Introduction to Information Retrieval

http://informationretrieval.org

IIR 21: Link Analysis

Hinrich Schütze

Institute for Natural Language Processing, Universität Stuttgart

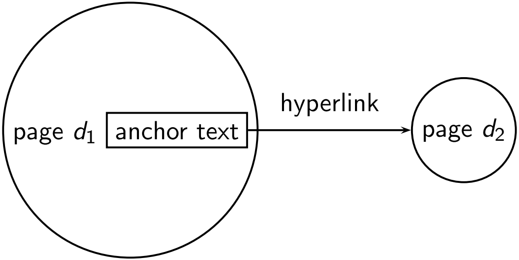
2008.07.01

Overview

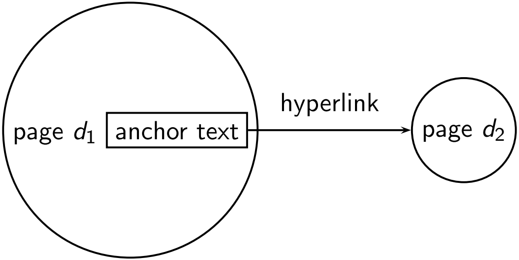
1. Anchor text
2. PageRank
3. HITS

Outline

The web as a directed graph

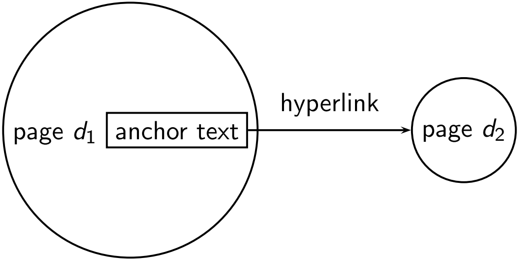


The web as a directed graph



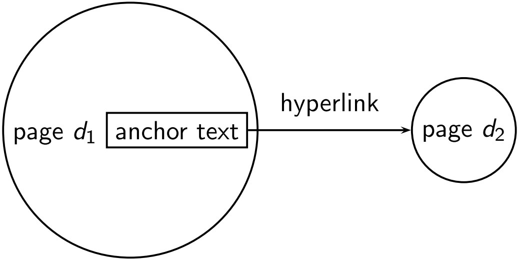
- Assumption 1: A hyperlink is a quality signal.

The web as a directed graph



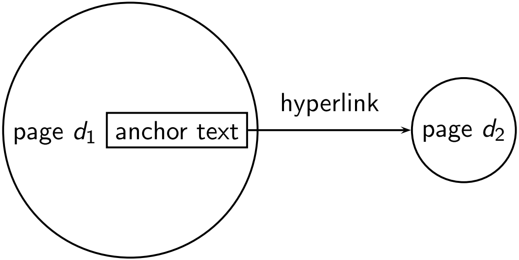
- Assumption 1: A hyperlink is a quality signal.
  - A hyperlink between pages denotes that the author perceived relevance.

The web as a directed graph



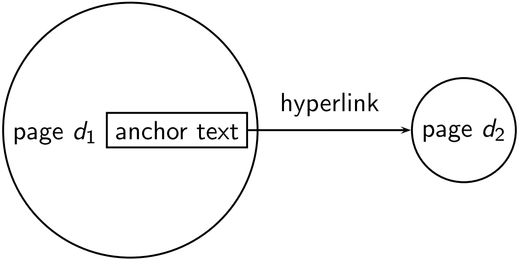
- Assumption 1: A hyperlink is a quality signal.
  - A hyperlink between pages denotes that the author perceived relevance.
- Assumption 2: The anchor text describes the target page d_2.

The web as a directed graph



- Assumption 1: A hyperlink is a quality signal.
    - A hyperlink between pages denotes that the author perceived relevance.
- Assumption 2: The anchor text describes the target page d_2.
    - We use anchor text somewhat loosely here: the text surrounding the hyperlink. Example: "You can find cheap cars here."

The web as a directed graph



- Assumption 1: A hyperlink is a quality signal.
  - A hyperlink between pages denotes that the author perceived relevance.
- Assumption 2: The anchor text describes the target page d_2.
  - We use anchor text somewhat loosely here: the text surrounding the hyperlink. Example: "You can find cheap cars here."
- Examples for hyperlinks that violate these two assumptions?

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article
  - May not match IBM home page! (if IBM home page is mostly graphical)

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article
  - May not match IBM home page! (if IBM home page is mostly graphical)
- Searching on anchor text is better for the query *IBM*.

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article
  - May not match IBM home page! (if IBM home page is mostly graphical)
- Searching on anchor text is better for the query *IBM*.
- Represent each page by all the anchor text pointing to it.

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
  - Matches IBM's copyright page
  - Matches many spam pages
  - Matches IBM wikipedia article
  - May not match IBM home page! (if IBM home page is mostly graphical)
- Searching on anchor text is better for the query *IBM*.
- Represent each page by all the anchor text pointing to it.
- In this representation, the page with the most occurrences of *IBM* is www.ibm.com.

[document text only] vs. [document text + anchor text]

- Searching on [document text + anchor text] is often more effective than searching on [document text only].
- Example: Query *IBM*
    - Matches IBM's copyright page
    - Matches many spam pages
    - Matches IBM wikipedia article
    - May not match IBM home page! (if IBM home page is mostly graphical)
- Searching on anchor text is better for the query *IBM*.
- Represent each page by all the anchor text pointing to it.
- In this representation, the page with the most occurrences of *IBM* is www.ibm.com.

Anchor text containing *IBM* pointing to www.ibm.com

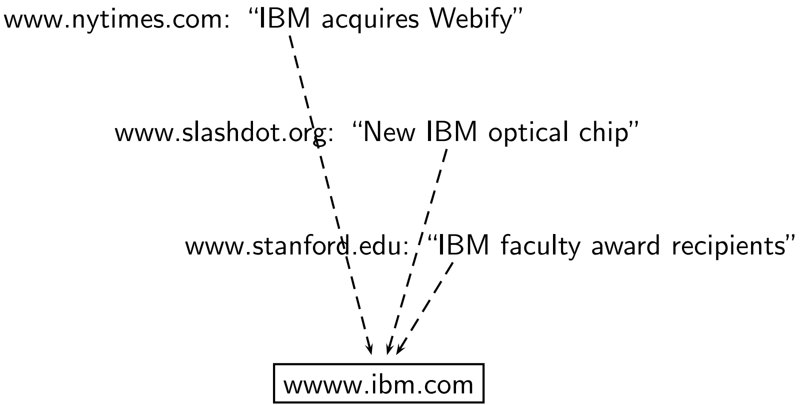
www.nytimes.com: "IBM acquires Webify"

www.slashdot.org: "New IBM optical chip"

www.stanford.edu: "IBM faculty award recipients"

wwww.ibm.com

Anchor text containing *IBM* pointing to www.ibm.com



Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.

Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.
- Anchor text can be weighted more highly than document text. (based on Assumptions 1&2)

Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.
- Anchor text can be weighted more highly than document text. (based on Assumptions 1&2)
- Indexing anchor text can have unexpected side effects – Google bombs.

Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.
- Anchor text can be weighted more highly than document text. (based on Assumptions 1&2)
- Indexing anchor text can have unexpected side effects – Google bombs.
- A Google bomb is a search with "bad" results due to maliciously manipulated anchor text.

Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.
- Anchor text can be weighted more highly than document text. (based on Assumptions 1&2)
- Indexing anchor text can have unexpected side effects – Google bombs.
- A Google bomb is a search with "bad" results due to maliciously manipulated anchor text.
- Google introduced a new weighting function in January 2007 that fixed many google bombs.

Indexing anchor text

- Thus: Anchor text is often a better description of a page's content than the page itself.
- Anchor text can be weighted more highly than document text. (based on Assumptions 1&2)
- Indexing anchor text can have unexpected side effects – Google bombs.
- A Google bomb is a search with "bad" results due to maliciously manipulated anchor text.
- Google introduced a new weighting function in January 2007 that fixed many google bombs.
- Any "live" Google bombs?

Google bomb

- "who is a failure" on Google

Outline

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."
- "Miller (2001)" is a hyperlink linking two scientific articles.

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."
- "Miller (2001)" is a hyperlink linking two scientific articles.
- One application of these "hyperlinks" in the scientific literature:

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."
- "Miller (2001)" is a hyperlink linking two scientific articles.
- One application of these "hyperlinks" in the scientific literature:
  - Measure the similarity of two articles by the overlap of other articles citing them.

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."
- "Miller (2001)" is a hyperlink linking two scientific articles.
- One application of these "hyperlinks" in the scientific literature:
  - Measure the similarity of two articles by the overlap of other articles citing them.
  - This is called cocitation similarity.

Origins of PageRank: Citation analysis (1)

- Citation analysis: analysis of citations in the scientific literature
- Example citation: "Miller (2001) has shown that physical activity alters the metabolism of estrogens."
- "Miller (2001)" is a hyperlink linking two scientific articles.
- One application of these "hyperlinks" in the scientific literature:
  - Measure the similarity of two articles by the overlap of other articles citing them.
  - This is called cocitation similarity.
- Cocitation similarity on the web?

Cocitation similarity on Google: similar pages

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.
  - Circular? No: can be formalized in a well-defined way.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.
  - Circular? No: can be formalized in a well-defined way.
  - This is basically PageRank.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.
  - Circular? No: can be formalized in a well-defined way.
  - This is basically PageRank.
  - PageRank was invented in the context of citation analysis by Pinsker and Narin in the 1960s.

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.
  - Circular? No: can be formalized in a well-defined way.
  - This is basically PageRank.
  - PageRank was invented in the context of citation analysis by Pinsker and Narin in the 1960s.
  - Citation analysis is a big deal: The budget and salary of this lecturer are / will be determined by the impact of his publications!

Origins of PageRank: Citation analysis (2)

- Citation frequency can be used to measure the impact of an article.
  - Each article gets one vote.
  - Not a very accurate measure
- Better measure: weighted citation frequency / citation rank
  - An article's vote is weighted according to its citation impact.
  - Circular? No: can be formalized in a well-defined way.
  - This is basically PageRank.
  - PageRank was invented in the context of citation analysis by Pinsker and Narin in the 1960s.
  - Citation analysis is a big deal: The budget and salary of this lecturer are / will be determined by the impact of his publications!
- Recall: Citation in scientific literature = hyperlink on the web

Link-based ranking for web search

- Simple version of using links for ranking on the web

Link-based ranking for web search

- Simple version of using links for ranking on the web
  - First retrieve all pages satisfying the query (say *venture capital*)

Link-based ranking for web search

- Simple version of using links for ranking on the web
  - First retrieve all pages satisfying the query (say *venture capital*)
  - Order these by the number of in-links

Link-based ranking for web search

- Simple version of using links for ranking on the web
  - First retrieve all pages satisfying the query (say *venture capital*)
  - Order these by the number of in-links
- Simple link popularity (= number of in-links) is easy to spam. Why?

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably
- In the steady state, each page has a long-term visit rate.

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably
- In the steady state, each page has a long-term visit rate.
- This long-term visit rate is the page's PageRank.

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably
- In the steady state, each page has a long-term visit rate.
- This long-term visit rate is the page's PageRank.
- PageRank = steady state probability = long-term visit rate

Basis for PageRank: Random walk

- Imagine a web surfer doing a random walk on the web
  - Start at a random page
  - At each step, go out of the current page along one of the links on that page, equiprobably
- In the steady state, each page has a long-term visit rate.
- This long-term visit rate is the page's PageRank.
- PageRank = steady state probability = long-term visit rate
- Concept of long-term visit rate clear?

Markov chains

- A Markov chain consists of N states, plus an $N \times N$ transition probability matrix P.

Markov chains

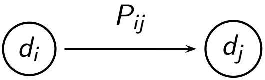
- A Markov chain consists of N states, plus an $N \times N$ transition probability matrix P.
- state = page

Markov chains

- A Markov chain consists of N states, plus an $N \times N$ transition probability matrix P.
- state = page
- At each step, we are on exactly one of the pages.

Markov chains

- A Markov chain consists of N states, plus an $N \times N$ transition probability matrix P.
- state = page
- At each step, we are on exactly one of the pages.
- For $1 \leq i, j \leq N$, the matrix entry P_{ij} tells us the probability of j being the next page, given we are currently on page i.



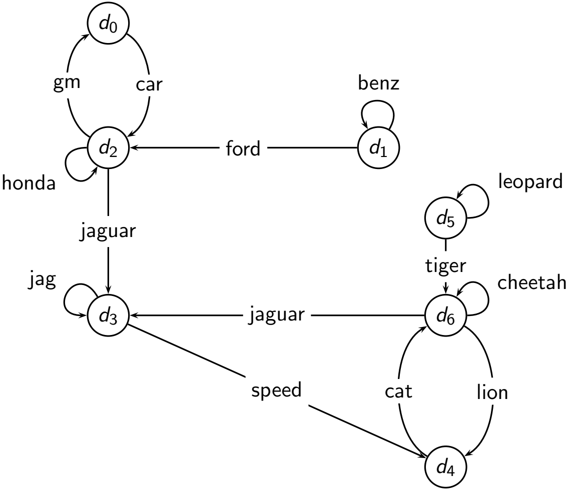
Markov chains

- Clearly, for all i, $\sum_{j=1}^{N} P_{ij} = 1$

Markov chains

- Clearly, for all i, $\sum_{j=1}^{N} P_{ij} = 1$
- Markov chains are abstractions of random walks.

Example web graph



Link matrix for example

	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	1	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	1	1	0	1

Transition probability matrix P for example

	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

Long-term visit rate

- Recall: PageRank = long-term visit rate

Long-term visit rate

- Recall: PageRank = long-term visit rate
- Long-term visit rate of page d is the probability that a web surfer is at page d at a given point in time.

Long-term visit rate

- Recall: PageRank = long-term visit rate
- Long-term visit rate of page d is the probability that a web surfer is at page d at a given point in time.
- Next: what properties must hold of the web graph for the long-term visit rate to be well defined?

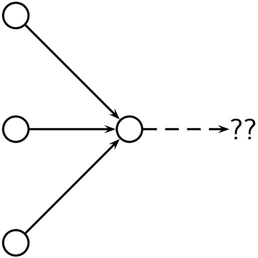
Long-term visit rate

- Recall: PageRank = long-term visit rate
- Long-term visit rate of page d is the probability that a web surfer is at page d at a given point in time.
- Next: what properties must hold of the web graph for the long-term visit rate to be well defined?
- The web graph must correspond to an ergodic Markov chain.

Long-term visit rate

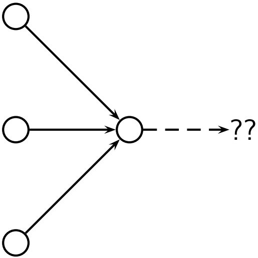
- Recall: PageRank = long-term visit rate
- Long-term visit rate of page d is the probability that a web surfer is at page d at a given point in time.
- Next: what properties must hold of the web graph for the long-term visit rate to be well defined?
- The web graph must correspond to an ergodic Markov chain.
- First a special case: The web graph must not contain dead ends.

Dead ends



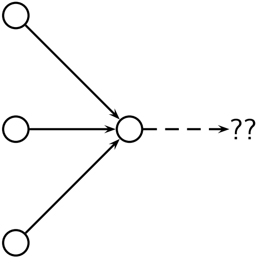
- The web is full of dead ends.

Dead ends



- The web is full of dead ends.
- Random walk can get stuck in dead ends.

Dead ends



- The web is full of dead ends.
- Random walk can get stuck in dead ends.
- If there are dead ends, long-term visit rates are not well-defined (or non-sensical).

Teleporting

- At a dead end, jump to a random web page

Teleporting

- At a dead end, jump to a random web page
- At any non-dead end, with probability 10%, jump to a random web page

Teleporting

- At a dead end, jump to a random web page
- At any non-dead end, with probability 10%, jump to a random web page
- With remaining probability (90%), go out on a random hyperlink (e.g., randomly choose with probability (1-0.1)/4=0.225 one of the four hyperlinks of the page)

Teleporting

- At a dead end, jump to a random web page
- At any non-dead end, with probability 10%, jump to a random web page
- With remaining probability (90%), go out on a random hyperlink (e.g., randomly choose with probability (1-0.1)/4=0.225 one of the four hyperlinks of the page)
- 10% is a parameter.

Result of teleporting

- With teleporting, we cannot get stuck in a dead end.

Result of teleporting

- With teleporting, we cannot get stuck in a dead end.
- Concept of teleporting clear?

Result of teleporting

- With teleporting, we cannot get stuck in a dead end.
- Concept of teleporting clear?
- Even without dead-ends, a graph may not have well-defined long-term visit rates.

Result of teleporting

- With teleporting, we cannot get stuck in a dead end.
- Concept of teleporting clear?
- Even without dead-ends, a graph may not have well-defined long-term visit rates.
- More generally, we require that the Markov chain be ergodic.

Ergodic Markov chains

Ergodic Markov chains

- A Markov chain is ergodic iff it is irreducible and aperiodic.

Ergodic Markov chains

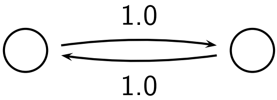
- A Markov chain is ergodic iff it is irreducible and aperiodic.
- Irreducibility. Roughly: there is a path from any page to any other page.

Ergodic Markov chains

- A Markov chain is ergodic iff it is irreducible and aperiodic.
- Irreducibility. Roughly: there is a path from any page to any other page.
- Aperiodicity. Roughly: The pages cannot be partitioned such that the random walker visits the partitions sequentially.

Ergodic Markov chains

- A Markov chain is ergodic iff it is irreducible and aperiodic.
- Irreducibility. Roughly: there is a path from any page to any other page.
- Aperiodicity. Roughly: The pages cannot be partitioned such that the random walker visits the partitions sequentially.
- A non-ergodic Markov chain:



Ergodic Markov chains

- Theorem: For any ergodic Markov chain, there is a unique long-term visit rate for each state.

Ergodic Markov chains

- Theorem: For any ergodic Markov chain, there is a unique long-term visit rate for each state.
- This is the steady-state probability distribution.

Ergodic Markov chains

- Theorem: For any ergodic Markov chain, there is a unique long-term visit rate for each state.
- This is the steady-state probability distribution.
- Over a long time period, we visit each state in proportion to this rate.

Ergodic Markov chains

- Theorem: For any ergodic Markov chain, there is a unique long-term visit rate for each state.
- This is the steady-state probability distribution.
- Over a long time period, we visit each state in proportion to this rate.
- It doesn't matter where we start.

Formalization of “visit”: Probability vector

- A probability (row) vector $\vec{x} = (x_1, \ldots, x_N)$ tells us where the random walk is at any point.

Formalization of "visit": Probability vector

- A probability (row) vector $\vec{x} = (x_1, \ldots, x_N)$ tells us where the random walk is at any point.
- Example: $\begin{pmatrix} 0 & 0 & 0 & \ldots & 1 & \ldots & 0 & 0 & 0 \\ 1 & 2 & 3 & \ldots & i & \ldots & \text{N-2} & \text{N-1} & \text{N} \end{pmatrix}$

Formalization of "visit": Probability vector

- A probability (row) vector $\vec{x} = (x_1, \ldots, x_N)$ tells us where the random walk is at any point.
- Example: $\begin{pmatrix} 0 & 0 & 0 & \ldots & 1 & \ldots & 0 & 0 & 0 \\ 1 & 2 & 3 & \ldots & i & \ldots & \text{N-2} & \text{N-1} & \text{N} \end{pmatrix}$
- More generally: the random walk is on page i with probability x_i.

Formalization of "visit": Probability vector

- A probability (row) vector $\vec{x} = (x_1, \ldots, x_N)$ tells us where the random walk is at any point.
- Example:

(	0	0	0	...	1	...	0	0	0	)
	1	2	3	...	i	...	N-2	N-1	N	

- More generally: the random walk is on page i with probability x_i.
- Example:

(	0.05	0.01	0.0	...	0.2	...	0.01	0.05	0.03	)
	1	2	3	...	i	...	N-2	N-1	N	

Formalization of "visit": Probability vector

- A probability (row) vector $\vec{x} = (x_1, \ldots, x_N)$ tells us where the random walk is at any point.
- Example:

(	0	0	0	...	1	...	0	0	0	)
	1	2	3	...	i	...	N-2	N-1	N	

- More generally: the random walk is on page i with probability x_i.
- Example:

(	0.05	0.01	0.0	...	0.2	...	0.01	0.05	0.03	)
	1	2	3	...	i	...	N-2	N-1	N	

- $\sum x_i = 1$

Change in probability vector

- If the probability vector is $\vec{x} = (x_1, \ldots, x_N)$ at this step, what is it at the next step?

Change in probability vector

- If the probability vector is $\vec{x} = (x_1, \ldots, x_N)$ at this step, what is it at the next step?
- Recall that row i of the transition probability matrix P tells us where we go next from state i.

Change in probability vector

- If the probability vector is $\vec{x} = (x_1, \ldots, x_N)$ at this step, what is it at the next step?
- Recall that row i of the transition probability matrix P tells us where we go next from state i.
- Equivalently: column j of P tells us "where we came from" (and with which probability).

Change in probability vector

- If the probability vector is $\vec{x} = (x_1, \ldots, x_N)$ at this step, what is it at the next step?
- Recall that row i of the transition probability matrix P tells us where we go next from state i.
- Equivalently: column j of P tells us "where we came from" (and with which probability).
- So from $\vec{x}$, our next state is distributed as $\vec{x}P$.

Steady state in vector notation

- The steady state in vector notation is simply a vector $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ of probabilities.

Steady state in vector notation

- The steady state in vector notation is simply a vector $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ of probabilities.
- (We use $\vec{\pi}$ to distinguish it from the generic notation $\vec{x}$ for a probability vector.)

Steady state in vector notation

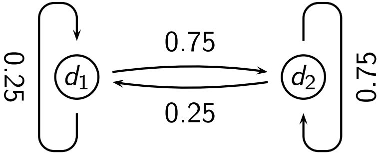
- The steady state in vector notation is simply a vector $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ of probabilities.
- (We use $\vec{\pi}$ to distinguish it from the generic notation $\vec{x}$ for a probability vector.)
- π_i is the long-term visit rate (or PageRank) of page i.

Steady state in vector notation

- The steady state in vector notation is simply a vector $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ of probabilities.
- (We use $\vec{\pi}$ to distinguish it from the generic notation $\vec{x}$ for a probability vector.)
- π_i is the long-term visit rate (or PageRank) of page i.
- So we can think of PageRank as a very long vector – one entry per page.

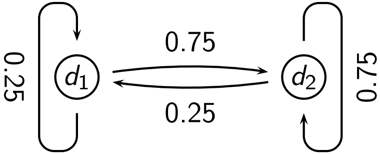
Steady state example

- What is the steady state in this example?



Steady state example

- What is the steady state in this example?



- Solution: $\vec{\pi} = (\pi_1 \ \pi_2) = (0.25 \ 0.75)$

How do we compute the steady state vector?

- In other words: how do we compute PageRank?

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities . . .

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ... and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ... and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.
- But $\vec{\pi}$ is the steady state! So: $\vec{\pi} = \vec{\pi}P$

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ... and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.
- But $\vec{\pi}$ is the steady state! So: $\vec{\pi} = \vec{\pi}P$
- Solving this matrix equation gives us $\vec{\pi}$.

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ... and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.
- But $\vec{\pi}$ is the steady state! So: $\vec{\pi} = \vec{\pi}P$
- Solving this matrix equation gives us $\vec{\pi}$.
- $\vec{\pi}$ is the principal left eigenvector for P ...

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ...and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.
- But $\vec{\pi}$ is the steady state! So: $\vec{\pi} = \vec{\pi}P$
- Solving this matrix equation gives us $\vec{\pi}$.
- $\vec{\pi}$ is the principal left eigenvector for P ...
- ...that is, $\vec{\pi}$ is the left eigenvector with the largest eigenvalue.

How do we compute the steady state vector?

- In other words: how do we compute PageRank?
- Recall: $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$ is the PageRank vector, the vector of steady-state probabilities ...
- ... and if the distribution in this step is described by $\vec{x}$, then the distribution in the next step is distributed as $\vec{x}P$.
- But $\vec{\pi}$ is the steady state! So: $\vec{\pi} = \vec{\pi}P$
- Solving this matrix equation gives us $\vec{\pi}$.
- $\vec{\pi}$ is the principal left eigenvector for P ...
- ... that is, $\vec{\pi}$ is the left eigenvector with the largest eigenvalue.
- Transition probability matrices always have largest eigenvalue 1.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution
- After one step, we're at $\vec{x}P$.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution
- After one step, we're at $\vec{x}P$.
- After two steps, we're at $\vec{x}P^2$.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution
- After one step, we're at $\vec{x}P$.
- After two steps, we're at $\vec{x}P^2$.
- After k steps, we're at $\vec{x}P^k$.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution
- After one step, we're at $\vec{x}P$.
- After two steps, we're at $\vec{x}P^2$.
- After k steps, we're at $\vec{x}P^k$.
- Algorithm: multiply $\vec{x}$ by increasing powers of P until the product looks stable.

One way of computing the PageRank $\vec{\pi}$

- Recall: regardless of where we start (except for pathological cases), we eventually reach the steady state $\vec{\pi}$.
- Start with (almost) any distribution $\vec{x}$, e.g., uniform distribution
- After one step, we're at $\vec{x}P$.
- After two steps, we're at $\vec{x}P^2$.
- After k steps, we're at $\vec{x}P^k$.
- Algorithm: multiply $\vec{x}$ by increasing powers of P until the product looks stable.
- This is called the power method.

Power method: Example

- Two-node example: $\vec{x} = (0.5, 0.5)$, $P = \begin{pmatrix} 0.25 & 0.75 \\ 0.25 & 0.75 \end{pmatrix}$

Power method: Example

- Two-node example: $\vec{x} = (0.5, 0.5)$, $P = \begin{pmatrix} 0.25 & 0.75 \\ 0.25 & 0.75 \end{pmatrix}$
- $\vec{x}P = (0.25, 0.75)$

Power method: Example

- Two-node example: $\vec{x} = (0.5, 0.5)$, $P = \begin{pmatrix} 0.25 & 0.75 \\ 0.25 & 0.75 \end{pmatrix}$
- $\vec{x}P = (0.25, 0.75)$
- $\vec{x}P^2 = (0.25, 0.75)$

Power method: Example

- Two-node example: $\vec{x} = (0.5, 0.5)$, $P = \begin{pmatrix} 0.25 & 0.75 \\ 0.25 & 0.75 \end{pmatrix}$
- $\vec{x}P = (0.25, 0.75)$
- $\vec{x}P^2 = (0.25, 0.75)$
- Convergence in one iteration!

PageRank summary

- Preprocessing

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$
  - $\vec{\pi}_i$ is the PageRank of page i.

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$
  - $\vec{\pi}_i$ is the PageRank of page i.
- Query processing

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$
  - $\vec{\pi}_i$ is the PageRank of page i.
- Query processing
  - Retrieve pages satisfying the query

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$
  - $\vec{\pi}_i$ is the PageRank of page i.
- Query processing
  - Retrieve pages satisfying the query
  - Rank them by their PageRank

PageRank summary

- Preprocessing
  - Given graph of links, build matrix P
  - Apply teleportation
  - From modified matrix, compute $\vec{\pi}$
  - $\vec{\pi}_i$ is the PageRank of page i.
- Query processing
  - Retrieve pages satisfying the query
  - Rank them by their PageRank
  - Return reranked list to the user

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.
  - Consider the query *video service*

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.
  - Consider the query *video service*
  - The Yahoo home page (i) has a very high PageRank and (ii) contains both words.

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.
  - Consider the query *video service*
  - The Yahoo home page (i) has a very high PageRank and (ii) contains both words.
  - If we rank all Boolean hits according to PageRank, then the Yahoo home page would be top-ranked.

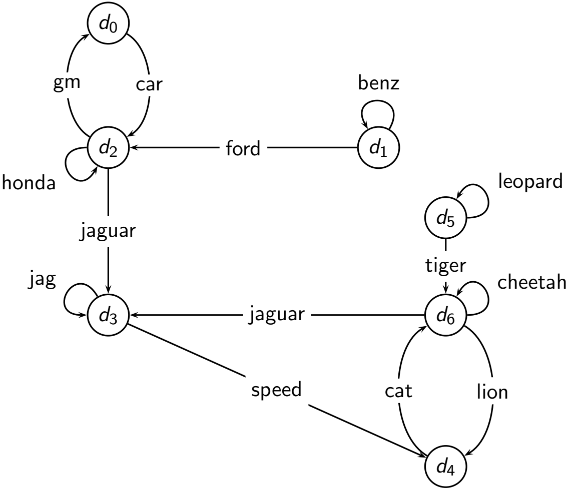
PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.
  - Consider the query *video service*
  - The Yahoo home page (i) has a very high PageRank and (ii) contains both words.
  - If we rank all Boolean hits according to PageRank, then the Yahoo home page would be top-ranked.
  - Clearly not desirable

PageRank issues

- Real surfers are not random surfers – Markov model is not a good model of surfing.
  - Issues: back button, short vs. long paths, bookmarks, directories – and search!
- Simple PageRank ranking (as described on previous slide) produces bad results for many pages.
  - Consider the query *video service*
  - The Yahoo home page (i) has a very high PageRank and (ii) contains both words.
  - If we rank all Boolean hits according to PageRank, then the Yahoo home page would be top-ranked.
  - Clearly not desirable
- In practice: rank according to weighted combination of many factors, including raw text match, anchor text match, PageRank and many other factors

Web graph 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

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

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

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.
- The reality:

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.
- The reality:
  - There are several components that are at least as important: e.g., anchor text, phrases, proximity, tiered indexes . . .

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.
- The reality:
  - There are several components that are at least as important: e.g., anchor text, phrases, proximity, tiered indexes . . .
  - Rumor has it that PageRank in its original form (as presented here) has a negligible impact on ranking!

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.
- The reality:
  - There are several components that are at least as important: e.g., anchor text, phrases, proximity, tiered indexes ...
  - Rumor has it that PageRank in its original form (as presented here) has a negligible impact on ranking!
  - However, variants of a page's PageRank are still an essential part of ranking.

How important is PageRank?

- Frequent claim: PageRank is the most important component of web ranking.
- The reality:
  - There are several components that are at least as important: e.g., anchor text, phrases, proximity, tiered indexes ...
  - Rumor has it that PageRank in its original form (as presented here) has a negligible impact on ranking!
  - However, variants of a page's PageRank are still an essential part of ranking.
  - Adressing link spam is difficult and crucial.

Outline

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.
- Relevance type 1: Hubs. A hub page is a good list of links to pages answering the information need.

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.
- Relevance type 1: Hubs. A hub page is a good list of links to pages answering the information need.
  - Bob's list of recommended hotels in London

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.
- Relevance type 1: Hubs. A hub page is a good list of links to pages answering the information need.
  - Bob's list of recommended hotels in London
- Relevance type 2: Authorities. An authority page is a direct answer to the information need. Authority pages occur repeatedly on hub pages.

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.
- Relevance type 1: Hubs. A hub page is a good list of links to pages answering the information need.
  - Bob's list of recommended hotels in London
- Relevance type 2: Authorities. An authority page is a direct answer to the information need. Authority pages occur repeatedly on hub pages.
  - Home page of Four Seasons Hotel London

HITS – Hyperlink-Induced Topic Search

- Premise: there are two different types of relevance on the web.
- Relevance type 1: Hubs. A hub page is a good list of links to pages answering the information need.
  - Bob's list of recommended hotels in London
- Relevance type 2: Authorities. An authority page is a direct answer to the information need. Authority pages occur repeatedly on hub pages.
  - Home page of Four Seasons Hotel London
- Most approaches to search (including PageRank ranking) don't make the distinction between these two very different types of relevance.

Hubs and authorities

- A good hub page for a topic points to many authority pages for that topic.

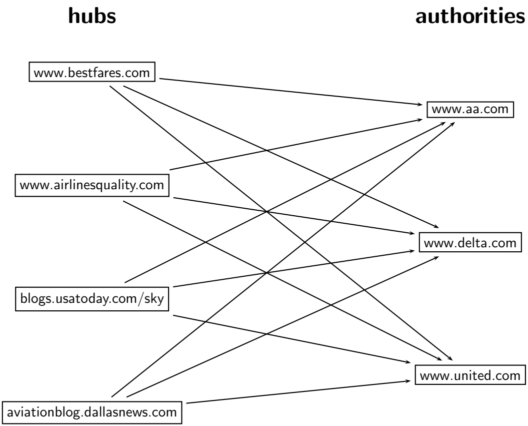
Hubs and authorities

- A good hub page for a topic points to many authority pages for that topic.
- A good authority page for a topic is pointed to by many hub pages for that topic.

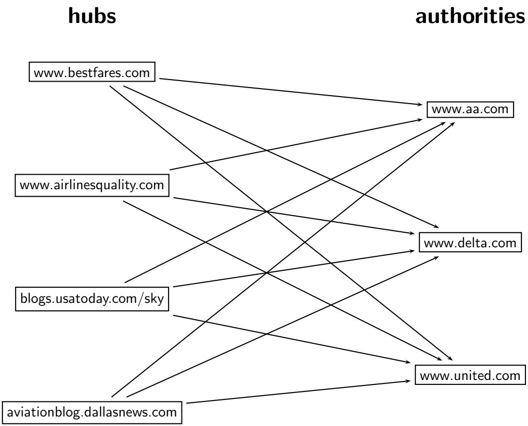
Hubs and authorities

- A good hub page for a topic points to many authority pages for that topic.
- A good authority page for a topic is pointed to by many hub pages for that topic.
- Circular definition – we will turn this into an iterative computation.

Example for hubs and authorities



Example for hubs and authorities



Definition clear?

Root set and base set (1)

- Do a regular web search first

Root set and base set (1)

- Do a regular web search first
- Call the search result the root set

Root set and base set (1)

- 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

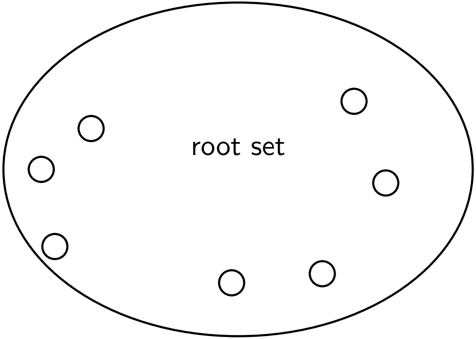
Root set and base set (1)

- 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 this larger set the base set

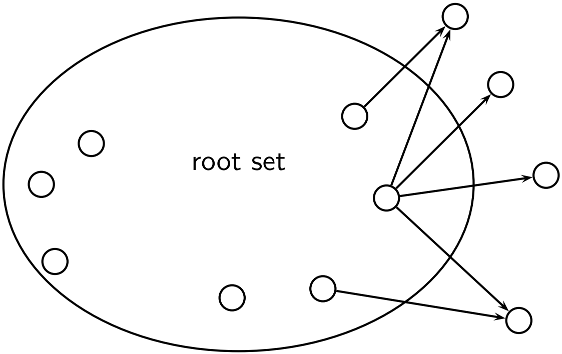
Root set and base set (1)

- 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 this larger set the base set
- Finally, compute hubs and authorities for this (small) web graph

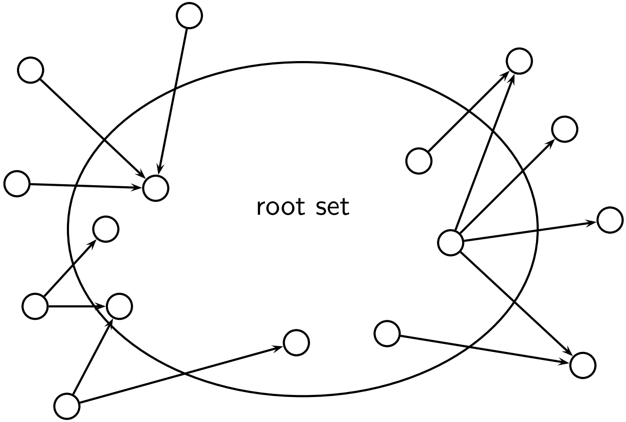
Root set and base set (2)



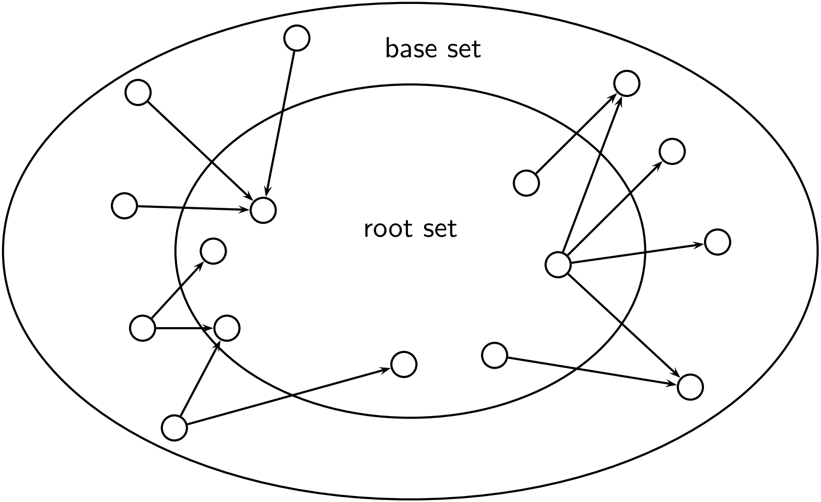
Root set and base set (2)



Root set and base set (2)



Root set and base set (2)



Root set and base set (3)

- Root set typically has 200–1000 nodes.

Root set and base set (3)

- Root set typically has 200–1000 nodes.
- Base set may have up to 5000 nodes.

Root set and base set (3)

- Root set typically has 200–1000 nodes.
- Base set may have up to 5000 nodes.
- Computation of base set:

Root set and base set (3)

- Root set typically has 200–1000 nodes.
- Base set may have up to 5000 nodes.
- Computation of base set:
  - Follow out-links by parsing the pages in the root set

Root set and base set (3)

- Root set typically has 200–1000 nodes.
- Base set may have up to 5000 nodes.
- Computation of base set:
  - Follow out-links by parsing the pages in the root set
  - Find d's in-links by searching for all pages containing a link to d

Root set and base set (3)

- Root set typically has 200–1000 nodes.
- Base set may have up to 5000 nodes.
- Computation of base set:
  - Follow out-links by parsing the pages in the root set
  - Find d's in-links by searching for all pages containing a link to d
  - This assumes that our inverted index supports search for links (in addition to terms).

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$
- Iteratively update all $h(d), a(d)$

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$
- Iteratively update all $h(d), a(d)$
- After convergence:

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$
- Iteratively update all $h(d), a(d)$
- After convergence:
  - Output pages with highest h scores as top hubs

Hub and authority scores

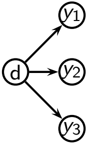
- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$
- Iteratively update all $h(d), a(d)$
- After convergence:
  - Output pages with highest h scores as top hubs
  - Output pages with highest a scores as top authorities

Hub and authority scores

- Compute for each page d in the base set a hub score $h(d)$ and an authority score $a(d)$
- Initialization: for all d: $h(d) = 1$, $a(d) = 1$
- Iteratively update all $h(d), a(d)$
- After convergence:
  - Output pages with highest h scores as top hubs
  - Output pages with highest a scores as top authorities
  - So we output two ranked lists

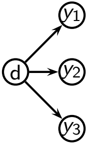
Iterative update

- For all d: $h(d) = \sum_{d \mapsto y} a(y)$

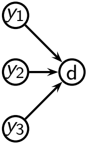


Iterative update

- For all d: $h(d) = \sum_{d \mapsto y} a(y)$

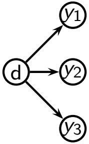


- For all d: $a(d) = \sum_{y \mapsto d} h(y)$

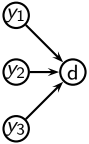


Iterative update

- For all d: $h(d) = \sum_{d \mapsto y} a(y)$



- For all d: $a(d) = \sum_{y \mapsto d} h(y)$



- Iterate these two steps until convergence

Details

- Scaling

Details

- Scaling
  - To prevent the $a()$ and $h()$ values from getting too big, can scale down after each iteration

Details

- Scaling
  - To prevent the $a()$ and $h()$ values from getting too big, can scale down after each iteration
  - Scaling factor doesn't really matter.

Details

- Scaling
  - To prevent the $a()$ and $h()$ values from getting too big, can scale down after each iteration
  - Scaling factor doesn't really matter.
  - We care about the relative (as opposed to absolute) values of the scores.

Details

- Scaling
  - To prevent the $a()$ and $h()$ values from getting too big, can scale down after each iteration
  - Scaling factor doesn't really matter.
  - We care about the relative (as opposed to absolute) values of the scores.
- In most cases, the algorithm converges after a few iterations.

Japan elementary schools

Hubs

- schools
- LINK Page-13
- “ú–{‚ÌŠw□Z
- □a‰„□¬Šw□Zƒz□[ƒ□ƒy□[ƒW
- 100 Schools Home Pages (English)
- K-12 from Japan 10/...rnet and Education)
- http://www...iglobe.ne.jp/~IKESAN
- ‚l‚f‚j□¬Šw□Z‚U”N‚P‘g•¨Œê
- □ÒŠ—’¬—§□ÒŠ—“Œ□¬Šw□Z
- Koulutus ja oppilaitokset
- TOYODA HOMEPAGE
- Education
- Cay's Homepage(Japanese)
- –y“ì□¬Šw□Z‚Ìƒz□[ƒ□ƒy□[ƒW
- UNIVERSITY
- ‰J—³□¬Šw□Z DRAGON97-TOP
- □Â‰ª□¬Šw□Z‚T”N‚P‘gƒz□[ƒ□ƒy□[ƒW
- ¶µ°é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼

Authorities

- The American School in Japan
- The Link Page
- ‰ª□è□s—§ˆä“c□¬Šw□Zƒz□[ƒ□ƒy□[ƒW
- Kids' Space
- ˆÀ□é□s—§ˆÀ□é□¼•”□¬Šw□Z
- ‹{□é‹³ˆç‘åŠw•□‘®□¬Šw□Z
- KEIMEI GAKUEN Home Page (Japanese)
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- □_“Þ□ìŒ§□E‰¡•l□s—§‘†□ì□¼□¬Šw□Z‚Ìƒy
- http://www...p/~m_maru/index.html
- fukui haruyama-es HomePage
- Torisu primary school
- goo
- Yakumo Elementary,Hokkaido,Japan
- FUZOKU Home Page
- Kamishibun Elementary School...

Japan elementary schools

Hubs

- schools
- LINK Page-13
- "ú–{,ÌŠw⊐Z
- ⊐a‰„⊐¬ŠwⅡZƒzⅡ[ƒ⊐ƒy⊐[ƒW
- 100 Schools Home Pages (English)
- K-12 from Japan 10/...rnet and Education)
- http://www...iglobe.ne.jp/~IKESAN
- ,l,f,j⊐¬ŠwⅡZ,U"N,P'g•"Œê
- ⊐ÒŠ—'¬—§⊐ÒŠ—"Œ⊐¬ŠwⅡZ
- Koulutus ja oppilaitokset
- TOYODA HOMEPAGE
- Education
- Cay's Homepage(Japanese)
- –y"ì⊐¬ŠwⅡZ,Ìƒz⊐[ƒ⊐ƒy⊐[ƒW
- UNIVERSITY
- ‰J—³⊐¬ŠwⅡZ DRAGON97-TOP
- ⊐Â‰ª⊐¬ŠwⅡZ,T"N,P'gƒz⊐[ƒ⊐ƒy⊐[ƒW
- ¶µ°é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼

Authorities

- The American School in Japan
- The Link Page
- ‰ª⊐è⊐s—§ˆä'c⊐¬ŠwⅡZƒz⊐[ƒ⊐ƒy⊐[ƒW
- Kids' Space
- ˆÀ⊐é⊐s—§ˆÀ⊐é⊐¼•"⊐¬ŠwⅡZ
- ‹{⊐é‹³ˆç'åŠw•⊐'®⊐¬ŠwⅡZ
- KEIMEI GAKUEN Home Page (Japanese)
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- ⊐_"Þ⊐ìŒ§⊐E‰¡•l⊐s—§'†⊐ì⊐¼⊐¬ŠwⅡZ,Ìƒy
- http://www...p/~m_maru/index.html
- fukui haruyama-es HomePage
- Torisu primary school
- goo
- Yakumo Elementary,Hokkaido,Japan
- FUZOKU Home Page
- Kamishibun Elementary School...

- The query was "Japan elementary schools".

Japan elementary schools

Hubs	Authorities
schools	The American School in Japan
LINK Page-13	The Link Page
"ú–{,ÌŠw□Z	‰ª□è□s—§ˆä"c□¬Šw□Zƒz□[ƒ□ƒy□[ƒW
□a‰„□¬Šw□Zƒz□[ƒ□ƒy□[ƒW	Kids' Space
100 Schools Home Pages (English)	ˆÀ□é□s—§ˆÀ□é□¼•"□¬Šw□Z
K-12 from Japan 10/...rnet and Education)	‹{□é‹³ˆç'åŠw•□'®□¬Šw□Z
http://www...iglobe.ne.jp/~IKESAN	KEIMEI GAKUEN Home Page (Japanese)
,l,f,j□¬Šw□Z,U"N,P'g•¨Œê	Shiranuma Home Page
□ÒŠ—'¬—§□ÒŠ—"Œ□¬Šw□Z	fuzoku-es.fukui-u.ac.jp
Koulutus ja oppilaitokset	welcome to Miasa E&J school
TOYODA HOMEPAGE	□_"Þ□ìŒ§□E‰¡•l□s—§'†□ì□¼□¬Šw□Z,Ìƒy
Education	http://www...p/~m_maru/index.html
Cay's Homepage(Japanese)	fukui haruyama-es HomePage
–y"ì□¬Šw□Z,Ìƒz□[ƒ□ƒy□[ƒW	Torisu primary school
UNIVERSITY	goo
‰J—³□¬Šw□Z DRAGON97-TOP	Yakumo Elementary,Hokkaido,Japan
□Â‰ª□¬Šw□Z,T"N,P'gƒz□[ƒ□ƒy□[ƒW	FUZOKU Home Page
¶µ°é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼	Kamishibun Elementary School...

- The query was "Japan elementary schools".
- HITS pulled together good pages regardless of page content.

Japan elementary schools

Hubs	Authorities
schools	The American School in Japan
LINK Page-13	The Link Page
“ú–{‚ÌŠw□Z	‰ª□è□s—§ˆä“c□¬Šw□Zƒz□[ƒ□ƒy□[ƒW
□a‰„□¬Šw□Zƒz□[ƒ□ƒy□[ƒW	Kids' Space
100 Schools Home Pages (English)	ˆÀ□é□s—§ˆÀ□é□¼•”□¬Šw□Z
K-12 from Japan 10/...rnet and Education)	‹{□é‹³ˆç‘åŠw•□‘®□¬Šw□Z
http://www...iglobe.ne.jp/~IKESAN	KEIMEI GAKUEN Home Page (Japanese)
‚l‚f‚j□¬Šw□Z‚U”N‚P‘g•¨Œê	Shiranuma Home Page
□ÒŠ—’¬—§□ÒŠ—“Œ□¬Šw□Z	fuzoku-es.fukui-u.ac.jp
Koulutus ja oppilaitokset	welcome to Miasa E&J school
TOYODA HOMEPAGE	□_“Þ□ìŒ§□E‰¡•l□s—§‘†□ì□¼□¬Šw□Z‚Ìƒy
Education	http://www...p/~m_maru/index.html
Cay's Homepage(Japanese)	fukui haruyama-es HomePage
–y“ì□¬Šw□Z‚Ìƒz□[ƒ□ƒy□[ƒW	Torisu primary school
UNIVERSITY	goo
‰J—³□¬Šw□Z DRAGON97-TOP	Yakumo Elementary,Hokkaido,Japan
□Â‰ª□¬Šw□Z‚T”N‚P‘gƒz□[ƒ□ƒy□[ƒW	FUZOKU Home Page
¶µ°é¼ÂÁ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼	Kamishibun Elementary School...

- The query was “Japan elementary schools”.
- HITS pulled together good pages regardless of page content.
- An English query was able to retrieve Japanese-language pages!

Japan elementary schools

Hubs

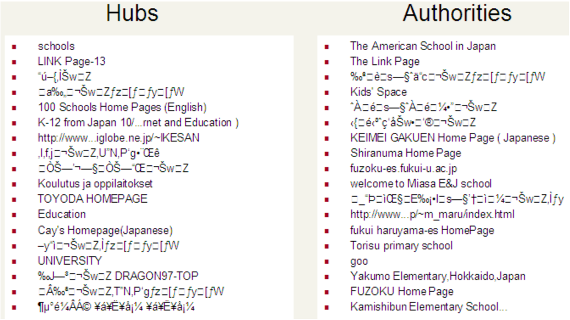
- schools
- LINK Page-13
- "ú–{,ÌŠw⊐Z
- ⊐a‰„⊐¬ŠwⅡZfz⊐[ƒ⊐ƒy⊐[ƒW
- 100 Schools Home Pages (English)
- K-12 from Japan 10/...rnet and Education)
- http://www...iglobe.ne.jp/~IKESAN
- ,l,f,j⊐¬Šw⊐Z,U"N,P'g•"Œê
- ⊐ÒŠ—'¬—§⊐ÒŠ—"Œ⊐¬Šw⊐Z
- Koulutus ja oppilaitokset
- TOYODA HOMEPAGE
- Education
- Cay's Homepage(Japanese)
- –y"ì⊐¬Šw⊐Z,Ìƒz⊐[ƒ⊐ƒy⊐[ƒW
- UNIVERSITY
- ‰J—³⊐¬Šw⊐Z DRAGON97-TOP
- ⊐Â‰ª⊐¬Šw⊐Z,T"N,P'gƒz⊐[ƒ⊐ƒy⊐[ƒW
- ¶µ°é¼ÂÀ© ¥á¥Ë¥å¡¼ ¥á¥Ë¥å¡¼

Authorities

- The American School in Japan
- The Link Page
- ‰ª⊐è⊐s—§ˆä'c⊐¬Šw⊐Zƒz⊐[ƒ⊐ƒy⊐[ƒW
- Kids' Space
- ˆÀ⊐é⊐s—§ˆÀ⊐é⊐¼•"⊐¬Šw⊐Z
- ‹{⊐é‹³ˆç'åŠw•⊐'®⊐¬Šw⊐Z
- KEIMEI GAKUEN Home Page (Japanese)
- Shiranuma Home Page
- fuzoku-es.fukui-u.ac.jp
- welcome to Miasa E&J school
- ⊐_"Þ⊐ìŒ§⊐E‰¡•l⊐s—§'†⊐ì⊐¼⊐¬Šw⊐Z,Ìƒy
- http://www...p/~m_maru/index.html
- fukui haruyama-es HomePage
- Torisu primary school
- goo
- Yakumo Elementary,Hokkaido,Japan
- FUZOKU Home Page
- Kamishibun Elementary School...

- The query was "Japan elementary schools".
- HITS pulled together good pages regardless of page content.
- An English query was able to retrieve Japanese-language pages!
- Once the base set is assembled, we only do link analysis, no text matching.

Japan elementary schools



- The query was "Japan elementary schools".
- HITS pulled together good pages regardless of page content.
- An English query was able to retrieve Japanese-language pages!
- Once the base set is assembled, we only do link analysis, no text matching.
- Danger: topic drift – the pages found by following links may not be related to the original query.

Proof of convergence

Proof of convergence

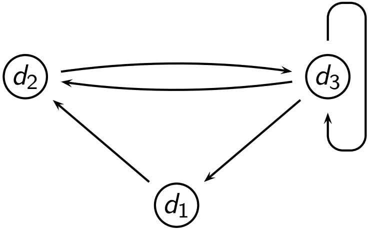
- We define an $N \times N$ adjacency matrix A.

Proof of convergence

- We define an $N \times N$ adjacency matrix A.
- For $1 \leq i, j \leq N$, the matrix entry A_{ij} tells us whether there is a link from page i to page j ($A_{ij} = 1$) or not ($A_{ij} = 0$).

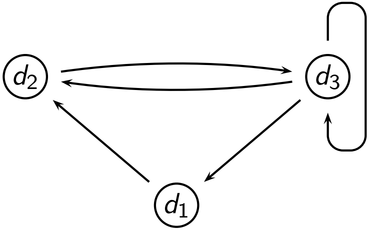
Proof of convergence

- We define an $N \times N$ adjacency matrix A.
- For $1 \leq i, j \leq N$, the matrix entry A_{ij} tells us whether there is a link from page i to page j ($A_{ij} = 1$) or not ($A_{ij} = 0$).
- Example:



Proof of convergence

- We define an $N \times N$ adjacency matrix A.
- For $1 \leq i, j \leq N$, the matrix entry A_{ij} tells us whether there is a link from page i to page j ($A_{ij} = 1$) or not ($A_{ij} = 0$).
- Example:



	d_1	d_2	d_3
d_1	0	1	0
d_2	1	1	1
d_3	1	0	0

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...
- ... and we can write $a(d) = \sum_{y \mapsto d} h(y)$ as $\vec{a} = A^T \vec{h}$

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...
- ... and we can write $a(d) = \sum_{y \mapsto d} h(y)$ as $\vec{a} = A^T\vec{h}$
- HITS algorithm in matrix notation:

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \dots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...
- ... and we can write $a(d) = \sum_{y \mapsto d} h(y)$ as $\vec{a} = A^T \vec{h}$
- HITS algorithm in matrix notation:
  - Compute $\vec{h} = A\vec{a}$

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...
- ... and we can write $a(d) = \sum_{y \mapsto d} h(y)$ as $\vec{a} = A^T\vec{h}$
- HITS algorithm in matrix notation:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$

Write update rules as matrix operations

- Define the hub vector $\vec{h} = (h_1, \ldots, h_N)$ as the vector of hub scores. h_i is the hub score of page d_i.
- Similarly for $\vec{a}$, the vector of authority scores
- Now we can write $h(d) = \sum_{d \mapsto y} a(y)$ as a matrix operation: $\vec{h} = A\vec{a}$ ...
- ... and we can write $a(d) = \sum_{y \mapsto d} h(y)$ as $\vec{a} = A^T \vec{h}$
- HITS algorithm in matrix notation:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T \vec{h}$
  - Iterate until convergence

HITS as eigenvector problem

- HITS algorithm in matrix notation. Iterate:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$

HITS as eigenvector problem

- HITS algorithm in matrix notation. Iterate:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$
- By substitution we get: $\vec{h} = AA^T\vec{h}$ and $\vec{a} = A^TA\vec{a}$

HITS as eigenvector problem

- HITS algorithm in matrix notation. Iterate:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$
- By substitution we get: $\vec{h} = AA^T\vec{h}$ and $\vec{a} = A^TA\vec{a}$
- Thus, $\vec{h}$ is an eigenvector of AA^T and $\vec{a}$ is an eigenvector of A^TA.

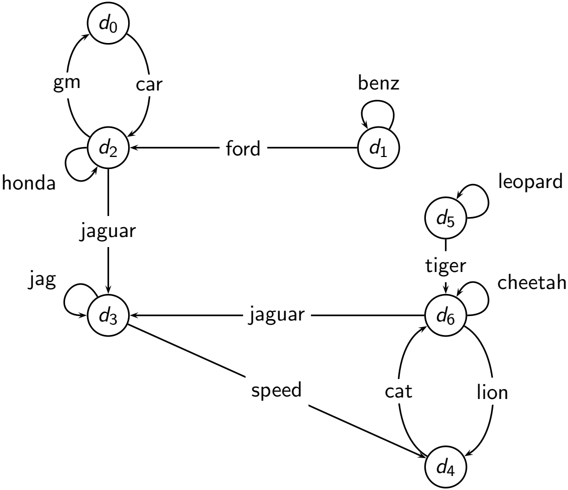
HITS as eigenvector problem

- HITS algorithm in matrix notation. Iterate:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$
- By substitution we get: $\vec{h} = AA^T\vec{h}$ and $\vec{a} = A^TA\vec{a}$
- Thus, $\vec{h}$ is an eigenvector of AA^T and $\vec{a}$ is an eigenvector of A^TA.
- So the HITS algorithm is actually a special case of the power method and hub and authority scores are eigenvector values.

HITS as eigenvector problem

- HITS algorithm in matrix notation. Iterate:
  - Compute $\vec{h} = A\vec{a}$
  - Compute $\vec{a} = A^T\vec{h}$
- By substitution we get: $\vec{h} = AA^T\vec{h}$ and $\vec{a} = A^TA\vec{a}$
- Thus, $\vec{h}$ is an eigenvector of AA^T and $\vec{a}$ is an eigenvector of A^TA.
- So the HITS algorithm is actually a special case of the power method and hub and authority scores are eigenvector values.
- HITS and PageRank both formalize link analysis as eigenvector problems.

Example web graph



Raw matrix H 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

Hub vectors $h_0, \vec{h}_i = \frac{1}{d_i} H \cdot \vec{a}_i, i \geq 1$

	$\vec{h}_0$	$\vec{h}_1$	$\vec{h}_2$	$\vec{h}_3$	$\vec{h}_4$	$\vec{h}_5$
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

Authority vectors $\vec{a}_i = \frac{1}{c_i} H^T \cdot \vec{h}_{i-1}, i \geq 1$

	$\vec{a}_1$	$\vec{a}_2$	$\vec{a}_3$	$\vec{a}_4$	$\vec{a}_5$	$\vec{a}_6$	$\vec{a}_7$
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

Top-ranked pages

- Pages with highest in-degree: d_2, d_3, d_6

Top-ranked pages

- Pages with highest in-degree: d_2, d_3, d_6
- Pages with highest out-degree: d_2, d_6

Top-ranked pages

- Pages with highest in-degree: d_2, d_3, d_6
- Pages with highest out-degree: d_2, d_6
- Pages with highest PageRank: d_6

Top-ranked pages

- Pages with highest in-degree: d_2, d_3, d_6
- Pages with highest out-degree: d_2, d_6
- Pages with highest PageRank: d_6
- Pages with highest hub score: d_6 (close: d_2)

Top-ranked pages

- Pages with highest in-degree: d_2, d_3, d_6
- Pages with highest out-degree: d_2, d_6
- Pages with highest PageRank: d_6
- Pages with highest hub score: d_6 (close: d_2)
- Pages with highest authority score: d_3

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
    - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.
    - We could also apply HITS to the entire web and PageRank to a small base set.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
    - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.
    - We could also apply HITS to the entire web and PageRank to a small base set.
- On the web, a good hub almost always is also a good authority.

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.
  - We could also apply HITS to the entire web and PageRank to a small base set.
- On the web, a good hub almost always is also a good authority.
- Why?

PageRank vs. HITS: Discussion

- PageRank can be precomputed, HITS has to be computed at query time.
  - HITS is too expensive in most application scenarios.
- The PageRank and HITS make two different design choices concerning (i) the eigenproblem formalization (ii) the set of pages to apply the formalization to.
- These two are orthogonal.
  - We could also apply HITS to the entire web and PageRank to a small base set.
- On the web, a good hub almost always is also a good authority.
- Why?
- The actual difference between PageRank ranking and HITS ranking is therefore not as large as one might expect.

Resources

- Chapter 21 of IIR

Resources

- Chapter 21 of IIR
- Resources at `http://ifnlp.org/ir`

Resources

- Chapter 21 of IIR
- Resources at `http://ifnlp.org/ir`
- American Mathematical Society article on PageRank (popular science style)

Resources

- Chapter 21 of IIR
- Resources at `http://ifnlp.org/ir`
- American Mathematical Society article on PageRank (popular science style)
- Jon Kleinberg's home page (main person behind HITS)

Resources

- Chapter 21 of IIR
- Resources at `http://ifnlp.org/ir`
- American Mathematical Society article on PageRank (popular science style)
- Jon Kleinberg's home page (main person behind HITS)
- Google's official description of PageRank: *PageRank reflects our view of the importance of web pages by considering more than 500 million variables and 2 billion terms. Pages that we believe are important pages receive a higher PageRank and are more likely to appear at the top of the search results.*