenvectors and Clustering of G_σ

Proposition

$AA^\top$ and $A^\top A$ have the same multiset of eigenvalues, and their eigenvectors can be chosen so that $\omega_i(AA^\top) = A\omega_i(A^\top A)$

Non-Principal Eigenvectors and Clustering of G_σ

Proposition

$AA^\top$ and $A^\top A$ have the same multiset of eigenvalues, and their eigenvectors can be chosen so that $\omega_i(AA^\top) = A\omega_i(A^\top A)$

Observation

- *For each pair of eigenvectors* $x_i^* = \omega_i(A^\top A), y_i^* = \omega_i(AA^\top)$ *operation* $\mathcal{I}(x_i^*, y_i^*)$ *keeps* $x-$*weights parallel to* x_i^**:*
  $x = A^\top \omega_i(AA^\top) = A^\top A \omega_i(A^\top A) = \lambda_i x_i^*$
  Simirarly operation $\mathcal{O}(x_i^*, y_i^*)$ *keeps* $x-$*weights parallel to* y_i^*

Non-Principal Eigenvectors and Clustering of G_σ

Proposition

$AA^\top$ and $A^\top A$ have the same multiset of eigenvalues, and their eigenvectors can be chosen so that $\omega_i(AA^\top) = A\omega_i(A^\top A)$

Observation

- *For each pair of eigenvectors $x_i^* = \omega_i(A^\top A), y_i^* = \omega_i(AA^\top)$ operation $\mathcal{I}(x_i^*, y_i^*)$ keeps $x-$weights parallel to x_i^*:*
  $x = A^\top \omega_i(AA^\top) = A^\top A\omega_i(A^\top A) = \lambda_i x_i^*$
  Simirarly operation $\mathcal{O}(x_i^*, y_i^*)$ keeps $x-$weights parallel to y_i^*
- *Non-principal eigenvectors have both positive and negative entries. Hence each pair of (x_i^*, y_i^*) provide us with two densely connected set of hubs and authorities*
  - *Pages that correspond to the c coordinates with the most positive values*
  - *Pages that correspond to the c coordinates with the most negative values*

Non-Principal Eigenvectors and Clustering of G_σ

Description of the method

- For each of the fist few non-principal eigenvectors (x_i^*, y_i^*) find the two densely connected set of hubs and authorities through an algorithm similar(but less clean conceptually) to the $Iterate$ $algorithm$

Description of the method

- For each of the fist few non-principal eigenvectors (x_i^*, y_i^*) find the two densely connected set of hubs and authorities through an algorithm similar(but less clean conceptually) to the $Iterate$ $algorithm$
- The pages with large coordinates in the first few nonprincipal eigenvectors tend to recur, so that essentially the same collection of hubs and authorities will often be generated by several of the strongest nonprincipal eigenvectors

Similar Concepts in Other Areas

- $Social\ networks \longrightarrow$ *"Standing"*
- $Bibliometrics \longrightarrow$ *"Impact"*

Similar Concepts in Other Areas

- *Social networks* $\longrightarrow$ *"Standing"*
- *Bibliometrics* $\longrightarrow$ *"Impact"*
- *Economics*
  (Wassily Leontief 1941-Nobel prize 1973)