Link Analysis

Paolo Boldi
DSI
LAW (Laboratory for Web Algorithmics)
Università degli Studi di Milan

Ranking, search engines, social networks

Ranking is of uttermost importance in IR, search engines and also in other social networks (e.g., facebook):

Ranking, search engines, social networks

Ranking is of uttermost importance in IR, search engines and also in other social networks (e.g., facebook):

- Choosing which of your friends' signals are relevant for you?

Ranking, search engines, social networks

Ranking is of uttermost importance in IR, search engines and also in other social networks (e.g., facebook):

- Choosing which of your friends' signals are relevant for you?
- Choosing which of your non-friends should be suggested as new contact?

Ranking, search engines, social networks

Ranking is of uttermost importance in IR, search engines and also in other social networks (e.g., facebook):

- Choosing which of your friends' signals are relevant for you?
- Choosing which of your non-friends should be suggested as new contact?

In traditional information retrieval, ranking is typically realized through a scoring system:

$$\sigma : \mathcal{D} \times \mathcal{Q} \to \mathbf{R}$$

that assigns a "relevance" score to every document/query pair.

Ranking, search engines, social networks

Ranking is of uttermost importance in IR, search engines and also in other social networks (e.g., facebook):

- Choosing which of your friends' signals are relevant for you?
- Choosing which of your non-friends should be suggested as new contact?

In traditional information retrieval, ranking is typically realized through a scoring system:

$$\sigma : \mathcal{D} \times \mathcal{Q} \to \mathbf{R}$$

that assigns a "relevance" score to every document/query pair.

Rankings may be composed (e.g., by linear combination): this is called *rank aggregation*.

What happens when a search engine receives a certain query q from a user?

Web Search

What happens when a search engine receives a certain query q from a user?

- *Selection*: it selects, from the set D of all available documents, a subset $S(q)$ of documents that satisfy q;

Web Search

What happens when a search engine receives a certain query q from a user?

- *Selection*: it selects, from the set D of all available documents, a subset $S(q)$ of documents that satisfy q;
- *Ranking*: it establishes a total order on $S(q)$ determining how the results should be presented to the user.

Ranking

A most crucial step!

Ranking

A most crucial step!

- Typical queries have just one or two words (recent survey says 3.08 on average), and million of results

Ranking

A most crucial step!

- Typical queries have just one or two words (recent survey says 3.08 on average), and million of results
- The typical user just looks at the first result page; often, just the first link (*Feeling lucky*)

Ranking

A most crucial step!

- Typical queries have just one or two words (recent survey says 3.08 on average), and million of results
- The typical user just looks at the first result page; often, just the first link (*Feeling lucky*)
- In the past, a search engine's share of market used to depend on freshness, usability, coverage, additional features. . .

Ranking

A most crucial step!

- Typical queries have just one or two words (recent survey says 3.08 on average), and million of results
- The typical user just looks at the first result page; often, just the first link (*Feeling lucky*)
- In the past, a search engine's share of market used to depend on freshness, usability, coverage, additional features...
- ...now, the competitive edge is determined mostly by ranking!

Link Analysis — Problem and assumptions

- **Static Ranking problem:** Assign to each web page a score that is proportional to its importance.

Link Analysis — Problem and assumptions

- **Static Ranking problem:** Assign to each web page a score that is proportional to its importance. Use only linkage structure to this aim.

Link Analysis — Problem and assumptions

- **Static Ranking problem:** Assign to each web page a score that is proportional to its importance. Use only linkage structure to this aim.
- **Basic assumption:** A link is a way to confer importance.

The Web as a Graph

You can think of the Web as a (directed) graph:

The Web as a Graph

You can think of the Web as a (directed) graph:

- its nodes are the URLs

The Web as a Graph

You can think of the Web as a (directed) graph:

- its nodes are the URLs
- there is an arc from node x to node y iff the page with URL x contains a hyperlink towards URL y.

The Web as a Graph

You can think of the Web as a (directed) graph:

- its nodes are the URLs
- there is an arc from node x to node y iff the page with URL x contains a hyperlink towards URL y.

This is called the *Web graph*.

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)
- about 3 000/5 000 billions of arcs

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)
- about 3 000/5 000 billions of arcs
- about $800/1,000$ millions of hosts

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)
- about 3 000/5 000 billions of arcs
- about 800/1, 000 millions of hosts
- about 2,500 millions of users.

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)
- about 3 000/5 000 billions of arcs
- about 800/1,000 millions of hosts
- about 2,500 millions of users.

Dark mass = pages that cannot be reached: according to prudent estimates, currently indexed web is just the 16-20% of the "real" web!

How Large Is It?

It is *extremely* difficult to evaluate the Web size!

- about 30/45 billions of nodes (excluding the *dark mass*)
- about 3 000/5 000 billions of arcs
- about 800/1,000 millions of hosts
- about 2,500 millions of users.

Dark mass = pages that cannot be reached: according to prudent estimates, currently indexed web is just the 16-20% of the "real" web!

BE CAREFUL! Our knowledge of the Web is only indirect! (it is what we can collect using web crawlers, or interrogating search engines. . .)

What Is the Shape of the Web?

- The (known part of the) Web graph has a bowtie-like shape. . .
- Subdomains (e.g., country-code domains [`.fr`, `.it`, . . .], large intranets etc.) are similar: a fractal-like graph
- It is *highly dynamic*

What Is the Shape of the Web?

Giant component (core)

- A strongly-connected component containing about 30% of the pages
- Estimated diameter: directed $\to$ 20/30; undirected $\to$ 10/17
- "It's a small-world!"

Left-hand side

- Ancestors of the giant component in the scc DAG
- About 25%
- Estimating its size is very difficult (how can you get there???)
- They are the Web pariahs: they are there, but no one wants to link them!

Right-hand side

- Descendants of the giant component in the scc DAG
- About 25%
- Contains all "documents" with no hyperlinks in them (e.g., pure text, word/PDF/PostScript with no links etc.)

What Is the Shape of the Web?

Tubes, tendrils and isolated components

- Remaining nodes (20%)

Ranking Techniques: A Taxonomy

Depending on whether the scoring (ranking) function depends or not on the query, and whether it depends or not on the text of the page (or only on its links):

Ranking Techniques: A Taxonomy

Depending on whether the scoring (ranking) function depends or not on the query, and whether it depends or not on the text of the page (or only on its links):

	Query-dependent (dynamic)	**Query-independent (static)**
Text-based	IR (already treated)	-
Link-based	e.g., HITS	e.g., PageRank

Centrality in networks

Query-independent link-based techniques are methods that assign an *importance* to the nodes of a graph based on their position in the graph itself. This is often referred to as *graph centrality*

Centrality in networks

Query-independent link-based techniques are methods that assign an *importance* to the nodes of a graph based on their position in the graph itself. This is often referred to as *graph centrality*

Graph centrality is a topic of uttermost importance in social sciences. Briefly recalled in the next few slides.

History of centrality (in a nutshell)

History of centrality (in a nutshell)

- first attempts in the late 1940s at M.I.T. (Bavelas(, in the framework of communication patterns and group collaboration;

History of centrality (in a nutshell)

- first attempts in the late 1940s at M.I.T. (Bavelas(, in the framework of communication patterns and group collaboration;
- in the following decades, various measures of centralities were proposed and employed by social scientists in a myriad of contexts

History of centrality (in a nutshell)

- first attempts in the late 1940s at M.I.T. (Bavelas(, in the framework of communication patterns and group collaboration;
- in the following decades, various measures of centralities were proposed and employed by social scientists in a myriad of contexts
- a new interest raised in the mid-90s with the advent of search engines: a “reincarnation” of centrality.

History of centrality (in a nutshell)

- first attempts in the late 1940s at M.I.T. (Bavelas(, in the framework of communication patterns and group collaboration;
- in the following decades, various measures of centralities were proposed and employed by social scientists in a myriad of contexts
- a new interest raised in the mid-90s with the advent of search engines: a "reincarnation" of centrality.

Freeman (1979) observed:

> *"several measures are often only vaguely related to the intuitive ideas they purport to index, and many are so complex that it is difficult or impossible to discover what, if anything, they are measuring"*

Types of centralities

Starting point: the central node of a star is the most important!
Why?

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;
4. the node with the largest number of incoming paths of length k, for every k;

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;
4. the node with the largest number of incoming paths of length k, for every k;
5. the node that maximizes the dominant eigenvector of the graph matrix;

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;
4. the node with the largest number of incoming paths of length k, for every k;
5. the node that maximizes the dominant eigenvector of the graph matrix;
6. the node with highest probability in the stationary distribution of the natural random walk on the graph.

Types of centralities

Starting point: the central node of a star is the most important! Why?

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;
4. the node with the largest number of incoming paths of length k, for every k;
5. the node that maximizes the dominant eigenvector of the graph matrix;
6. the node with highest probability in the stationary distribution of the natural random walk on the graph.

These observations lead to corresponding competing views of centrality.

Types of centralities

This observation leads to the following classes of indices of centrality:

Types of centralities

This observation leads to the following classes of indices of centrality:

1. measures based on degree [degree, SALSA];

Types of centralities

This observation leads to the following classes of indices of centrality:

1. measures based on degree [degree, SALSA];
2. measures based on paths [betweenness, Katz's index];

Types of centralities

This observation leads to the following classes of indices of centrality:

1. measures based on degree [degree, SALSA];
2. measures based on paths [betweenness, Katz's index];
3. measures based on distances [closeness, Lin's index];

Types of centralities

This observation leads to the following classes of indices of centrality:

1. measures based on degree [degree, SALSA];
2. measures based on paths [betweenness, Katz's index];
3. measures based on distances [closeness, Lin's index];
4. measures based on spectral measures [dominant eigenvector, Seeley's index, PageRank, HITS].

PageRank [Brin, Page, 1998]

An extremely popular ranking technique, because. . .

PageRank [Brin, Page, 1998]

An extremely popular ranking technique, because...

- it is static, so it can be computed beforehand (not at query time)

PageRank [Brin, Page, 1998]

An extremely popular ranking technique, because. . .

- it is static, so it can be computed beforehand (not at query time)
- it can be computed efficiently

PageRank [Brin, Page, 1998]

An extremely popular ranking technique, because...

- it is static, so it can be computed beforehand (not at query time)
- it can be computed efficiently
- it is (used to be) the main ranking technique used at Google.

PageRank — An introductory metaphor (1)

- Every page has an amount of money that, at the end of the game, will be proportional to its importance.

PageRank — An introductory metaphor (1)

- Every page has an amount of money that, at the end of the game, will be proportional to its importance.
- At the beginning, everybody has the same amount of money.

PageRank — An introductory metaphor (1)

- Every page has an amount of money that, at the end of the game, will be proportional to its importance.
- At the beginning, everybody has the same amount of money.
- At every step, every page x gives away all of its money, redistributing it equally among its out-neighbors.

PageRank — An introductory metaphor (1)

- Every page has an amount of money that, at the end of the game, will be proportional to its importance.
- At the beginning, everybody has the same amount of money.
- At every step, every page x gives away all of its money, redistributing it equally among its out-neighbors.

Problem with this solution: Formation of oligopolies that "suck away" all money from the system, without ever giving it back.

PageRank — An introductory metaphor (2)

- At every step, only a fixed fraction $\alpha < 1$ of the money a page has is redistributed to its neighbors; the remaining fraction $1 - \alpha$ is paid to the state (a form of taxation).

PageRank — An introductory metaphor (2)

- At every step, only a fixed fraction $\alpha < 1$ of the money a page has is redistributed to its neighbors; the remaining fraction $1 - \alpha$ is paid to the state (a form of taxation).
- The state redistributes the money collected to all nodes, according to a certain *preference vector v* (e.g., the uniform distribution, the "Berlusconi" distribution...).

PageRank — An introductory metaphor (2)

- At every step, only a fixed fraction $\alpha < 1$ of the money a page has is redistributed to its neighbors; the remaining fraction $1 - \alpha$ is paid to the state (a form of taxation).
- The state redistributes the money collected to all nodes, according to a certain *preference vector* v (e.g., the uniform distribution, the "Berlusconi" distribution...).

Another problem: What should the dangling nodes do? (A *dangling node* is one that has no out-neighbors)

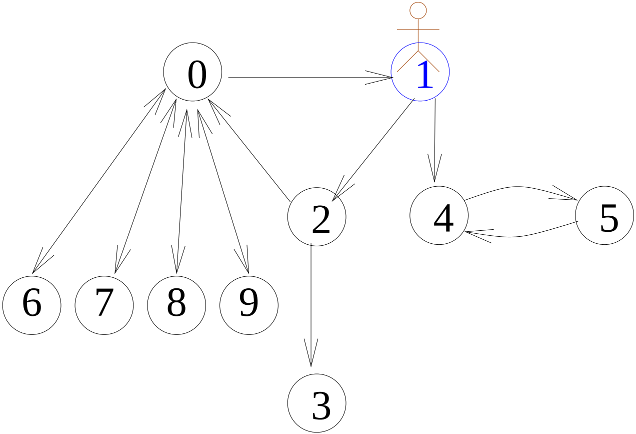
PageRank — An introductory metaphor (2)

- At every step, only a fixed fraction $\alpha < 1$ of the money a page has is redistributed to its neighbors; the remaining fraction $1 - \alpha$ is paid to the state (a form of taxation).
- The state redistributes the money collected to all nodes, according to a certain *preference vector v* (e.g., the uniform distribution, the "Berlusconi" distribution. . .).

Another problem: What should the dangling nodes do? (A *dangling node* is one that has no out-neighbors)

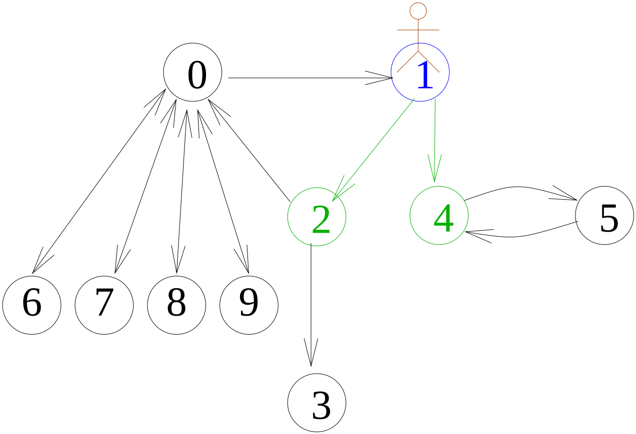
Dangling nodes pay, as every other node, $1 - \alpha$ in taxes, and distribute α to the nodes according to a fixed *dangling-node distribution u*.

PageRank: the Web-Surfer Metaphor



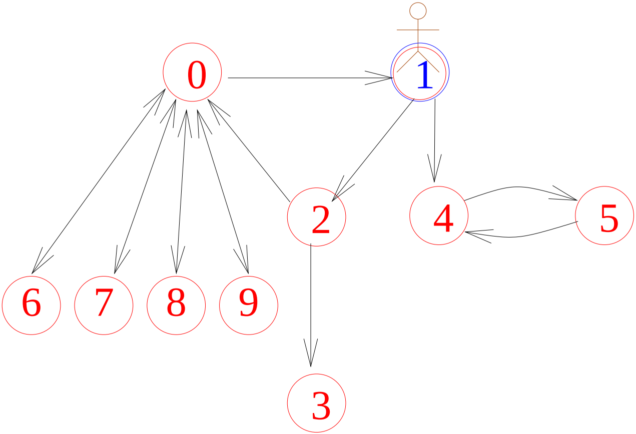
A surfer is wandering about the web. . .

PageRank: the Web-Surfer Metaphor



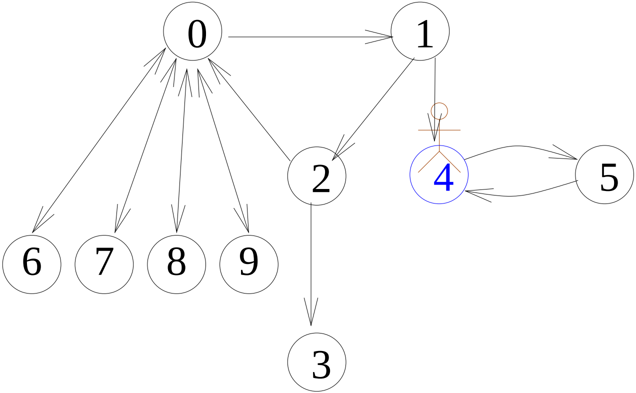
At each step, with probability α (s)he chooses the next page by clicking on a random link. . .

PageRank: the Web-Surfer Metaphor

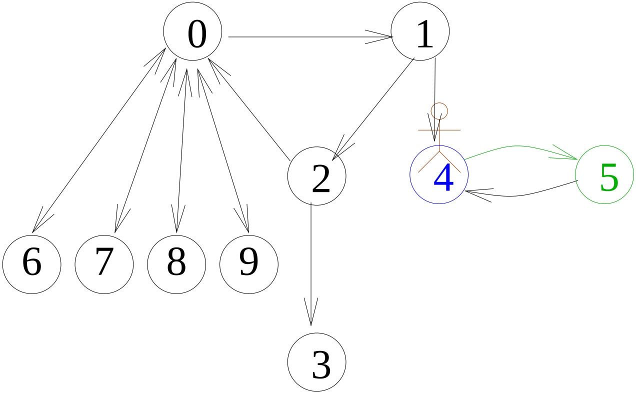


...with probability $1 - \alpha$, (s)he jumps to a random node (chosen uniformly or according to a fixed distribution, the *preference vector*)

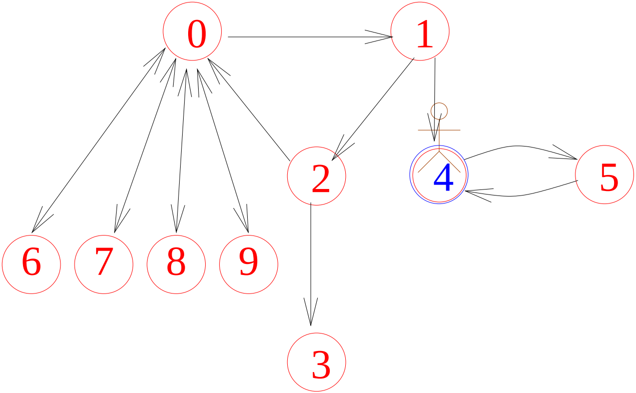
PageRank: the Web-Surfer Metaphor



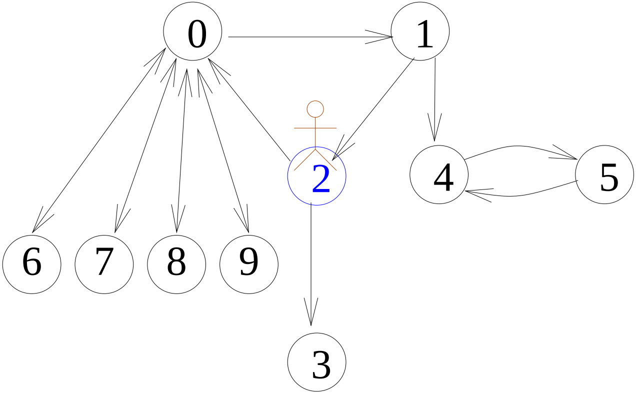
PageRank: the Web-Surfer Metaphor



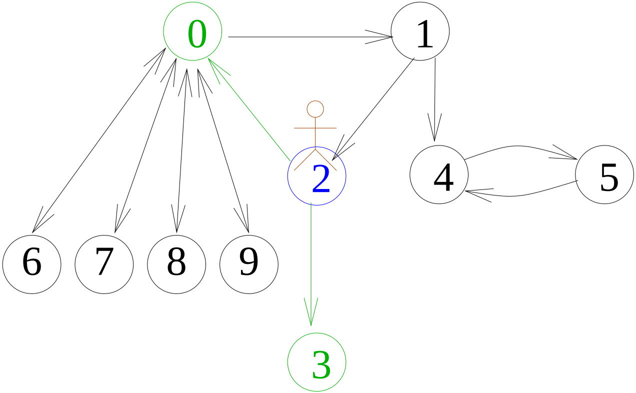
PageRank: the Web-Surfer Metaphor



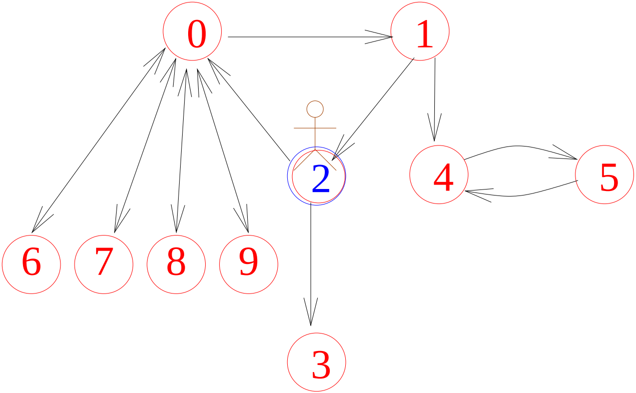
PageRank: the Web-Surfer Metaphor



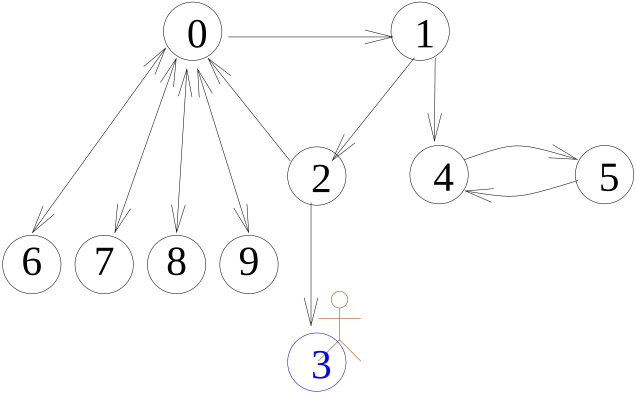
PageRank: the Web-Surfer Metaphor



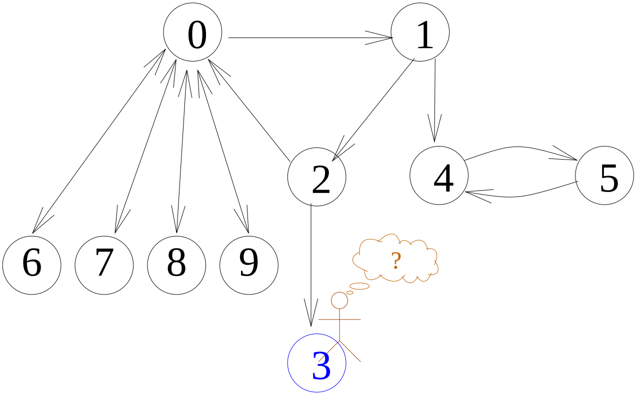
PageRank: the Web-Surfer Metaphor



PageRank: the Web-Surfer Metaphor

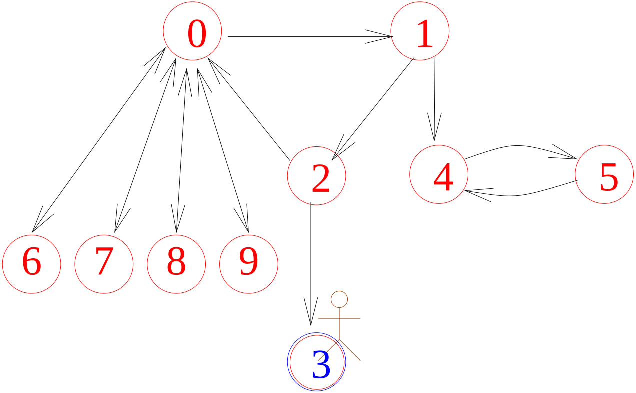


PageRank: the Web-Surfer Metaphor



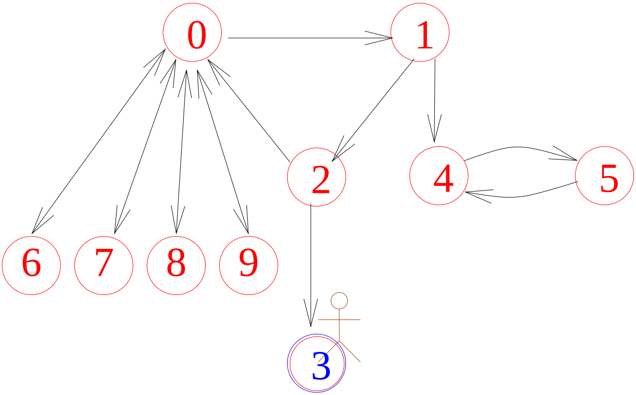
What if (s)he reaches a node with no outlinks (a *dangling node*)?

PageRank: the Web-Surfer Metaphor



In that case, (s)he jumps to a random node *with probability 1*.

PageRank: the Web-Surfer Metaphor



The PageRank of a page is the average fraction of time spent by the surfer on that page.

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

- the *web graph* G;

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

- the *web graph* G;
- the *preference vector* v;

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

- the *web graph* G;
- the *preference vector* v;
- the *dangling-node distribution* u;

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

- the *web graph* G;
- the *preference vector* v;
- the *dangling-node distribution* u;
- the *damping factor* α.

What does PageRank depends on?

PageRank can be formally defined as the limit distribution of a stochastic process whose states are Web pages.

What does this distribution depend on? (more on all this later)

- the *web graph* G;
- the *preference vector* v;
- the *dangling-node distribution* u;
- the *damping factor* α.

How does PageRank depends on each of these factors? What happens at limit values (e.g., $\alpha \to 1$)?

PageRank: formal definition

PageRank: formal definition

- Is the definition of PageRank well-given? Are we all using the same definition?

PageRank: formal definition

- Is the definition of PageRank well-given? Are we all using the same definition?
- The *row-normalised matrix* of a (web) graph G is the matrix $\bar{G}$ such that $(\bar{G})_{ij}$ is one over the outdegree of i if there is an arc from i to j in G (in general, and usually, not stochastic because of rows of zeroes).

PageRank: formal definition

- Is the definition of PageRank well-given? Are we all using the same definition?
- The *row-normalised matrix* of a (web) graph G is the matrix $\bar{G}$ such that $(\bar{G})_{ij}$ is one over the outdegree of i if there is an arc from i to j in G (in general, and usually, not stochastic because of rows of zeroes).
- d is the characteristic vector of dangling nodes (nodes without outgoing arcs).

PageRank: formal definition

- Is the definition of PageRank well-given? Are we all using the same definition?
- The *row-normalised matrix* of a (web) graph G is the matrix $\bar{G}$ such that $(\bar{G})_{ij}$ is one over the outdegree of i if there is an arc from i to j in G (in general, and usually, not stochastic because of rows of zeroes).
- d is the characteristic vector of dangling nodes (nodes without outgoing arcs).
- Let v and u be distributions, which we will call the *preference* and the *dangling-node* distribution.

PageRank: formal definition

- Is the definition of PageRank well-given? Are we all using the same definition?
- The *row-normalised matrix* of a (web) graph G is the matrix $\bar{G}$ such that $(\bar{G})_{ij}$ is one over the outdegree of i if there is an arc from i to j in G (in general, and usually, not stochastic because of rows of zeroes).
- d is the characteristic vector of dangling nodes (nodes without outgoing arcs).
- Let v and u be distributions, which we will call the *preference* and the *dangling-node* distribution.
- Let α be the *damping factor*.

PageRank: formal definition (2)

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation
$$r\big(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\big) = r$$

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation

$$r\big(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\big) = r$$

- Equivalently, as the unique stationary state of the Markov chain

$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$

that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation

$$r\left(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\right) = r$$

- Equivalently, as the unique stationary state of the Markov chain

$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$

  that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].

- Some notation:

$$r\left(\alpha \boxed{(\bar{G} + d^T u)} + (1-\alpha)\mathbf{1}^T v\right) = r$$

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation

$$r\left(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\right) = r$$

- Equivalently, as the unique stationary state of the Markov chain

$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$

  that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].
- Some notation:

$$r\left(\alpha \boxed{P} + (1-\alpha)\mathbf{1}^T v\right) = r$$

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation

$$r\big(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\big) = r$$

- Equivalently, as the unique stationary state of the Markov chain

$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$

  that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].
- Some notation:

$$r\big(\alpha P + (1-\alpha)\mathbf{1}^T v\big) = r$$

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation
$$r\left(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\right) = r$$
- Equivalently, as the unique stationary state of the Markov chain
$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$
that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].
- Some notation:
$$r\boxed{\left(\alpha P + (1-\alpha)\mathbf{1}^T v\right)} = r$$

PageRank: formal definition (2)

- PageRank r is defined (up to a scalar) by the eigenvector equation
$$r\left(\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v\right) = r$$
- Equivalently, as the unique stationary state of the Markov chain
$$\alpha(\bar{G} + d^T u) + (1-\alpha)\mathbf{1}^T v$$
that we call a *Markov chain with restart* [Boldi, Lonati, Santini & Vigna 2006].
- Some notation:
$$r\boxed{M} = r$$

PageRank closed formula

PageRank closed formula

Fixing $r\mathbf{1}^T = 1$,

$$\begin{aligned} rM &= r \\ r\left(\alpha P + (1-\alpha)\mathbf{1}^T v\right) &= r \\ \alpha r P + (1-\alpha) v &= r \\ (1-\alpha) v &= r(I - \alpha P), \end{aligned}$$

PageRank closed formula

Fixing $r\mathbf{1}^T = 1$,

$$\begin{aligned} rM &= r \\ r\big(\alpha P + (1-\alpha)\mathbf{1}^T v\big) &= r \\ \alpha r P + (1-\alpha) v &= r \\ (1-\alpha) v &= r(I - \alpha P), \end{aligned}$$

...which yields the following closed formula for PageRank:

$$r = (1-\alpha) v (1 - \alpha P)^{-1}.$$

PageRank closed formula

Fixing $r\mathbf{1}^T = 1$,

$$\begin{aligned} rM &= r \\ r\left(\alpha P + (1-\alpha)\mathbf{1}^T v\right) &= r \\ \alpha r P + (1-\alpha)v &= r \\ (1-\alpha)v &= r(I - \alpha P), \end{aligned}$$

...which yields the following closed formula for PageRank:

$$r = (1-\alpha)v(1-\alpha P)^{-1}.$$

So it's a linear system—*use Gauss–Seidel!*

PageRank closed formula

Fixing $r\mathbf{1}^T = 1$,

$$\begin{aligned} rM &= r \\ r\big(\alpha P + (1-\alpha)\mathbf{1}^T v\big) &= r \\ \alpha r P + (1-\alpha)v &= r \\ (1-\alpha)v &= r(I - \alpha P), \end{aligned}$$

...which yields the following closed formula for PageRank:

$$r = (1-\alpha)v(1-\alpha P)^{-1}.$$

So it's a linear system—*use Gauss–Seidel!*

Or use the Power Iteration Method:

$$\lim_{k\to\infty} x \cdot M^k$$

PageRank closed formula

Fixing $r\mathbf{1}^T = 1$,

$$\begin{aligned} rM &= r \\ r\left(\alpha P + (1-\alpha)\mathbf{1}^T v\right) &= r \\ \alpha r P + (1-\alpha)v &= r \\ (1-\alpha)v &= r(I - \alpha P), \end{aligned}$$

...which yields the following closed formula for PageRank:

$$r = (1-\alpha)v(1-\alpha P)^{-1}.$$

So it's a linear system—*use Gauss–Seidel!*

Or use the Power Iteration Method:

$$\lim_{k\to\infty} x \cdot M^k$$

Equivalently:

$$r = (1-\alpha)v\sum_{k=0}^{\infty}(\alpha P)^k.$$

PageRank and graph paths

- Let $G^*(-, i)$ be the set of all paths ending into i;

PageRank and graph paths

- Let $G^*(-, i)$ be the set of all paths ending into i;
- For any $\pi \in G^*(-, i)$, let $b(\pi)$ denote the *branching contribution* of π, i.e., the product of outdegrees of the nodes that are met on the path (excluding the ending node);

PageRank and graph paths

- Let $G^*(-, i)$ be the set of all paths ending into i;
- For any $\pi \in G^*(-, i)$, let $b(\pi)$ denote the *branching contribution* of π, i.e., the product of outdegrees of the nodes that are met on the path (excluding the ending node);
- The expression

$$r = (1-\alpha)v\sum_{k=0}^{\infty}(\alpha P)^k,$$

can be rewritten as

$$(r)_i = (1-\alpha)\sum_{\pi\in G^*(-,i)}\frac{v_{s(\pi)}}{b(\pi)}\alpha^{|\pi|}$$

PageRank and graph paths

- Let $G^*(-, i)$ be the set of all paths ending into i;
- For any $\pi \in G^*(-, i)$, let $b(\pi)$ denote the *branching contribution* of π, i.e., the product of outdegrees of the nodes that are met on the path (excluding the ending node);
- The expression

$$r = (1-\alpha)v\sum_{k=0}^{\infty}(\alpha P)^k,$$

can be rewritten as

$$(r)_i = (1-\alpha)\sum_{\pi \in G^*(-,i)} \frac{v_{s(\pi)}}{b(\pi)} \boxed{\alpha^{|\pi|}}$$

PageRank and graph paths

- Let $G^*(-, i)$ be the set of all paths ending into i;
- For any $\pi \in G^*(-, i)$, let $b(\pi)$ denote the *branching contribution* of π, i.e., the product of outdegrees of the nodes that are met on the path (excluding the ending node);
- The expression

$$r = (1-\alpha)v\sum_{k=0}^{\infty}(\alpha P)^k,$$

can be rewritten as

$$(r)_i = (1-\alpha)\sum_{\pi \in G^*(-,i)} \frac{v_{s(\pi)}}{b(\pi)} \boxed{f(|\pi|)}$$

where $f(-)$ is a suitable *damping function* that goes to zero sufficiently fast [Baeza–Yates, Boldi & Castillo 2006].

Iteration vs. approximation

Iteration vs. approximation

We can rewrite the summation as follows:

$$r = v + v \sum_{k=1}^{\infty} \alpha^k \left(P^k - P^{k-1}\right).$$

Iteration vs. approximation

We can rewrite the summation as follows:

$$r = v + v \sum_{k=1}^{\infty} \alpha^k \left(P^k - P^{k-1}\right).$$

Thus, the rational function r can be approximated using its Maclaurin polynomials (i.e., truncated series).

Iteration vs. approximation

We can rewrite the summation as follows:

$$r = v + v \sum_{k=1}^{\infty} \alpha^k \left(P^k - P^{k-1}\right).$$

Thus, the rational function r can be approximated using its Maclaurin polynomials (i.e., truncated series).

Theorem

The n-th approximation of PageRank computed by the Power Method with damping factor α and starting vector v coincides with the n-th degree Maclaurin polynomial of PageRank evaluated in α.

$$vM^n = v + v \sum_{k=1}^{n} \alpha^k \left(P^k - P^{k-1}\right).$$

One α to rule them all...

One α to rule them all...

Corollary

The difference between the k-th and the $(k-1)$-th approximation of PageRank (as computed by the Power Method with starting vector v), divided by α^k, is the k-th coefficient of the power series of PageRank.

One α to rule them all...

Corollary

The difference between the k-th and the $(k-1)$-th approximation of PageRank (as computed by the Power Method with starting vector v), divided by α^k, is the k-th coefficient of the power series of PageRank.

As a consequence the data obtained computing PageRank for a given α can be used to compute immediately PageRank for *any other* α, obtaining the result of the Power Method after the same number of iterations.

One α to rule them all...

Corollary

The difference between the k-th and the $(k-1)$-th approximation of PageRank (as computed by the Power Method with starting vector v), divided by α^k, is the k-th coefficient of the power series of PageRank.

As a consequence the data obtained computing PageRank for a given α can be used to compute immediately PageRank for *any other* α, obtaining the result of the Power Method after the same number of iterations.

By saving the Maclaurin coefficients during the computation of PageRank with a specific α it is possible to study the behaviour of PageRank when α varies.

One α to rule them all...

Corollary

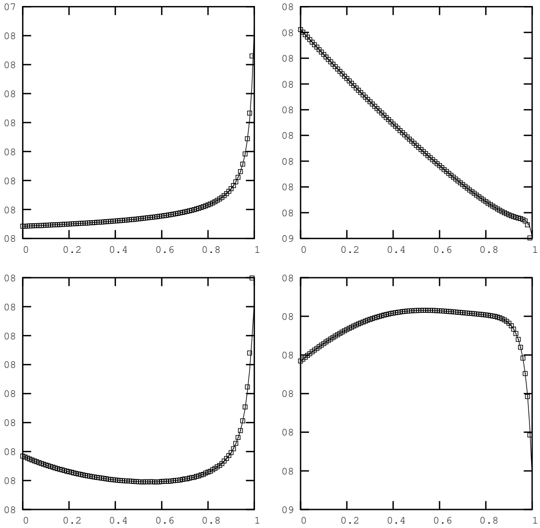
The difference between the k-th and the $(k-1)$-th approximation of PageRank (as computed by the Power Method with starting vector v), divided by α^k, is the k-th coefficient of the power series of PageRank.

As a consequence the data obtained computing PageRank for a given α can be used to compute immediately PageRank for *any other* α, obtaining the result of the Power Method after the same number of iterations.

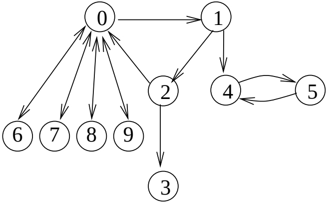
By saving the Maclaurin coefficients during the computation of PageRank with a specific α it is possible to study the behaviour of PageRank when α varies.

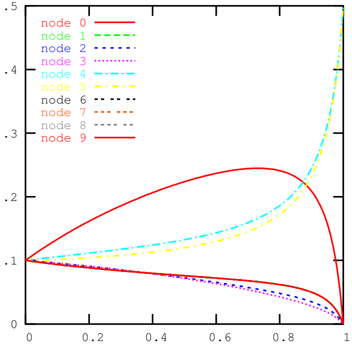
Even more is true, of course: using standard series derivation techniques, one can approximate the k-th derivative.

Some typical behaviours



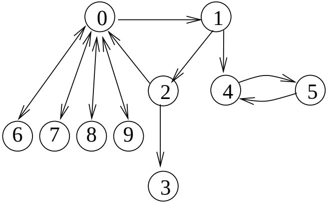
An example

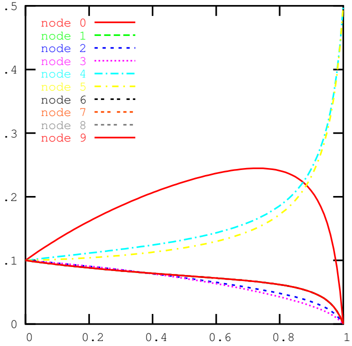




$$r_0(\alpha) = -5\,\frac{(-1+\alpha)\left(\alpha^2+18\,\alpha+4\right)}{8\,\alpha^4+\alpha^3-170\,\alpha^2-20\,\alpha+200}$$

An example





$$r_1(\alpha) = -2\,\frac{(-1+\alpha)\left(\alpha^2+2\,\alpha+10\right)}{8\,\alpha^4+\alpha^3-170\,\alpha^2-20\,\alpha+200}$$

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

- "The smart guys at Google use 0.85" (???).

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

- "The smart guys at Google use 0.85" (???).
- "It works pretty well".

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

- "The smart guys at Google use 0.85" (???).
- "It works pretty well".
- Iterative algorithms that approximate PageRank converge quickly if $\alpha = 0.85$: larger values would require more iterations; moreover...

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

- "The smart guys at Google use 0.85" (???).
- "It works pretty well".
- Iterative algorithms that approximate PageRank converge quickly if $\alpha = 0.85$: larger values would require more iterations; moreover...
- ...numeric instability arises when α is too close to 1...

The magic value $\alpha = 0.85$

One usually computes and considers only $r(0.85)$. Why 0.85?

- "The smart guys at Google use 0.85" (???).
- "It works pretty well".
- Iterative algorithms that approximate PageRank converge quickly if $\alpha = 0.85$: larger values would require more iterations; moreover...
- ...numeric instability arises when α is too close to 1...
- ...yet, we believe that understanding how $r(\alpha)$ changes when α is modified is important.

Some literature

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].
- The condition number of the PageRank problem is $(1+\alpha)/(1-\alpha)$ [Haveliwala & Kamvar 2003].

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].
- The condition number of the PageRank problem is $(1+\alpha)/(1-\alpha)$ [Haveliwala & Kamvar 2003].
- PageRank can be computed in the $\alpha \approx 1$ zone using Arnoldi-type methods [Del Corso, Gullì & Romani 2005; Golub & Grief 2006].

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].
- The condition number of the PageRank problem is $(1+\alpha)/(1-\alpha)$ [Haveliwala & Kamvar 2003].
- PageRank can be computed in the $\alpha \approx 1$ zone using Arnoldi-type methods [Del Corso, Gullì & Romani 2005; Golub & Grief 2006].
- PageRank can be extrapolated when $\alpha \approx 1$ (even $\alpha > 1$!) using an explicit formula based on the Jordan normal form [Serra–Capizzano 2005; Brezinski & Redivo–Zaglia 2006]

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].
- The condition number of the PageRank problem is $(1+\alpha)/(1-\alpha)$ [Haveliwala & Kamvar 2003].
- PageRank can be computed in the $\alpha \approx 1$ zone using Arnoldi-type methods [Del Corso, Gullì & Romani 2005; Golub & Grief 2006].
- PageRank can be extrapolated when $\alpha \approx 1$ (even $\alpha > 1$!) using an explicit formula based on the Jordan normal form [Serra–Capizzano 2005; Brezinski & Redivo–Zaglia 2006]
- Choose $\alpha = 1/2$! [Avrachenkov, Litvak & Kim 2006]

Some literature

- PageRank (values and rankings) change significantly when α is modified [Pretto 2002; Langville & Meyer 2004].
- Convergence rate of the Power Method is α [Haveliwala & Kamvar 2003].
- The condition number of the PageRank problem is $(1+\alpha)/(1-\alpha)$ [Haveliwala & Kamvar 2003].
- PageRank can be computed in the $\alpha \approx 1$ zone using Arnoldi-type methods [Del Corso, Gullì & Romani 2005; Golub & Grief 2006].
- PageRank can be extrapolated when $\alpha \approx 1$ (even $\alpha > 1$!) using an explicit formula based on the Jordan normal form [Serra–Capizzano 2005; Brezinski & Redivo–Zaglia 2006]
- Choose $\alpha = 1/2$! [Avrachenkov, Litvak & Kim 2006]
- . . . and many others.

What happens when $\alpha \to 1$?

$$\lim_{\alpha \to 1} M = P.$$

What happens when $\alpha \to 1$?

$$\lim_{\alpha \to 1} M = P.$$

The "preferential" part added to P vanishes, whereas the part due to $\bar{G}$ and u becomes larger: some interpret this fact as a hint that r becomes "more faithful to reality" when $\alpha \to 1$.

What happens when $\alpha \to 1$?

$$\lim_{\alpha \to 1} M = P.$$

The "preferential" part added to P vanishes, whereas the part due to $\bar{G}$ and u becomes larger: some interpret this fact as a hint that r becomes "more faithful to reality" when $\alpha \to 1$.

Is this true?

What happens when $\alpha \to 1$?

$$\lim_{\alpha \to 1} M = P.$$

The "preferential" part added to P vanishes, whereas the part due to $\bar{G}$ and u becomes larger: some interpret this fact as a hint that r becomes "more faithful to reality" when $\alpha \to 1$.

Is this true?

Since r is a coordinatewise bounded function defined on $[0, 1)$, the limit

$$r^* = \lim_{\alpha \to 1^-} r$$

exists.

A ready-made solution

A ready-made solution

In fact, since the *resolvent* $(I/\alpha - P)$ has a Laurent expansion around 1 in the largest disc not containing $1/\lambda$ for another eigenvalue λ of P, PageRank is analytic in the same disc; a standard computation yields

$$(1-\alpha)(1-\alpha P)^{-1} = P^* - \sum_{n=0}^{\infty} \left(\frac{\alpha - 1}{\alpha} \right)^{n+1} Q^{n+1},$$

where $Q = (I - P + P^*)^{-1} - P^*$ and

$$P^* = \lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n-1} P^k$$

is the *Cesáro limit* of P.

A ready-made solution

In fact, since the *resolvent* $(I/\alpha - P)$ has a Laurent expansion around 1 in the largest disc not containing $1/\lambda$ for another eigenvalue λ of P, PageRank is analytic in the same disc; a standard computation yields

$$(1-\alpha)(1-\alpha P)^{-1} = P^* - \sum_{n=0}^{\infty} \left(\frac{\alpha - 1}{\alpha}\right)^{n+1} Q^{n+1},$$

where $Q = (I - P + P^*)^{-1} - P^*$ and

$$P^* = \lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n-1} P^k$$

is the *Cesáro limit* of P.

We conclude that

$$r^* = vP^*.$$

A ready-made solution

In fact, since the *resolvent* $(I/\alpha - P)$ has a Laurent expansion around 1 in the largest disc not containing $1/\lambda$ for another eigenvalue λ of P, PageRank is analytic in the same disc; a standard computation yields

$$(1-\alpha)(1-\alpha P)^{-1} = P^* - \sum_{n=0}^{\infty} \left(\frac{\alpha-1}{\alpha}\right)^{n+1} Q^{n+1},$$

where $Q = (I - P + P^*)^{-1} - P^*$ and

$$P^* = \lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n-1} P^k$$

is the *Cesáro limit* of P.

We conclude that

$$r^* = vP^*.$$

What makes r^* different from other limit distributions? How can we describe its structure?

Characterising r^*

Characterising r^*

We shall characterise r^* using the structure of G (even in the presence of dangling nodes).

Characterising r^*

We shall characterise r^* using the structure of G (even in the presence of dangling nodes).

A node x of G is a *bucket* iff it is contained in a non-trivial strongly connected component with no arcs toward other components.

Characterising r^*

We shall characterise r^* using the structure of G (even in the presence of dangling nodes).

A node x of G is a *bucket* iff it is contained in a non-trivial strongly connected component with no arcs toward other components. (Non-trivial means that it contains at least one arc)

A characterisation theorem

A characterisation theorem

Corollary

Assume $u = \mathbf{1}/n$. *Then:*

1. *if* G *contains a bucket then a node is recurrent for* P *iff it is a bucket;*
2. *if* G *does not contain a bucket all nodes are recurrent for* P.

A characterisation theorem

Corollary

Assume $u = \mathbf{1}/n$. Then:

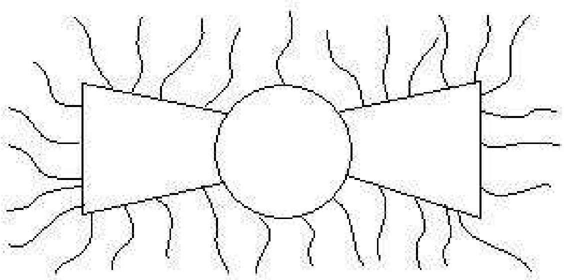
1. *if G contains a bucket then a node is recurrent for P iff it is a bucket;*
2. *if G does not contain a bucket all nodes are recurrent for P.*

Theorem

1. *If a bucket of G is reachable from the support of u then a node is recurrent for P iff it is a bucket of G;*
2. *if no bucket of G is reachable from the support of u, all nodes reachable from the support of u form a bucket component of P; hence, a node is recurrent for P iff it is in a bucket component of G or it is reachable from the support of u.*

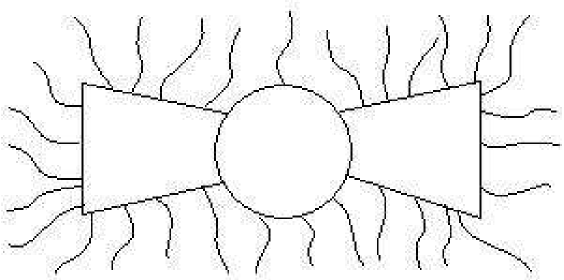
Bowtie

As a consequence, when $\alpha \to 1$, all PageRank concentrates in a bunch of pages that live in the rightmost part of the bowtie [Kumar et al., '00]:



Bowtie

As a consequence, when $\alpha \to 1$, all PageRank concentrates in a bunch of pages that live in the rightmost part of the bowtie [Kumar et al., '00]:



$r(\alpha)$ becomes meaningless as $\alpha \to 1$!

Interpretation

The statement of the previous theorem may seem a bit unfathomable. The essence, however, could be stated as follows: except for strongly connected graphs, or graphs whose terminal components are dangling, **the recurrent nodes are exactly the buckets** (unless we are in the very pathological case in which no bucket is reachable from the support of u).

Interpretation

The statement of the previous theorem may seem a bit unfathomable. The essence, however, could be stated as follows: except for strongly connected graphs, or graphs whose terminal components are dangling, **the recurrent nodes are exactly the buckets** (unless we are in the very pathological case in which no bucket is reachable from the support of u).

As we remarked, a real-world graph will certainly contain many buckets, so the first statement of the theorem will hold. This means that *most* nodes x will have zero rank when $\alpha \to 1$; particular, all nodes in the core component.

Interpretation

The statement of the previous theorem may seem a bit unfathomable. The essence, however, could be stated as follows: except for strongly connected graphs, or graphs whose terminal components are dangling, **the recurrent nodes are exactly the buckets** (unless we are in the very pathological case in which no bucket is reachable from the support of u).

As we remarked, a real-world graph will certainly contain many buckets, so the first statement of the theorem will hold. This means that *most* nodes x will have zero rank when $\alpha \to 1$; particular, all nodes in the core component.

In a word: **PageRank when $\alpha \to 1$ is nonsense** in all real-world cases. . .

Interpretation

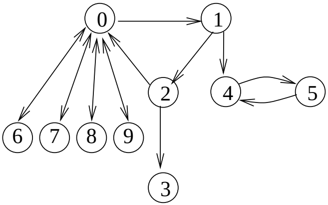
The statement of the previous theorem may seem a bit unfathomable. The essence, however, could be stated as follows: except for strongly connected graphs, or graphs whose terminal components are dangling, **the recurrent nodes are exactly the buckets** (unless we are in the very pathological case in which no bucket is reachable from the support of u).

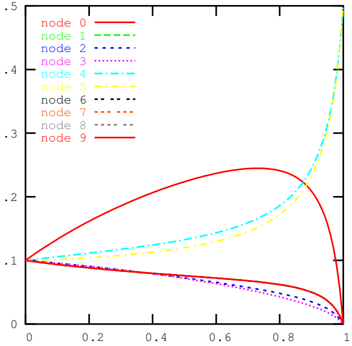
As we remarked, a real-world graph will certainly contain many buckets, so the first statement of the theorem will hold. This means that *most* nodes x will have zero rank when $\alpha \to 1$; particular, all nodes in the core component.

In a word: **PageRank when** $\alpha \to 1$ **is nonsense** in all real-world cases. . .

. . . and if you want the dire truth, there is an explicit formula in [Avrachenkov, Litvak & Kim 2006].

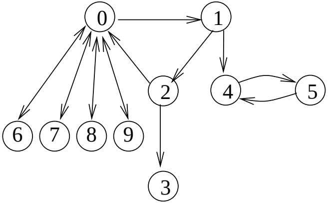
An example

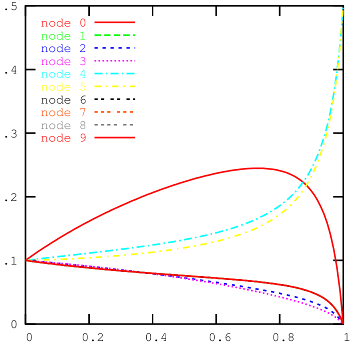




$$r_0(\alpha) = -5\,\frac{(-1+\alpha)\left(\alpha^2+18\,\alpha+4\right)}{8\,\alpha^4+\alpha^3-170\,\alpha^2-20\,\alpha+200}$$

An example





$$r_1(\alpha) = -2\,\frac{(-1+\alpha)\left(\alpha^2+2\,\alpha+10\right)}{8\,\alpha^4+\alpha^3-170\,\alpha^2-20\,\alpha+200}$$

General behaviour

What about the general behaviour of r?

General behaviour

What about the general behaviour of r?

We have an explicit formula for derivatives of PageRank ($k > 0$):

$$r^{(k)}(\alpha) = k!v\big(P^k - P^{k-1}\big)(I - \alpha P)^{-(k+1)}.$$

General behaviour

What about the general behaviour of r?

We have an explicit formula for derivatives of PageRank ($k > 0$):

$$r^{(k)}(\alpha) = k!v\big(P^k - P^{k-1}\big)(I - \alpha P)^{-(k+1)}.$$

Approximating them is also not difficult, since we have Maclaurin polynomials ($[\![r^{(k)}(\alpha)]\!]_t$ is the polynomial of order t):

Theorem

If $t \geq k/(1-\alpha)$,

$$\left\|r^{(k)}(\alpha) - [\![r^{(k)}(\alpha)]\!]_t\right\| \leq \frac{\delta_t}{1-\delta_t}\left\|[\![r^{(k)}(\alpha)]\!]_t - [\![r^{(k)}(\alpha)]\!]_{t-1}\right\|,$$

where

$$1 > \delta_t = \frac{\alpha(t+1)}{t+1-k}.$$

An alternative proposal. . .

Instead of using a *specific* value of α, one could try to use the *average* value

An alternative proposal...

Instead of using a *specific* value of α, one could try to use the *average* value, or equivalently:

$$T_i = \int_0^1 (r)_i \, d\alpha \qquad \text{(TotalRank [Boldi 2005])}$$

Also TotalRank is a special case of the general ranking technique of [Baeza–Yates, Boldi & Castillo 2006].

An alternative proposal...

Instead of using a *specific* value of α, one could try to use the *average* value, or equivalently:

$$T_i = \int_0^1 (r)_i \, d\alpha \qquad \text{(TotalRank [Boldi 2005])}$$

Also TotalRank is a special case of the general ranking technique of [Baeza–Yates, Boldi & Castillo 2006]. The two damping functions for TotalRank and PageRank are:

$$\begin{aligned} d_T(\ell) &= \frac{1}{(t+1)(t+2)} \\ d_P(\ell) &= (1-\alpha)\alpha^\ell. \end{aligned}$$

. . . and a possible explanation for .85

If you consider the sum of their differences up to length ℓ (average path length in the graph you are considering), you get:

$$\alpha^{\ell+1} - \frac{1}{\ell+2}.$$

. . . and a possible explanation for .85

If you consider the sum of their differences up to length ℓ (average path length in the graph you are considering), you get:

$$\alpha^{\ell+1} - \frac{1}{\ell+2}.$$

For a given ℓ, the value $\alpha^*(\ell)$ minimizing this sum is:

$$\alpha^*(\ell) = 1 - \frac{\log \ell}{\ell} + O\left(\frac{\log^2 \ell}{\ell^2}\right).$$

. . . and a possible explanation for .85

If you consider the sum of their differences up to length ℓ (average path length in the graph you are considering), you get:

$$\alpha^{\ell+1} - \frac{1}{\ell+2}.$$

For a given ℓ, the value $\alpha^*(\ell)$ minimizing this sum is:

$$\alpha^*(\ell) = 1 - \frac{\log \ell}{\ell} + O\left(\frac{\log^2 \ell}{\ell^2}\right).$$

The average path length of the Web is about 20, and $\alpha^*(20) \approx .85\ldots$

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but...

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but. . .
- . . . in real-world crawls, which have a large number of dangling nodes, the dangling preference is also very important.

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but...
- ... in real-world crawls, which have a large number of dangling nodes, the dangling preference is also very important.
- In the literature one can find several alternatives (e.g., $u = v$ or $u = \mathbf{1}/n$).

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G} + d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but. . .
- . . . in real-world crawls, which have a large number of dangling nodes, the dangling preference is also very important.
- In the literature one can find several alternatives (e.g., $u = v$ or $u = \mathbf{1}/n$).
- We suggest to distinguish clearly between *strongly preferential* PageRank ($u = v$) and *weakly preferential* PageRank.

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but...
- ... in real-world crawls, which have a large number of dangling nodes, the dangling preference is also very important.
- In the literature one can find several alternatives (e.g., $u = v$ or $u = \mathbf{1}/n$).
- We suggest to distinguish clearly between *strongly preferential* PageRank ($u = v$) and *weakly preferential* PageRank.
- Papers abound on both sides (and even on the I-don't-care-about-dangling-nodes side!)...

Strong vs. weak

$$r = (1-\alpha)v(1-\alpha(\bar{G}+d^T u))^{-1}.$$

- Clearly, the preference vector conditions significantly PageRank, but...
- ... in real-world crawls, which have a large number of dangling nodes, the dangling preference is also very important.
- In the literature one can find several alternatives (e.g., $u = v$ or $u = \mathbf{1}/n$).
- We suggest to distinguish clearly between *strongly preferential* PageRank ($u = v$) and *weakly preferential* PageRank.
- Papers abound on both sides (and even on the I-don't-care-about-dangling-nodes side!)...
- ...but the two versions are *very different!*: On a 100 million pages snapshot of the `.uk` domain, Kendall's τ is $\approx$ .25 for a topic-based v and $u = \mathbf{1}/n$! [Boldi *et al.* 2006]

Weakly preferential

Weakly preferential

Clearly, **weakly preferential** PageRank is a *linear operator* associating to the preference distribution another distribution. Said otherwise, for a fixed α PageRank is a linear function applied to the preference vector:

$$r = (1 - \alpha)v(1 - \alpha P)^{-1}.$$

Weakly preferential

Clearly, **weakly preferential** PageRank is a *linear operator* associating to the preference distribution another distribution. Said otherwise, for a fixed α PageRank is a linear function applied to the preference vector:

$$r = (1 - \alpha)v(1 - \alpha P)^{-1}.$$

This linear dependence makes it possible **to compute directly PageRank on any convex combination of preference vectors** for which it is already known.

Weakly preferential

Clearly, **weakly preferential** PageRank is a *linear operator* associating to the preference distribution another distribution. Said otherwise, for a fixed α PageRank is a linear function applied to the preference vector:

$$r = (1 - \alpha)v(1 - \alpha P)^{-1}.$$

This linear dependence makes it possible **to compute directly PageRank on any convex combination of preference vectors** for which it is already known.

This property is essential to compute personalised scores [Jeh & Widom 2002].

Weakly preferential

Clearly, **weakly preferential** PageRank is a *linear operator* associating to the preference distribution another distribution. Said otherwise, for a fixed α PageRank is a linear function applied to the preference vector:

$$r = (1-\alpha)v(1-\alpha P)^{-1}.$$

This linear dependence makes it possible **to compute directly PageRank on any convex combination of preference vectors** for which it is already known.

This property is essential to compute personalised scores [Jeh & Widom 2002].

Using the Sherman–Morrison formula it is possible to make the dependence on v and u explicit, and sort out what happens in the strongly preferential case.

Pseudoranks

Pseudoranks

Let us define the *pseudorank* of G with preference vector v and damping factor $\alpha \in [0\,..\,1]$:

$$\widetilde{v}(\alpha) = (1-\alpha)v\left(I - \alpha\bar{G}\right)^{-1}.$$

Pseudoranks

Let us define the *pseudorank* of G with preference vector v and damping factor $\alpha \in [0\,..\,1]$:

$$\widetilde{v}(\alpha) = (1-\alpha)v\left(I - \alpha\bar{G}\right)^{-1}.$$

The above definition can be extended by continuity to $\alpha = 1$ even when 1 is an eigenvalue of $\bar{G}$, always using the fact that $\left(I/\alpha - \bar{G}\right)$ has a Laurent expansion around 1, getting again $v\bar{G}^*$.

Pseudoranks

Let us define the *pseudorank* of G with preference vector v and damping factor $\alpha \in [0\,..\,1]$:

$$\widetilde{v}(\alpha) = (1-\alpha)v\left(I - \alpha\bar{G}\right)^{-1}.$$

The above definition can be extended by continuity to $\alpha = 1$ even when 1 is an eigenvalue of $\bar{G}$, always using the fact that $\left(I/\alpha - \bar{G}\right)$ has a Laurent expansion around 1, getting again $v\bar{G}^*$.

When $\alpha < 1$ the matrix $(I - \alpha\bar{G})$ is strictly diagonally dominant, so the Gauss–Seidel method can be used to compute quickly pseudoranks.

Pseudoranks

Let us define the *pseudorank* of G with preference vector v and damping factor $\alpha \in [0\mathinner{\ldotp\ldotp}1]$:

$$\widetilde{v}(\alpha) = (1-\alpha)v\bigl(I - \alpha\bar{G}\bigr)^{-1}.$$

The above definition can be extended by continuity to $\alpha = 1$ even when 1 is an eigenvalue of $\bar{G}$, always using the fact that $\bigl(I/\alpha - \bar{G}\bigr)$ has a Laurent expansion around 1, getting again $v\bar{G}^*$.

When $\alpha < 1$ the matrix $(I - \alpha\bar{G})$ is strictly diagonally dominant, so the Gauss–Seidel method can be used to compute quickly pseudoranks.

Note that $\widetilde{v}(\alpha)$ is *linear* in v.

Pseudoranks

Let us define the *pseudorank* of G with preference vector v and damping factor $\alpha \in [0\,..\,1]$:

$$\widetilde{v}(\alpha) = (1-\alpha)v\left(I - \alpha\bar{G}\right)^{-1}.$$

The above definition can be extended by continuity to $\alpha = 1$ even when 1 is an eigenvalue of $\bar{G}$, always using the fact that $\left(I/\alpha - \bar{G}\right)$ has a Laurent expansion around 1, getting again $v\bar{G}^*$.

When $\alpha < 1$ the matrix $(I - \alpha\bar{G})$ is strictly diagonally dominant, so the Gauss–Seidel method can be used to compute quickly pseudoranks.

Note that $\widetilde{v}(\alpha)$ is *linear* in v.

The notion appears in [Del Corso, Gullì & Romani 2004] and it has been used in [McSherry 2005; Fogaras, Rácz, Csalogány & Sarlós 2005] (actually, as *the* definition of PageRank).

Explicit dependence

Explicit dependence

Using pseudoranks we can easily express the dependence [Boldi, Posenato, Santini & Vigna 2006]:

$$r = \widetilde{v}(\alpha) - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{u}(\alpha)d^T}\widetilde{u}(\alpha).$$

Explicit dependence

Using pseudoranks we can easily express the dependence [Boldi, Posenato, Santini & Vigna 2006]:

$$r = \widetilde{v}(\alpha) - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{u}(\alpha)d^T}\widetilde{u}(\alpha).$$

Using this formula, once the pseudoranks for certain distributions have been computed, it is possible to compute PageRank using any *convex combination* of such distributions as preference and dangling-node distribution.

Explicit dependence

Using pseudoranks we can easily express the dependence [Boldi, Posenato, Santini & Vigna 2006]:

$$r = \widetilde{v}(\alpha) - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{u}(\alpha)d^T}\widetilde{u}(\alpha).$$

Using this formula, once the pseudoranks for certain distributions have been computed, it is possible to compute PageRank using any *convex combination* of such distributions as preference and dangling-node distribution.

Another evident feature of the above formula is that the dependence on the dangling-node distribution is *not linear*, so *we cannot expect strongly preferential PageRank to be linear in* v.

The strongly preferential case

The strongly preferential case

Nonetheless, if we fix $u = v$ and simplify the resulting formula (getting back the formula obtained by Del Corso, Gullì and Romani)...

$$r = \widetilde{v}(\alpha)\left(1 - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{v}(\alpha)d^T}\right)$$

The strongly preferential case

Nonetheless, if we fix $u = v$ and simplify the resulting formula (getting back the formula obtained by Del Corso, Gullì and Romani)...

$$r = \widetilde{v}(\alpha)\left(1 - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{v}(\alpha)d^T}\right)$$

So *pseudoranks are just multiples of strongly preferential ranks*, and the side effect is that *strongly preferential PageRank can be computed by convex combination of pseudoranks*.

The strongly preferential case

Nonetheless, if we fix $u = v$ and simplify the resulting formula (getting back the formula obtained by Del Corso, Gullì and Romani)...

$$r = \widetilde{v}(\alpha)\left(1 - \frac{\widetilde{v}(\alpha)d^T}{1 - \frac{1}{\alpha} + \widetilde{v}(\alpha)d^T}\right)$$

So *pseudoranks are just multiples of strongly preferential ranks*, and the side effect is that *strongly preferential PageRank can be computed by convex combination of pseudoranks*.

Assuming that $v = \lambda x + (1 - \lambda)y$, we have

$$r = r_{\lambda x+(1-\lambda)y}(\alpha) \quad \propto \quad \lambda\widetilde{x}(\alpha) + (1 - \lambda)\widetilde{y}(\alpha)$$

Alternatives to PageRank

PageRank is but one of the many link-based methods to establish page importance. Other notable examples are:

Alternatives to PageRank

PageRank is but one of the many link-based methods to establish page importance. Other notable examples are:

- HITS (Kleinberg)

Alternatives to PageRank

PageRank is but one of the many link-based methods to establish page importance. Other notable examples are:

- HITS (Kleinberg)
- SALSA (Lempel, Moran), a variant of HITS (not covered here)

HITS

HITS (Hyperlink-Induced Topic Search) is based on the idea that the web contains, for every topic, two "types" of pages:

HITS

HITS (Hyperlink-Induced Topic Search) is based on the idea that the web contains, for every topic, two "types" of pages:

- *authoritative* pages about the topic

HITS

HITS (Hyperlink-Induced Topic Search) is based on the idea that the web contains, for every topic, two “types” of pages:

- *authoritative* pages about the topic
- *hub* pages that are not authoritative but contain link to many authoritative pages.

HITS

HITS (Hyperlink-Induced Topic Search) is based on the idea that the web contains, for every topic, two “types” of pages:

- *authoritative* pages about the topic
- *hub* pages that are not authoritative but contain link to many authoritative pages.

HITS gives two scores to every page, measuring their authoritativeness and their hubbiness.

HITS

HITS (Hyperlink-Induced Topic Search) is based on the idea that the web contains, for every topic, two “types” of pages:

- *authoritative* pages about the topic
- *hub* pages that are not authoritative but contain link to many authoritative pages.

HITS gives two scores to every page, measuring their authoritativeness and their hubbiness.

Differently from PageRank it is *not query independent*.

HITS (cont'd)

The algorithm works in two phases:

HITS (cont'd)

The algorithm works in two phases:

- a graph G_q (a subgraph of the whole web graph) is singled out (depending on the query)

HITS (cont'd)

The algorithm works in two phases:

- a graph G_q (a subgraph of the whole web graph) is singled out (depending on the query)
- the authoritativeness/hubbiness scores are computed for the pages in G_q

HITS — Phase 1

HITS — Phase 1

G_q is obtained as follows:

- the set S_q of the top k pages relative to q are obtained using some techniques (e.g., BM25)
- for each $x \in S_q$, all nodes in $N^+(x)$ are added
- for each $x \in S_q$, at most h nodes of $N^-(x)$ are added

HITS — Phase 2

HITS — Phase 2

At every iteration, we will have two scores $h_x(t)$ and $a_x(t)$ for every node $x \in N_{G_q}$.

HITS — Phase 2

At every iteration, we will have two scores $h_x(t)$ and $a_x(t)$ for every node $x \in N_{G_q}$.

$$h_x(t+1) \propto \sum_{x \to y} a_y(t)$$
$$a_x(t+1) \propto \sum_{x \leftarrow y} h_y(t)$$

HITS — Phase 2

At every iteration, we will have two scores $h_x(t)$ and $a_x(t)$ for every node $x \in N_{G_q}$.

$$
\begin{aligned}
h_x(t+1) &\propto \sum_{x \to y} a_y(t) \\
a_x(t+1) &\propto \sum_{x \leftarrow y} h_y(t)
\end{aligned}
$$

The $\propto$ is necessary to avoid divergence (the scores are normalized at every iteration).

HITS — In practice

HITS (proposed by Kleinberg in 1999) is not used by most search engine, probably due to:

- its dynamic nature (requiring computation at query time)
- its marginal benefits over PageRank.

HITS — In practice

HITS (proposed by Kleinberg in 1999) is not used by most search engine, probably due to:

- its dynamic nature (requiring computation at query time)
- its marginal benefits over PageRank.

It was supposedly used by Teoma (later Ask.com).