Random Walks on Graphs - Part I

Pawan Goyal

CSE, IITKGP

October 5th-6th, 2016

Social Networks: underlying data

The underlying data is naturally a graph

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation
- Authors linked by co-authorship

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation
- Authors linked by co-authorship
- Bipartite graph of customers and products

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation
- Authors linked by co-authorship
- Bipartite graph of customers and products
- Web-graph: who links to whom

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation
- Authors linked by co-authorship
- Bipartite graph of customers and products
- Web-graph: who links to whom
- Friendship networks: who knows whom

Social Networks: underlying data

The underlying data is naturally a graph

- Papers linked by citation
- Authors linked by co-authorship
- Bipartite graph of customers and products
- Web-graph: who links to whom
- Friendship networks: who knows whom
- Follower-followee network

Social Networks: what we are looking for

Rank nodes for a particular query

- Top k matches for "Social Computing" from Citeseer

Social Networks: what we are looking for

Rank nodes for a particular query

- Top k matches for "Social Computing" from Citeseer
- Who are the most likely co-authors of Manning.

Social Networks: what we are looking for

Rank nodes for a particular query

- Top k matches for "Social Computing" from Citeseer
- Who are the most likely co-authors of Manning.
- Top k book recommendations from Amazon

Social Networks: what we are looking for

Rank nodes for a particular query

- Top k matches for “Social Computing” from Citeseer
- Who are the most likely co-authors of Manning.
- Top k book recommendations from Amazon
- Top k websites for a query

Social Networks: what we are looking for

Rank nodes for a particular query

- Top k matches for “Social Computing” from Citeseer
- Who are the most likely co-authors of Manning.
- Top k book recommendations from Amazon
- Top k websites for a query
- Top k Friend recommendation to X when he joins Facebook

Why Random Walks?

- A wide variety of interesting real world applications can be framed as ranking entities in a graph

Why Random Walks?

- A wide variety of interesting real world applications can be framed as ranking entities in a graph
- A graph-theoretic measure for ranking nodes as well as similarity: for example, two entities are similar, if lots of short paths between them.

Why Random Walks?

- A wide variety of interesting real world applications can be framed as ranking entities in a graph
- A graph-theoretic measure for ranking nodes as well as similarity: for example, two entities are similar, if lots of short paths between them.
- Random walks have proven to be a simple, but powerful mathematical tool for extracting this information.

What is Random Walk?

- Given a graph and a starting point (node), we select a neighbor of it at random, and move to this neighbor

What is Random Walk?

- Given a graph and a starting point (node), we select a neighbor of it at random, and move to this neighbor
- Then we select a neighbor of this node and move to it, and so on

What is Random Walk?

- Given a graph and a starting point (node), we select a neighbor of it at random, and move to this neighbor
- Then we select a neighbor of this node and move to it, and so on
- The (random) sequence of nodes selected this way is a *random walk* on the graph

Adjacency and Transition Matrix

$n \times n$ Adjacency matrix A

- $A(i,j)$: weight on edge from i to j
- If the graph is undirected $A(i,j) = A(j,i)$, i.e. A is symmetric

Adjacency and Transition Matrix

$n \times n$ Adjacency matrix A

- $A(i,j)$: weight on edge from i to j
- If the graph is undirected $A(i,j) = A(j,i)$, i.e. A is symmetric

$n \times n$ Transition matrix P

- P is row stochastic
- $P(i,j)$ = probability of stepping on node j from node i

Adjacency and Transition Matrix

$n \times n$ Adjacency matrix A

- $A(i,j)$: weight on edge from i to j
- If the graph is undirected $A(i,j) = A(j,i)$, i.e. A is symmetric

$n \times n$ Transition matrix P

- P is row stochastic
- $P(i,j)$ = probability of stepping on node j from node i = $\frac{A(i,j)}{\sum_i A(i,j)}$

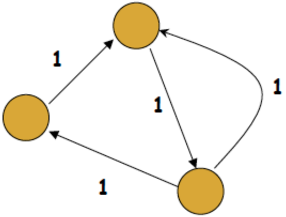
Adjacency and Transition Matrix: Example

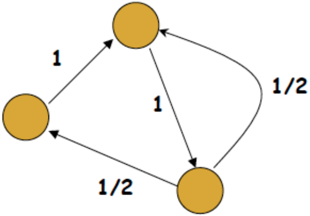
0	1	0
0	0	1
1	1	0

Adjacency matrix A

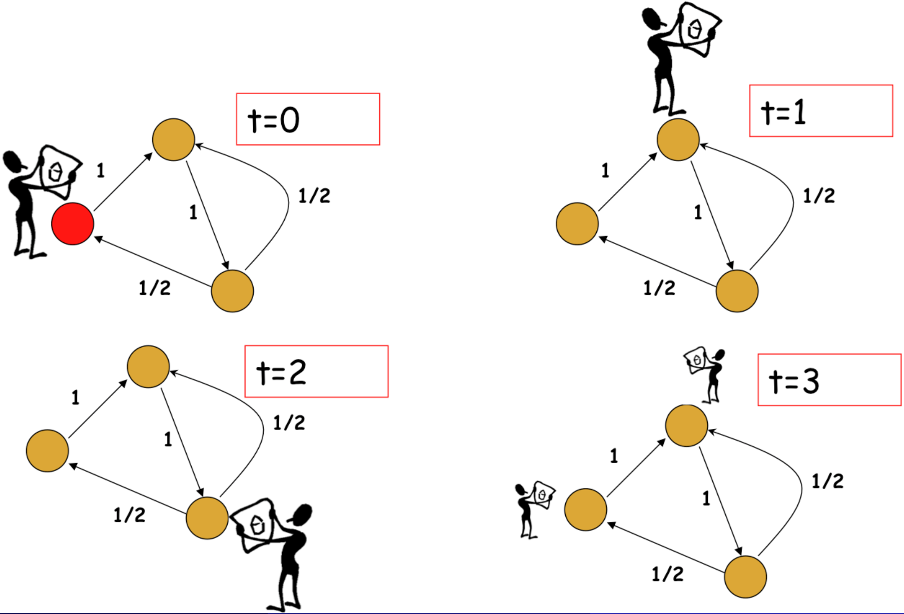
0	1	0
0	0	1
1/2	1/2	0

Transition matrix P





What is a random walk?



Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$:

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \rightarrow i)$ = $\sum_j x_t(j)^* P(j,i)$

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \rightarrow i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x^*_{t-1} P^* P = \ldots = x_0 P^t$

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \rightarrow i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x^*_{t-1} P^* P = \ldots = x_0 P^t$
- What if the surfer keeps walking for a long time?

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \rightarrow i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x^*_{t-1} P^* P = \ldots = x_0 P^t$
- What if the surfer keeps walking for a long time?

Stationary Distribution

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \to i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x^*_{t-1} P^* P = \ldots = x_0 P^t$
- What if the surfer keeps walking for a long time?

Stationary Distribution

- When the surfer keeps walking for a long time

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \to i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x^*_{t-1} P^* P = \ldots = x_0 P^t$
- What if the surfer keeps walking for a long time?

Stationary Distribution

- When the surfer keeps walking for a long time
- When the distribution does not change anymore, i.e. $x_{T+1} = x_T$

Probability Distributions

- $x_t(i)$: probability that the surfer is at node i at time t
- $x_{t+1}(i)$: $\sum_j$ (Probability of being at node j)* $Pr(j \rightarrow i)$ = $\sum_j x_t(j)^* P(j,i)$
- $x_{t+1} = x_t P = x_{t-1}^* P^* P = \ldots = x_0 P^t$
- What if the surfer keeps walking for a long time?

Stationary Distribution

- When the surfer keeps walking for a long time
- When the distribution does not change anymore, i.e. x_{T+1} = x_T
- *For well-behaved graphs, this does not depend on the start distribution*

What is a stationary distribution?

- Stationary distribution at a node is related to the amount of time a random walker spends visiting that node
- Probability distribution at a node can be written as $x_{t+1} = x_t P$

What is a stationary distribution?

- Stationary distribution at a node is related to the amount of time a random walker spends visiting that node
- Probability distribution at a node can be written as $x_{t+1} = x_t P$
- For the stationary distribution (say v_0), we have $v_0 = v_0 P$

What is a stationary distribution?

- Stationary distribution at a node is related to the amount of time a random walker spends visiting that node
- Probability distribution at a node can be written as $x_{t+1} = x_t P$
- For the stationary distribution (say v_0), we have $v_0 = v_0 P$
- This is the left eigenvector of the transition matrix

Interesting questions

Does a stationary distribution always exist? Is it unique?

Yes, if the graph is "well-behaved"

Interesting questions

Does a stationary distribution always exist? Is it unique?

Yes, if the graph is "well-behaved"

How fast the random surfer approach this stationary distribution?

Mixing time

Well behaved graphs

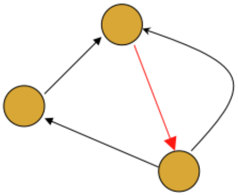
Irreducible

There is a path from every node to every other node.

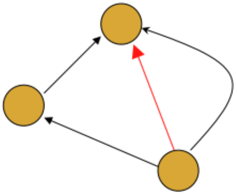
Well behaved graphs

Irreducible

There is a path from every node to every other node.



Irreducible



Not irreducible

Well behaved graphs

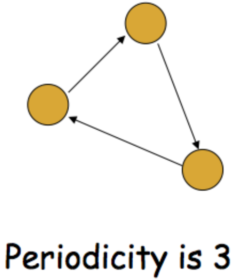
Aperiodic

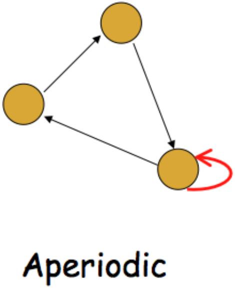
The GCD of all cycle lengths is 1. The GCD is also called period.

Well behaved graphs

Aperiodic

The GCD of all cycle lengths is 1. The GCD is also called period.





Perron Frobenius Theorem: Implications

Theorem Statement

Let $A = (a_{ij})$ be an $n \times n$ positive matrix: $a_{ij} > 0 \forall 1 \leq i,j \leq n$. Then

- There is a positive real number r, such that r is an eigenvalue of A and any other eigenvalue is strictly smaller than r in absolute value.

Perron Frobenius Theorem: Implications

Theorem Statement

Let $A = (a_{ij})$ be an $n \times n$ positive matrix: $a_{ij} > 0 \forall 1 \leq i,j \leq n$. Then

- There is a positive real number r, such that r is an eigenvalue of A and any other eigenvalue is strictly smaller than r in absolute value.

Markov Chain: irreducible and aperiodic

- For any matrix A with eigenvalue σ, $|\sigma| \leq max_i \sum_j |A_{ij}|$.
- Since P is *row stochastic*, the largest eigenvalue of the transition matrix will be equal to 1 and all other eigenvalues will be strictly less than 1

Perron Frobenius Theorem: Implications

Theorem Statement

Let $A = (a_{ij})$ be an $n \times n$ positive matrix: $a_{ij} > 0 \forall 1 \leq i,j \leq n$. Then

- There is a positive real number r, such that r is an eigenvalue of A and any other eigenvalue is strictly smaller than r in absolute value.

Markov Chain: irreducible and aperiodic

- For any matrix A with eigenvalue σ, $|\sigma| \leq max_i \sum_j |A_{ij}|$.
- Since P is *row stochastic*, the largest eigenvalue of the transition matrix will be equal to 1 and all other eigenvalues will be strictly less than 1
- Let the eigenvalues of P be $\{\sigma_i | i = 0 : n-1\}$ in non-decreasing order of σ_i
- $\sigma_0 = 1 > \sigma_1 \geq \sigma_2 \geq \ldots \sigma_n$

Perron Frobenius Theorem: Implications

- $v_0 = v_0 P$ (unique for a well-behaved graph)
- Let x be an arbitrary initial distribution

$$x = \sum_{i=1}^{n} a_i u_i$$

- $xP = \sum_{i=1}^{n} a_i(u_i P)$
- $= \sum_{i=1}^{n} a_i(\sigma_i u_i)$
- Similarly, $xP^k = \sum_{i=1}^{n} a_i({\sigma_i}^k u_i)$
- $xPPP \ldots P = xP^k$ tends to v_0 as k goes to infinity.

Perron Frobenius Theorem: Implications

$$xP^k = \sigma_1{}^k\{a_1u_1 + a_2\left(\frac{\sigma_2}{\sigma_1}\right)^k u_2 + \ldots + a_n\left(\frac{\sigma_n}{\sigma_1}\right)^k u_n\}$$

$u_1 = v_0$, thus xP^k approaches to v_0 as k goes to infinity with a speed in the order of σ_2/σ_1 exponentially.

Show that $a_1 = 1$

Perron Frobenius Theorem: Implications

$$xP^k = \sigma_1{}^k \{a_1 u_1 + a_2 \left(\frac{\sigma_2}{\sigma_1}\right)^k u_2 + \ldots + a_n \left(\frac{\sigma_n}{\sigma_1}\right)^k u_n\}$$

$u_1 = v_0$, thus xP^k approaches to v_0 as k goes to infinity with a speed in the order of σ_2/σ_1 exponentially.

Show that $a_1 = 1$

- $1_{n\times 1}$ is the right eigenvector of P with eigenvalue 1, since P is stochastic, i.e. $P^* 1_{n\times 1} = 1_{n\times 1}$
- Hence, $u_i{}^* 1_{n\times 1} = 1$ for $i = 1$, 0 otherwise (relation between left and right eigen vectors)
- Now, $1 = x^* 1_{n\times 1} = a_1 u_1{}^* 1_{n\times 1} = a_1$ (Why?)

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

$h_{ij} = 1 + \sum_k p_{ik} h_{kj}, i \neq j$

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

$h_{ij} = 1 + \sum_k p_{ik} h_{kj}, i \neq j$

Commute time (c_{ij})

$c_{ij} = h_{ij} + h_{ji}$

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

$h_{ij} = 1 + \sum_k p_{ik} h_{kj}, i \neq j$

Commute time (c_{ij})

$c_{ij} = h_{ij} + h_{ji}$

Cover Time $Cov(G)$

$Cov(s, G)$: expected number of steps it takes a walk that starts at s to visit all vertices

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

$h_{ij} = 1 + \sum_k p_{ik} h_{kj}, i \neq j$

Commute time (c_{ij})

$c_{ij} = h_{ij} + h_{ji}$

Cover Time $Cov(G)$

$Cov(s, G)$: expected number of steps it takes a walk that starts at s to visit all vertices

$Cov(G)$: maximum over s of $Cov(s, G)$

Important Parameters of a random walk

Access time or hitting time (h_{ij})

h_{ij} is the expected number of steps before node j is first visited, starting from node i

$h_{ij} = 1 + \sum_k p_{ik} h_{kj}, i \neq j$

Commute time (c_{ij})

$c_{ij} = h_{ij} + h_{ji}$

Cover Time $Cov(G)$

$Cov(s, G)$: expected number of steps it takes a walk that starts at s to visit all vertices

$Cov(G)$: maximum over s of $Cov(s, G)$

$Cov^+(G)$: Cover and return to start

Mixing Rate

- How fast the random walk converges to its limiting distribution
- Mixing rate for some graphs can be very small: $O(logn)$

Mixing Rate

- How fast the random walk converges to its limiting distribution
- Mixing rate for some graphs can be very small: $O(logn)$
- Mixing rate depends on the spectral gap: $1 - \sigma_2$, where σ_2 is the second highest eigen value
- Smaller the value of σ_2, larger is the spectral gap, faster is the mixing rate

PageRank (Page and Brin, 1998)

PageRank (Page and Brin, 1998)

Basic Intuition

A webpage is important if other important pages point to it

PageRank (Page and Brin, 1998)

Basic Intuition

A webpage is important if other important pages point to it

- $v(i) = \sum_{j \to i} \frac{v(j)}{deg^{out}(j)}$

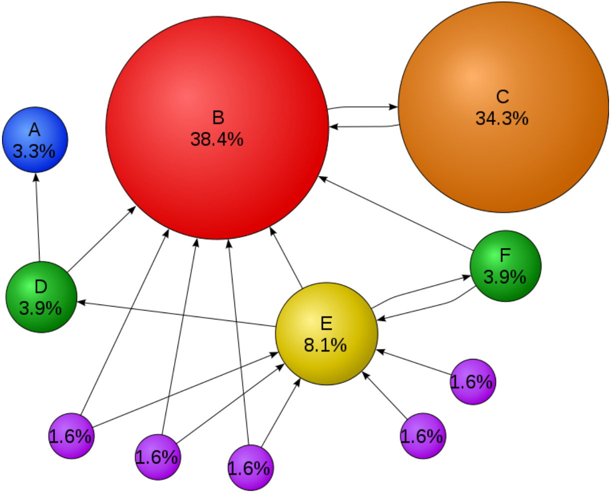
PageRank (Page and Brin, 1998)

Basic Intuition

A webpage is important if other important pages point to it

- $v(i) = \sum_{j \to i} \frac{v(j)}{deg^{out}(j)}$
- v is the stationary distribution of the Markov chain
- v is the stationary distribution of the Markov chain
- "*The $25,000,000,000 Eigenvector: The Linear Algebra Behind Google*"

PageRank Example (Source: Wikipedia)



Irreducibility and Aperiodicity

Irreducibility and Aperiodicity

How to guarantee this for a web graph?

Irreducibility and Aperiodicity

How to guarantee this for a web graph?

At any time-step the random surfer

- jumps (teleport) to any other node with probability c
- jumps to its direct neighbors with total probability $1 - c$

Irreducibility and Aperiodicity

How to guarantee this for a web graph?

At any time-step the random surfer

- jumps (teleport) to any other node with probability c
- jumps to its direct neighbors with total probability $1 - c$

$$\tilde{P} = (1 - c)P + cU$$

Irreducibility and Aperiodicity

How to guarantee this for a web graph?

At any time-step the random surfer

- jumps (teleport) to any other node with probability c
- jumps to its direct neighbors with total probability $1-c$

$$\tilde{P} = (1-c)P + cU$$

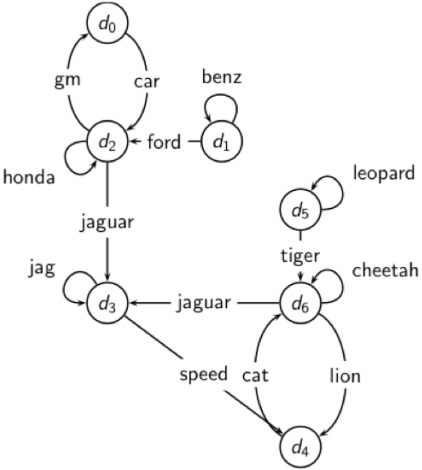
$$U_{ij} = \frac{1}{n} \forall i,j$$

Computing PageRank: The Power Method

- Start with any distribution x_0, e.g. uniform distribution
- Algorithm: multiply x_0 by increasing powers of P until convergence
- After one step, $x_1 = x_0 P$, after k steps $x_k = x_0 P^k$
- Regardless of where we start, we eventually reach the steady state v_0

PageRank: Example

Example web graph



From “Introduction to Information Retrieval” slides

PageRank: Example

Transition (probability) matrix

	d_0	d_1	d_2	d_3	d_4	d_5	d_6
d_0	0.00	0.00	1.00	0.00	0.00	0.00	0.00
d_1	0.00	0.50	0.50	0.00	0.00	0.00	0.00
d_2	0.33	0.00	0.33	0.33	0.00	0.00	0.00
d_3	0.00	0.00	0.00	0.50	0.50	0.00	0.00
d_4	0.00	0.00	0.00	0.00	0.00	0.00	1.00
d_5	0.00	0.00	0.00	0.00	0.00	0.50	0.50
d_6	0.00	0.00	0.00	0.33	0.33	0.00	0.33

206

From "Introduction to Information Retrieval" slides

PageRank: Example

Transition matrix with teleporting

	d_0	d_1	d_2	d_3	d_4	d_5	d_6
d_0	0.02	0.02	0.88	0.02	0.02	0.02	0.02
d_1	0.02	0.45	0.45	0.02	0.02	0.02	0.02
d_2	0.31	0.02	0.31	0.31	0.02	0.02	0.02
d_3	0.02	0.02	0.02	0.45	0.45	0.02	0.02
d_4	0.02	0.02	0.02	0.02	0.02	0.02	0.88
d_5	0.02	0.02	0.02	0.02	0.02	0.45	0.45
d_6	0.02	0.02	0.02	0.31	0.31	0.02	0.31

208

From "Introduction to Information Retrieval" slides

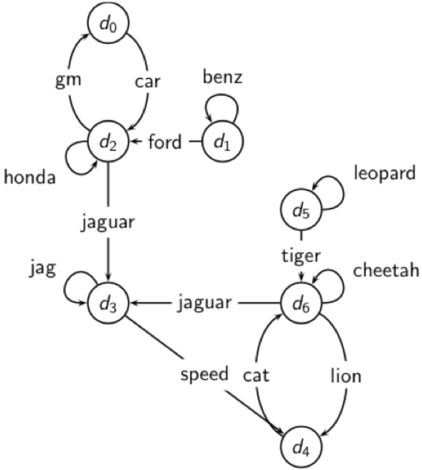
Power method vectors $\vec{x}P^k$

	$\vec{x}$	$\vec{x}P^{1}$	$\vec{x}P^{2}$	$\vec{x}P^{3}$	$\vec{x}P^{4}$	$\vec{x}P^{5}$	$\vec{x}P^{6}$	$\vec{x}P^{7}$	$\vec{x}P^{8}$	$\vec{x}P^{9}$	$\vec{x}P^{10}$	$\vec{x}P^{11}$	$\vec{x}P^{12}$	$\vec{x}P^{13}$
d_0	0.14	0.06	0.09	0.07	0.07	0.06	0.06	0.06	0.06	0.05	0.05	0.05	0.05	0.05
d_1	0.14	0.08	0.06	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04
d_2	0.14	0.25	0.18	0.17	0.15	0.14	0.13	0.12	0.12	0.12	0.12	0.11	0.11	0.11
d_3	0.14	0.16	0.23	0.24	0.24	0.24	0.24	0.25	0.25	0.25	0.25	0.25	0.25	0.25
d_4	0.14	0.12	0.16	0.19	0.19	0.20	0.21	0.21	0.21	0.21	0.21	0.21	0.21	0.21
d_5	0.14	0.08	0.06	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04
d_6	0.14	0.25	0.23	0.25	0.27	0.28	0.29	0.29	0.30	0.30	0.30	0.30	0.31	0.31

From "Introduction to Information Retrieval" slides

PageRank: Example

Example web graph



	PageRank
d_0	0.05
d_1	0.04
d_2	0.11
d_3	0.25
d_4	0.21
d_5	0.04
d_6	0.31

211

From "Introduction to Information Retrieval" slides

Personalized PageRank

- We are looking for the vector v such that

$$v = (1-c)vP + cr$$

- r is a distribution over web-pages

Personalized PageRank

- We are looking for the vector v such that

$$v = (1 - c)vP + cr$$

- r is a distribution over web-pages
- If r is the uniform distribution we get pagerank

Personalized PageRank

- We are looking for the vector v such that

$$v = (1-c)vP + cr$$

- r is a distribution over web-pages
- If r is the uniform distribution we get pagerank
- What happens if r is non-uniform?

Personalized PageRank

- We are looking for the vector v such that

$$v = (1-c)vP + cr$$

- r is a distribution over web-pages
- If r is the uniform distribution we get pagerank
- What happens if r is non-uniform? $\rightarrow$ Pesonalization

Personalized PageRank

- The only difference is that we use a non-uniform teleportation distribution, i.e. at any time step, teleport to a set of webpages.
- In other words we are looking for the vector v such that

$$v = (1-c)vP + cr$$

Personalized PageRank

- The only difference is that we use a non-uniform teleportation distribution, i.e. at any time step, teleport to a set of webpages.
- In other words we are looking for the vector v such that

$$v = (1-c)vP + cr$$

- r is a non-uniform preference vector specific to a user.

Personalized PageRank

- The only difference is that we use a non-uniform teleportation distribution, i.e. at any time step, teleport to a set of webpages.
- In other words we are looking for the vector v such that

$$v = (1-c)vP + cr$$

- r is a non-uniform preference vector specific to a user.
- v gives "personalized views" of the web.

Topic Sensitive PageRank

- Divide the webpages into 16 broad categories

Topic Sensitive PageRank

- Divide the webpages into 16 broad categories
- For each category, compute the biased personalized pagerank vector by teleporting uniformly to websites under that category.

Topic Sensitive PageRank

- Divide the webpages into 16 broad categories
- For each category, compute the biased personalized pagerank vector by teleporting uniformly to websites under that category.
- At query time, the probability of query being from any of the above classes is computed

Topic Sensitive PageRank

- Divide the webpages into 16 broad categories
- For each category, compute the biased personalized pagerank vector by teleporting uniformly to websites under that category.
- At query time, the probability of query being from any of the above classes is computed
- Final pageRank vector is computed by a linear combination of the biased pagerank vectors computed offline

HITS - Hyperlink-Induced Topic Search

- There are two different types of web-pages for searching a broad topic: the authorities and the hubs

HITS - Hyperlink-Induced Topic Search

- There are two different types of web-pages for searching a broad topic: the authorities and the hubs
- **Authorities:** pages which are good sources of information about a given topic

HITS - Hyperlink-Induced Topic Search

- There are two different types of web-pages for searching a broad topic: the authorities and the hubs
- **Authorities:** pages which are good sources of information about a given topic
- **Hub:** provides pointers to many authorities

HITS - Hyperlink-Induced Topic Search

- There are two different types of web-pages for searching a broad topic: the authorities and the hubs
- **Authorities:** pages which are good sources of information about a given topic
- **Hub:** provides pointers to many authorities
- *Works on a subgraph* - can consist of top k search results for the given query from a standard text-based engine

- Given this subgraph, the idea is to assign two numbers to a node: a hub-score and an authority score

- Given this subgraph, the idea is to assign two numbers to a node: a hub-score and an authority score
- A node is a good hub if it points to many good authorities, whereas a node is a good authority if many good hubs point to it.
- $a(i) \leftarrow \sum_{j:j \in I(i)} h(j)$
- $h(i) \leftarrow \sum_{j:j \in O(i)} a(j)$

Hubs and Authorities

- $a = A^T h$, $h = Aa$
- $h = AA^T h$, $a = A^T Aa$

Hubs and Authorities

- $a = A^T h$, $h = Aa$
- $h = AA^T h$, $a = A^T Aa$
- h converges to the principal eigenvector of AA^T and a converges to the principal eigenvector of $A^T A$

Hubs and Authorities

- $a = A^T h$, $h = Aa$
- $h = AA^T h$, $a = A^T Aa$
- h converges to the principal eigenvector of AA^T and a converges to the principal eigenvector of $A^T A$
- $AA^T(i,j) = \sum_k A(i,k)A(j,k)$: number of nodes both i and j point to, *bibliographic coupling*

Hubs and Authorities

- $a = A^T h$, $h = Aa$
- $h = AA^T h$, $a = A^T Aa$
- h converges to the principal eigenvector of AA^T and a converges to the principal eigenvector of $A^T A$
- $AA^T(i,j) = \sum_k A(i,k)A(j,k)$: number of nodes both i and j point to, *bibliographic coupling*
- $A^T A(i,j) = \sum_k A(k,i)A(k,j)$: number of nodes which point to both i and j, *co-citation matrix*

How to compute hub and authority scores

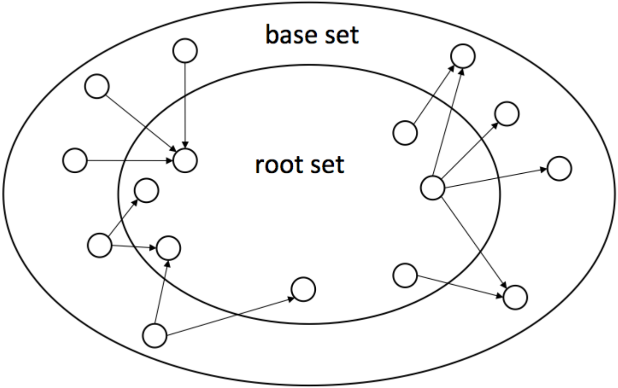
- Do a regular web search first
- Call the search result the root set
- Find all pages that are linked to or link to pages in the root set
- Call first larger set the base set
- Finally, compute hubs and authorities for the base set (which we'll view as a small web graph)

250

From "Introduction to Information Retrieval" slides

HITS: Example

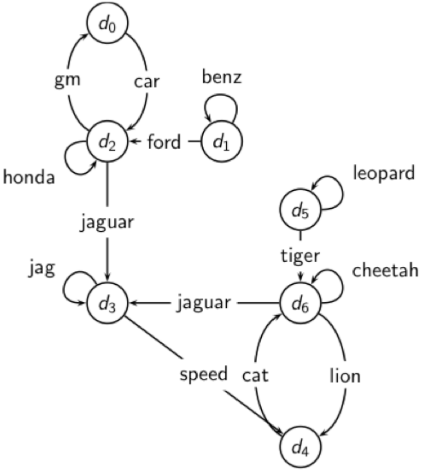
Root set and base set (1)



The base set

From “Introduction to Information Retrieval” slides

Example web graph



From "Introduction to Information Retrieval" slides

HITS: Example

Raw matrix A for HITS

	d_0	d_1	d_2	d_3	d_4	d_5	d_6
d_0	0	0	1	0	0	0	0
d_1	0	1	1	0	0	0	0
d_2	1	0	1	2	0	0	0
d_3	0	0	0	1	1	0	0
d_4	0	0	0	0	0	0	1
d_5	0	0	0	0	0	1	1
d_6	0	0	0	2	1	0	1

From "Introduction to Information Retrieval" slides

HITS: Example

Hub vectors h_0, $\vec{h}_i = \frac{1}{d_i} A^* a_i$, $i \geq 1$

	$\vec{h}_0$	$\vec{h}_1$	$\vec{h}_2$	$\vec{h}_3$	$\vec{h}_4$	$\vec{h}_5$
d_0	0.14	0.06	0.04	0.04	0.03	0.03
d_1	0.14	0.08	0.05	0.04	0.04	0.04
d_2	0.14	0.28	0.32	0.33	0.33	0.33
d_3	0.14	0.14	0.17	0.18	0.18	0.18
d_4	0.14	0.06	0.04	0.04	0.04	0.04
d_5	0.14	0.08	0.05	0.04	0.04	0.04
d_6	0.14	0.30	0.33	0.34	0.35	0.35

From "Introduction to Information Retrieval" slides

HITS: Example

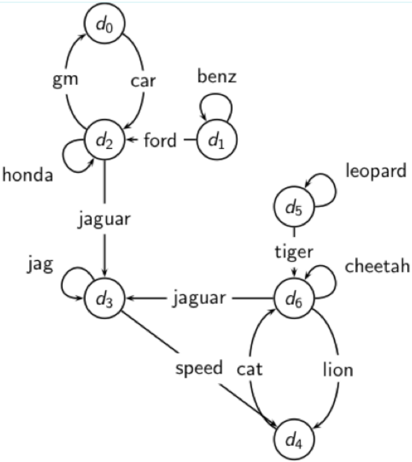
Authority vector $\vec{a} = \frac{1}{c_i} A^T * \vec{h}_{i-1}, i \geq 1$

	a_1	$\vec{a}_2$	$\vec{a}_3$	$\vec{a}_4$	$\vec{a}_5$	$\vec{a}_6$	$\vec{a}_7$
d_0	0.06	0.09	0.10	0.10	0.10	0.10	0.10
d_1	0.06	0.03	0.01	0.01	0.01	0.01	0.01
d_2	0.19	0.14	0.13	0.12	0.12	0.12	0.12
d_3	0.31	0.43	0.46	0.46	0.46	0.47	0.47
d_4	0.13	0.14	0.16	0.16	0.16	0.16	0.16
d_5	0.06	0.03	0.02	0.01	0.01	0.01	0.01
d_6	0.19	0.14	0.13	0.13	0.13	0.13	0.13

From "Introduction to Information Retrieval" slides

HITS: Example

Example web graph



	a	h
d_0	0.10	0.03
d_1	0.01	0.04
d_2	0.12	0.33
d_3	0.47	0.18
d_4	0.16	0.04
d_5	0.01	0.04
d_6	0.13	0.35

310

From “Introduction to Information Retrieval” slides