Buscadores Web

(o como pude encontrar la información para esta plática).

José David Flores Peñaloza

IIMAS-IMATE

UNAM

25/10/2005

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

Contenido

- Motivación.
- Prehistoria.
- Historia.
- Arquitectura de un buscador web.
- RI clásica.
- RI en web: Análisis de vínculos
- HITS
- PageRank

¡Gracias por su atención!

Motivación

La web presenta a los usuarios modernos un acceso a la información sin precedentes, pero...

Mucha es de muy mala calidad y...

Sin buscadores, encontrar información realmente buena es como encontrar una aguja en un pajar...

Es por ello que diseñar buenas técnicas de recuperación de información en la web es importantísimo.

¡Intenten navegar medio dia sin usar un buscador!

Motivación

La web presenta a los usuarios modernos un acceso a la información sin precedentes, pero...

Mucha es de muy mala calidad y...

Sin buscadores, encontrar información realmente buena es como encontrar una aguja en un pajar...

Es por ello que diseñar buenas técnicas de recuperación de información en la web es importantísimo.

¡Intenten navegar medio dia sin usar un buscador!

Motivación

La web presenta a los usuarios modernos un acceso a la información sin precedentes, pero...

Mucha es de muy mala calidad y...

Sin buscadores, encontrar información realmente buena es como encontrar una aguja en un pajar...

Es por ello que diseñar buenas técnicas de recuperación de información en la web es importantísimo.

¡Intenten navegar medio dia sin usar un buscador!

Motivación

La web presenta a los usuarios modernos un acceso a la información sin precedentes, pero...

Mucha es de muy mala calidad y...

Sin buscadores, encontrar información realmente buena es como encontrar una aguja en un pajar...

Es por ello que diseñar buenas técnicas de recuperación de información en la web es importantísimo.

¡Intenten navegar medio dia sin usar un buscador!

Motivación

La web presenta a los usuarios modernos un acceso a la información sin precedentes, pero...

Mucha es de muy mala calidad y...

Sin buscadores, encontrar información realmente buena es como encontrar una aguja en un pajar...

Es por ello que diseñar buenas técnicas de recuperación de información en la web es importantísimo.

¡Intenten navegar medio dia sin usar un buscador!

Motivación [cont.]

No es sencillo diseñar buenos buscadores. Existen muchísimos retos para hacerlo..

- Cada quien publica lo que se le da la gana sin avisarle a nadie.
- El tamaño de la red es inmenso. Se estima que el número de páginas indexables por buscadores es más de once mil millones![9].
- Hay intereses económicos para engañar a los buscadores y posicionarse en los mejores lugares de las listas de resultados[8].

Motivación [cont.]

No es sencillo diseñar buenos buscadores. Existen muchísimos retos para hacerlo..

- Cada quien publica lo que se le da la gana sin avisarle a nadie.
- El tamaño de la red es inmenso. Se estima que el número de páginas indexables por buscadores es más de once mil millones![9].
- Hay intereses económicos para engañar a los buscadores y posicionarse en los mejores lugares de las listas de resultados[8].

Motivación [cont.]

No es sencillo diseñar buenos buscadores. Existen muchísimos retos para hacerlo..

- Cada quien publica lo que se le da la gana sin avisarle a nadie.
- El tamaño de la red es inmenso. Se estima que el número de páginas indexables por buscadores es más de once mil millones![9].
- Hay intereses económicos para engañar a los buscadores y posicionarse en los mejores lugares de las listas de resultados[8].

Motivación [cont.]

No es sencillo diseñar buenos buscadores. Existen muchísimos retos para hacerlo..

- Cada quien publica lo que se le da la gana sin avisarle a nadie.
- El tamaño de la red es inmenso. Se estima que el número de páginas indexables por buscadores es más de once mil millones![9].
- Hay intereses económicos para engañar a los buscadores y posicionarse en los mejores lugares de las listas de resultados[8].

Motivación [cont.]

"Al igual que la web, el mundo de los servicios de búsqueda es ahora complejo, rico, volatil, y frecuentemente frustrante"[7]

Prehistoria[7]

- Al inicio de los tiempos la www no existia... Los recursos asociados a los nuevos servicios (telnet, ftp, gopher, etc), tenian que publicitarse de boca en boca, o más comunmente de e-mail en e-mail....
- Afortunadamente cada nueva función fue rapidamente enriquecida con el desarrollo de uno o más sistemas electrónicos de descubrimiento de recursos...
  - Los archivos disponibles por FTP podian encontrarse con archie
  - los archivos de servidores de listas podian buscarse con comandos enviados a los servidores.
  - Llegó gopher...

Prehistoria[7]

- Al inicio de los tiempos la www no existia... Los recursos asociados a los nuevos servicios (telnet, ftp, gopher, etc), tenian que publicitarse de boca en boca, o más comunmente de e-mail en e-mail....
- Afortunadamente cada nueva función fue rapidamente enriquecida con el desarrollo de uno o más sistemas electrónicos de descubrimiento de recursos...
  - Los archivos disponibles por FTP podian encontrarse con archie
  - los archivos de servidores de listas podian buscarse con comandos enviados a los servidores.
  - Llegó gopher...

Prehistoria[7]

- Al inicio de los tiempos la www no existia... Los recursos asociados a los nuevos servicios (telnet, ftp, gopher, etc), tenian que publicitarse de boca en boca, o más comunmente de e-mail en e-mail....
- Afortunadamente cada nueva función fue rapidamente enriquecida con el desarrollo de uno o más sistemas electrónicos de descubrimiento de recursos...
  - Los archivos disponibles por FTP podian encontrarse con archie
  - los archivos de servidores de listas podian buscarse con comandos enviados a los servidores.
  - Llegó gopher...

Prehistoria[7]

- Al inicio de los tiempos la www no existia... Los recursos asociados a los nuevos servicios (telnet, ftp, gopher, etc), tenian que publicitarse de boca en boca, o más comunmente de e-mail en e-mail....
- Afortunadamente cada nueva función fue rapidamente enriquecida con el desarrollo de uno o más sistemas electrónicos de descubrimiento de recursos...
  - Los archivos disponibles por FTP podian encontrarse con archie
  - los archivos de servidores de listas podian buscarse con comandos enviados a los servidores.
  - Llegó gopher...

Prehistoria[7]

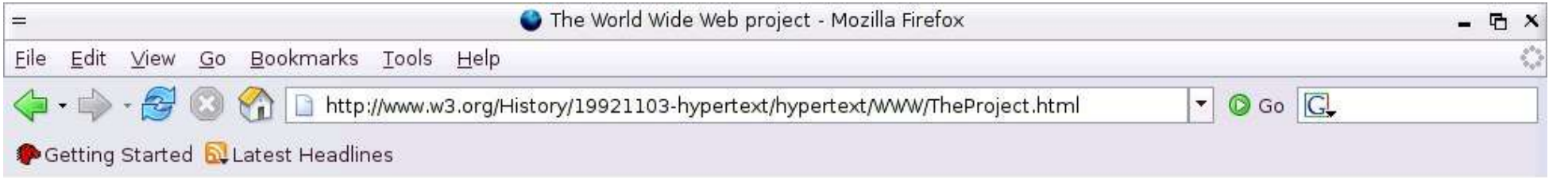
- Al inicio de los tiempos la www no existia... Los recursos asociados a los nuevos servicios (telnet, ftp, gopher, etc), tenian que publicitarse de boca en boca, o más comunmente de e-mail en e-mail....
- Afortunadamente cada nueva función fue rapidamente enriquecida con el desarrollo de uno o más sistemas electrónicos de descubrimiento de recursos...
  - Los archivos disponibles por FTP podian encontrarse con archie
  - los archivos de servidores de listas podian buscarse con comandos enviados a los servidores.
  - Llegó gopher...

Prehistoria [cont.]

- En 1991 aparecieron los primeros navegadores en modo texto en el CERN...
- Ahi tambien aparecio el primer servicio de descubrimiento de recursos, el cual incluia un listado alfabetico de las páginas que formaban la biblioteca virtual de la www

Prehistoria [cont.]

- En 1991 aparecieron los primeros navegadores en modo texto en el CERN...
- Ahi tambien aparecio el primer servicio de descubrimiento de recursos, el cual incluia un listado alfabetico de las páginas que formaban la biblioteca virtual de la www



World Wide Web

The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists , Policy , November's W3 news , Frequently Asked Questions .

What's out there?
: Pointers to the world's online information, subjects , W3 servers, etc.

Help
: on the browser you are using

Software Products
: A list of W3 project components and their current state. (e.g. Line Mode ,X11 Viola , NeXTStep , Servers , Tools , Mail robot , Library)

Technical
: Details of protocols, formats, program internals etc

Bibliography
: Paper documentation on W3 and references.

People
: A list of some people involved in the project.

History
: A summary of the history of the project.

How can I help ?
: If you would like to support the web..

Getting code
: Getting the code by anonymous FTP , etc.

Prehistoria [cont.]

- En 1994, al incrementarse el número de recursos http, los servicios que hoy conocemos como buscadores web comenzaron a aparecer (WWWW, Galaxy, Yahoo, Webcrawler,...).
- La mayoria era un directorio organizado en categorias jerarquicas construidas a mano!
- La mayoria comenzó como proyectos de investigación -o recreación- llevados a cabo por estudiantes graduados.
- En 1996 se comenzó a hablar de ellos en revistas de negocios, periodicos y television.

Prehistoria [cont.]

- En 1994, al incrementarse el número de recursos http, los servicios que hoy conocemos como buscadores web comenzaron a aparecer (WWWW, Galaxy, Yahoo, Webcrawler,...).
- La mayoria era un directorio organizado en categorias jerarquicas construidas a mano!
- La mayoria comenzó como proyectos de investigación -o recreación- llevados a cabo por estudiantes graduados.
- En 1996 se comenzó a hablar de ellos en revistas de negocios, periodicos y television.

Prehistoria [cont.]

- En 1994, al incrementarse el número de recursos http, los servicios que hoy conocemos como buscadores web comenzaron a aparecer (WWWW, Galaxy, Yahoo, Webcrawler,...).
- La mayoria era un directorio organizado en categorias jerarquicas construidas a mano!
- La mayoria comenzó como proyectos de investigación -o recreación- llevados a cabo por estudiantes graduados.
- En 1996 se comenzó a hablar de ellos en revistas de negocios, periodicos y television.

Prehistoria [cont.]

- En 1994, al incrementarse el número de recursos http, los servicios que hoy conocemos como buscadores web comenzaron a aparecer (WWWW, Galaxy, Yahoo, Webcrawler,...).
- La mayoria era un directorio organizado en categorias jerarquicas construidas a mano!
- La mayoria comenzó como proyectos de investigación -o recreación- llevados a cabo por estudiantes graduados.
- En 1996 se comenzó a hablar de ellos en revistas de negocios, periodicos y television.

Historia (pre-AV)

- A finales de 1995 DEC lanzó Altavista, corriendo en cúmulos de máquinas Alpha y con indexado en memoria principal.
- Inktomi aparecio en 1996.

Los buscadores más modernas de estas epocas usaban técnicas de RI tradicionales. Buscaban cuales eran los documentos que contenian los términos de la consulta.

Ordenaban los resultados usando tecnicas bibliotecarias, como proximidad de documentos.

Historia (pre-AV)

- A finales de 1995 DEC lanzó Altavista, corriendo en cúmulos de máquinas Alpha y con indexado en memoria principal.
- Inktomi aparecio en 1996.

Los buscadores más modernas de estas epocas usaban técnicas de RI tradicionales. Buscaban cuales eran los documentos que contenian los términos de la consulta.

Ordenaban los resultados usando tecnicas bibliotecarias, como proximidad de documentos.

Historia (pre-AV)

- A finales de 1995 DEC lanzó Altavista, corriendo en cúmulos de máquinas Alpha y con indexado en memoria principal.
- Inktomi aparecio en 1996.

Los buscadores más modernas de estas epocas usaban técnicas de RI tradicionales. Buscaban cuales eran los documentos que contenian los términos de la consulta.

Ordenaban los resultados usando tecnicas bibliotecarias, como proximidad de documentos.

Historia (pre-AV)

- A finales de 1995 DEC lanzó Altavista, corriendo en cúmulos de máquinas Alpha y con indexado en memoria principal.
- Inktomi aparecio en 1996.

Los buscadores más modernas de estas epocas usaban técnicas de RI tradicionales. Buscaban cuales eran los documentos que contenian los términos de la consulta.

Ordenaban los resultados usando tecnicas bibliotecarias, como proximidad de documentos.

Historia (post-AV)

- El parteaguas en la historia de los buscadores es el uso de análisis de vínculos.
- Las técnicas de RI usadas anteriormente fueron diseñadas para colecciones de documentos bien organizadas -nada que ver con la red-.
- Investigadores descubrieron que la estructura de vínculos de la red podia aprovecharse para obtener información valiosa acerca de la importancia de páginas web. Extendieron las tecnicas de bibliometría.

Historia (post-AV)

- El parteaguas en la historia de los buscadores es el uso de análisis de vínculos.
- Las técnicas de RI usadas anteriormente fueron diseñadas para colecciones de documentos bien organizadas -nada que ver con la red-.
- Investigadores descubrieron que la estructura de vínculos de la red podia aprovecharse para obtener información valiosa acerca de la importancia de páginas web. Extendieron las tecnicas de bibliometría.

Historia (post-AV)

- El parteaguas en la historia de los buscadores es el uso de análisis de vínculos.
- Las técnicas de RI usadas anteriormente fueron diseñadas para colecciones de documentos bien organizadas -nada que ver con la red-.
- Investigadores descubrieron que la estructura de vínculos de la red podia aprovecharse para obtener información valiosa acerca de la importancia de páginas web. Extendieron las tecnicas de bibliometría.

Historia (post-AV) [cont.]

- Este análisis, junto con el desarrollo de otras heuristicas hacen que el uso previo de buscadores en la web palideciera, en comparación con la precisión de los buscadores modernos.

Arquitectura[6]

Un navegador web se compone genéricamente de tres partes principales:

- Una araña (crawler).
- Un indexador.
- Software de búsqueda y ranqueo.

Arquitectura[6]

Un navegador web se compone genéricamente de tres partes principales:

- Una araña (crawler).
- Un indexador.
- Software de búsqueda y ranqueo.

Arquitectura[6]

Un navegador web se compone genéricamente de tres partes principales:

- Una araña (crawler).
- Un indexador.
- Software de búsqueda y ranqueo.

RI clásica

No toma en cuenta la estructura de enlaces de la web.

(Considera las páginas web como documentos aislados).

Una de sus técnicas es el modelo de espacio vectorial.

RI clásica

No toma en cuenta la estructura de enlaces de la web.

(Considera las páginas web como documentos aislados).

Una de sus técnicas es el modelo de espacio vectorial.

Modelo de espacio vectorial

Vamos a interpretar una colección de documentos como una matriz donde hay un renglón por cada término que aparece en algún lugar de la colección y cada columna representa un documento.

$$\begin{array}{c} \\ T_1 \\ T_2 \\ \vdots \\ T_m \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\ \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \end{array}$$

Modelo de espacio vectorial [Cont.]

T_i es el i-esimo término.

D_i es el i-esimo documento.

a_{ij} es la frecuencia de T_i en D_j.

$$\begin{array}{c} \\ T_1 \\ T_2 \\ \vdots \\ T_m \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\ \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \end{array}$$

Modelo de espacio vectorial [Cont.]

T_i es el i-esimo término.

D_i es el i-esimo documento.

a_{ij} es la frecuencia de T_i en D_j.

$$\begin{array}{c} \\ T_1 \\ T_2 \\ \vdots \\ T_m \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\ \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \end{array}$$

Modelo de espacio vectorial [Cont.]

T_i es el i-esimo término.

D_i es el i-esimo documento.

a_{ij} es la frecuencia de T_i en D_j.

$$
\begin{array}{c}
 \\
T_1 \\
T_2 \\
\vdots \\
T_m
\end{array}
\begin{array}{c}
\begin{array}{cccc} D_1 & D_2 & \cdots & D_n \end{array} \\
\begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
 & & & \\
a_{m1} & a_{m2} & \cdots & a_{mn}
\end{pmatrix}
\end{array}
$$

Modelo de espacio vectorial [Cont.]

Búsqueda booleana.

Una petición de busqueda:

$$\mathbf{q} = (q_1\ q_2 \ldots q_m)^t$$

$$q_i = \begin{cases} 1 \text{ si el término } T_i \text{ aparece en la petición.} \\ 0 \text{ de otra forma} \end{cases}$$

Modelo de espacio vectorial [Cont.]

Búsqueda booleana.

Entonces buscamos en el espacio columna aquellos documentos que contengan información relevante a la petición.

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos.

Es un refinamiento a la búsqueda boolena.

La idea intuitiva es que **q** estará más cerca a un vector documento si su ángulo entre ellos es pequeño.

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos.

Es un refinamiento a la búsqueda boolena.

La idea intuitiva es que **q** estará más cerca a un vector documento si su ángulo entre ellos es pequeño.

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos.

Si $\mathbf{a}_j$ es la j-ésima columna de $\mathbf{A}$, y θ_j es el ángulo que forman $\mathbf{q}$ y $\mathbf{a}_j$, entonces si $\mathbf{q}$ y $\mathbf{a}_j$ se normalizan:

$$\mathbf{q}^t\mathbf{A} = (\mathbf{cos}\ \theta_i)^t$$

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos. (Ejemplo)

Una colección de siete títulos de libros y seis palabras claves:

1. Club ***Monarcas Morelia*** en el ***futbol*** de México
2. ***Ecología*** de la ***mariposa monarca***
3. Lista de ***Monarcas*** de ***España***
4. Fundación Santuario de la ***mariposa monarca***
5. El equipo de la fuerza ***Monarcas Morelia***
6. Federación Mexicana de ***futbol*** Asociación
7. Instituto Nacional de ***Ecología***

Palabras clave (términos)

1. monarca	4. España
2. Morelia	5. mariposa
3. futbol	6. ecología

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos. (Ejemplo)

En este caso:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

una búsqueda de documentos que contienen las palabras *mariposa* y *monarca* es el vector

$$\mathbf{q} = (1\ 0\ 0\ 0\ 1\ 0)^t$$

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos. (Ejemplo)

A normalizada es:

$$A = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{3}} & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

y **q** normalizado es:

$$q = (\frac{1}{\sqrt{2}} \quad 0 \quad 0 \quad 0 \quad \frac{1}{\sqrt{2}} \quad 0)^t$$

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos. (Ejemplo)

Entonces el vector de cosenos es:

$$(\cos\theta_i)^t = \mathbf{q}^t\mathbf{A}$$

$$\mathbf{q}^t\mathbf{A} = \left(\frac{1}{\sqrt{6}} \quad \frac{2}{\sqrt{6}} \quad \frac{1}{2} \quad 1 \quad \frac{1}{2} \quad 0 \quad 0\right)^t$$

$$= (0.4082\ 0.8165\ 0.5000\ 1.0000\ 0.5000\ 0\ 0)^t$$

Asi, el resultado nos dice que el documento 4 es el más importante, seguido de los docs. 2, 5, 3 y 1.

Referencias de esta sección: [12]

Modelo de espacio vectorial [Cont.]

Proximidad por cosenos. (Ejemplo)

Entonces el vector de cosenos es:

$$(\cos\theta_i)^t = \mathbf{q}^t\mathbf{A}$$

$$\mathbf{q}^t\mathbf{A} = \left(\frac{1}{\sqrt{6}} \quad \frac{2}{\sqrt{6}} \quad \frac{1}{2} \quad 1 \quad \frac{1}{2} \quad 0 \quad 0\right)^t$$

$$= (0.4082\ 0.8165\ 0.5000\ 1.0000\ 0.5000\ 0\ 0)^t$$

Asi, el resultado nos dice que el documento 4 es el más importante, seguido de los docs. 2, 5, 3 y 1.

Referencias de esta sección: [12]

Análisis de vínculos

- Practicamente todos los motores de búsqueda web modernos usan análisis de vínculos para mejorar sus resultados de búsqueda.
- El análisis de vínculos es el uso de la estructura de hipervínculos de la red.
- Ha permitido revolucionar la búsqueda web, a tal grado que la búsqueda antes de su uso palidece en comparación a la precisión de la búsqueda actual.
- Están basados en fundamentos de teoría de matrices.

Análisis de vínculos

- Practicamente todos los motores de búsqueda web modernos usan análisis de vínculos para mejorar sus resultados de búsqueda.
- El análisis de vínculos es el uso de la estructura de hipervínculos de la red.
- Ha permitido revolucionar la búsqueda web, a tal grado que la búsqueda antes de su uso palidece en comparación a la precisión de la búsqueda actual.
- Están basados en fundamentos de teoría de matrices.

Análisis de vínculos

- Practicamente todos los motores de búsqueda web modernos usan análisis de vínculos para mejorar sus resultados de búsqueda.
- El análisis de vínculos es el uso de la estructura de hipervínculos de la red.
- Ha permitido revolucionar la búsqueda web, a tal grado que la búsqueda antes de su uso palidece en comparación a la precisión de la búsqueda actual.
- Están basados en fundamentos de teoría de matrices.

Análisis de vínculos

- Practicamente todos los motores de búsqueda web modernos usan análisis de vínculos para mejorar sus resultados de búsqueda.
- El análisis de vínculos es el uso de la estructura de hipervínculos de la red.
- Ha permitido revolucionar la búsqueda web, a tal grado que la búsqueda antes de su uso palidece en comparación a la precisión de la búsqueda actual.
- Están basados en fundamentos de teoría de matrices.

Análisis de vínculos

Los algoritmos de análisis de vinculos más populares son:

- **HITS**
- **PageRank**

Ambos fueron desarrollados alrededor de 1998.

Análisis de vínculos

Los algoritmos de análisis de vinculos más populares son:

- **HITS**
- **PageRank**

Ambos fueron desarrollados alrededor de 1998.

Análisis de vínculos

Los algoritmos de análisis de vinculos más populares son:

- **HITS**
- **PageRank**

Ambos fueron desarrollados alrededor de 1998.

Análisis de vínculos

Los algoritmos de análisis de vinculos más populares son:

- **HITS**
- **PageRank**

Ambos fueron desarrollados alrededor de 1998.

HITS

HITS: Hiper Link-induced Topic Search

- Desarrollado por John Kleinberg de la Universidad de Cornell durante sus estudios posdoctorales en IBM Almaden (1997).
- Usado por el buscador Teoma.
- Basado en la observación de un patron entre paginas web:
  - Algunas páginas sirven como *hubs* o portales, es decir, con muchos vinculos externos.
  - Otras son *autoridades* en temas por que tiene muchos enlaces que apuntan a ellas.

HITS

HITS: Hiper Link-induced Topic Search

- Desarrollado por John Kleinberg de la Universidad de Cornell durante sus estudios posdoctorales en IBM Almaden (1997).
- Usado por el buscador Teoma.
- Basado en la observación de un patron entre paginas web:
  - Algunas páginas sirven como *hubs* o portales, es decir, con muchos vinculos externos.
  - Otras son *autoridades* en temas por que tiene muchos enlaces que apuntan a ellas.

HITS

HITS: Hiper Link-induced Topic Search

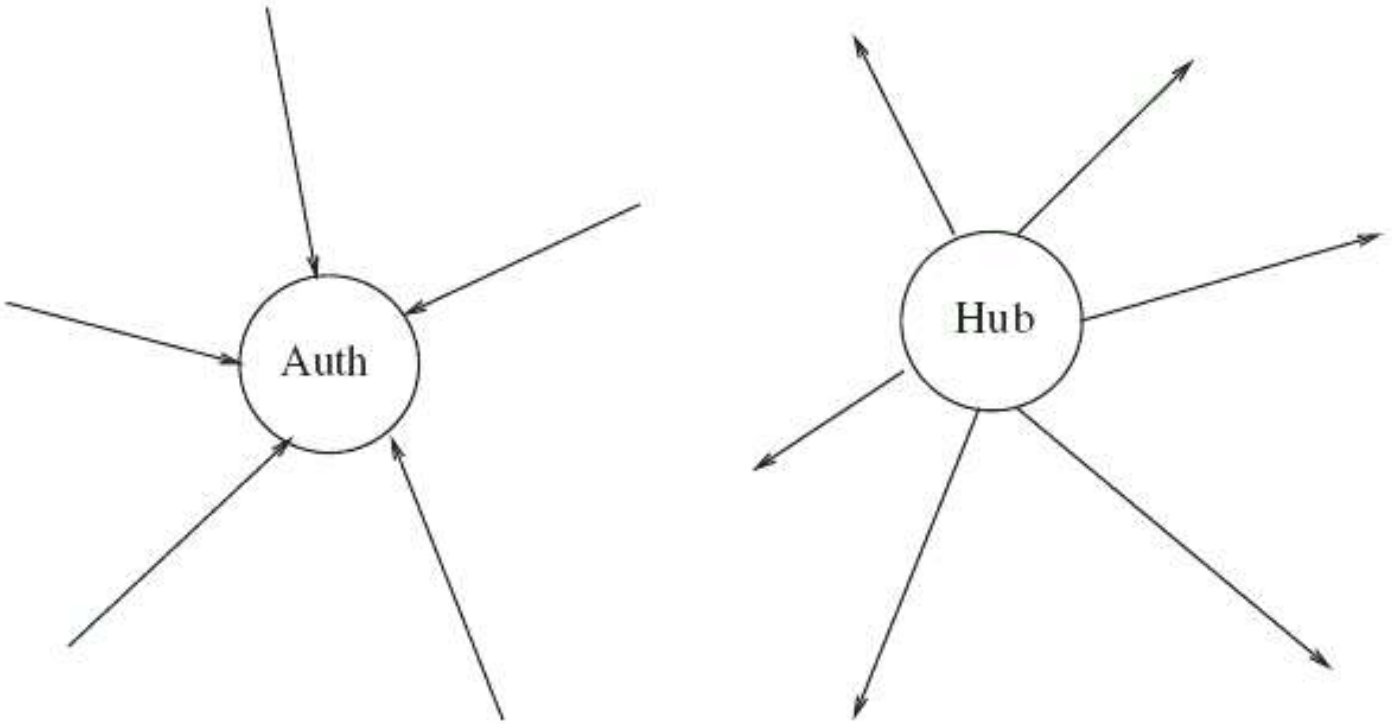
- Desarrollado por John Kleinberg de la Universidad de Cornell durante sus estudios posdoctorales en IBM Almaden (1997).
- Usado por el buscador Teoma.
- Basado en la observación de un patron entre paginas web:
  - Algunas páginas sirven como *hubs* o portales, es decir, con muchos vinculos externos.
  - Otras son *autoridades* en temas por que tiene muchos enlaces que apuntan a ellas.

HITS

HITS: Hiper Link-induced Topic Search

- Desarrollado por John Kleinberg de la Universidad de Cornell durante sus estudios posdoctorales en IBM Almaden (1997).
- Usado por el buscador Teoma.
- Basado en la observación de un patron entre paginas web:
  - Algunas páginas sirven como *hubs* o portales, es decir, con muchos vinculos externos.
  - Otras son *autoridades* en temas por que tiene muchos enlaces que apuntan a ellas.

HITS



HITS

HITS: Hiper Link-induced Topic Search

- La idea es que los buenos *hubs* apuntan a buenas *autoridades*, y las buenas *autoridades* son apuntadas por buenos *hubs*.
- Definición recursiva! Es necesario iterar...

HITS

HITS: Hiper Link-induced Topic Search

- La idea es que los buenos *hubs* apuntan a buenas *autoridades*, y las buenas *autoridades* son apuntadas por buenos *hubs*.
- Definición recursiva! Es necesario iterar...

HITS [cont.]

- Kleinberg decidio dar a cada página *i* dos calificaciones:
  - h_i: Calificación de hub.
  - a_i: Calificación de autoridad.

HITS [cont.]

- Kleinberg decidio dar a cada página *i* dos calificaciones:
  - h_i: Calificación de hub.
  - a_i: Calificación de autoridad.

HITS [cont.]

- Kleinberg decidio dar a cada página *i* dos calificaciones:
  - h_i: Calificación de hub.
  - a_i: Calificación de autoridad.

HITS [cont.]

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad y \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

- $_{eij}$ representa un enlace de la página *i* a la *j*.
- E es el conjunto de todos los enlaces.

Para calcular las calificaciones de cada página, comienza con calificaciones uniformes para todas las páginas, es decir:

$$h_i^{(1)} = 1/n \qquad a_i^{(1)} = 1/n$$

Donde *n* es el número de páginas en la vecindad de la consulta.

HITS [cont.]

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad y \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

- $_{eij}$ representa un enlace de la página *i* a la *j*.
- E es el conjunto de todos los enlaces.

Para calcular las calificaciones de cada página, comienza con calificaciones uniformes para todas las páginas, es decir:

$$h_i^{(1)} = 1/n \qquad a_i^{(1)} = 1/n$$

Donde *n* es el número de páginas en la vecindad de la consulta.

HITS [cont.]

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad y \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

- $_{eij}$ representa un enlace de la página *i* a la *j*.
- E es el conjunto de todos los enlaces.

Para calcular las calificaciones de cada página, comienza con calificaciones uniformes para todas las páginas, es decir:

$$h_i^{(1)} = 1/n \qquad\qquad a_i^{(1)} = 1/n$$

Donde *n* es el número de páginas en la vecindad de la consulta.

HITS [cont.]

La *vecindad* de una consulta consiste de las páginas en la web que contienen los términos de la consulta, mas todas las páginas que apuntan hacia o desde las páginas de este conjunto.

El conjunto inicial (o conjunto raíz) se obtiene de un buscador tradicional (tomando por ejemplo las primeras 200 respuestas del buscador).

¡El conjunto de vecinos puede ser demasiado grande! Hay que limitar la cantidad de vecinos que se añaden.

HITS [cont.]

La *vecindad* de una consulta consiste de las páginas en la web que contienen los términos de la consulta, mas todas las páginas que apuntan hacia o desde las páginas de este conjunto.

El conjunto inicial (o conjunto raíz) se obtiene de un buscador tradicional (tomando por ejemplo las primeras 200 respuestas del buscador).

¡El conjunto de vecinos puede ser demasiado grande! Hay que limitar la cantidad de vecinos que se añaden.

HITS [cont.]

La *vecindad* de una consulta consiste de las páginas en la web que contienen los términos de la consulta, mas todas las páginas que apuntan hacia o desde las páginas de este conjunto.

El conjunto inicial (o conjunto raíz) se obtiene de un buscador tradicional (tomando por ejemplo las primeras 200 respuestas del buscador).

¡El conjunto de vecinos puede ser demasiado grande! Hay que limitar la cantidad de vecinos que se añaden.

HITS [cont.]

Una vez que tenemos el conjunto vecindad, procedemos a calcular las calificaciones de *hub* y *autoridad* de cada página.

Iteramos nuestra fórmula hasta que los valores converjan.

HITS [cont.]

Usando un poco de algebra lineal, podemos reescribir nuestras fórmulas...

Sea **L** la matriz de adyacencia para el conjunto vecindad. Es decir, $\mathbf{L}_{ij} = 1$ sii la pag. *i* apunta a la pag. *j*.

La definición muestra que:

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad \text{y} \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

es equivalente a

$$\mathbf{a}^{(k)} = \mathbf{L}^t\mathbf{h}^{(k-1)} \qquad \text{y} \qquad \mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

HITS [cont.]

Usando un poco de algebra lineal, podemos reescribir nuestras fórmulas...

Sea **L** la matriz de adyacencia para el conjunto vecindad. Es decir, $\mathbf{L}_{ij} = 1$ sii la pag. *i* apunta a la pag. *j*.

La definición muestra que:

$$a_i^{(k)} = \sum_{j:\, eji \in E} h_j^{(k-1)} \qquad \text{y} \qquad h_i^{(k)} = \sum_{j:\, eij \in E} a_j^{(k)}$$

es equivalente a

$$\mathbf{a}^{(k)} = \mathbf{L}^t\mathbf{h}^{(k-1)} \qquad \text{y} \qquad \mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

HITS [cont.]

Usando un poco de algebra lineal, podemos reescribir nuestras fórmulas...

Sea **L** la matriz de adyacencia para el conjunto vecindad. Es decir, $\mathbf{L}_{ij} = 1$ sii la pag. *i* apunta a la pag. *j*.

La definición muestra que:

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad \text{y} \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

es equivalente a

$$\mathbf{a}^{(k)} = \mathbf{L}^t\mathbf{h}^{(k-1)} \qquad \text{y} \qquad \mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

HITS [cont.]

Usando un poco de algebra lineal, podemos reescribir nuestras fórmulas:

Sea **L** la matriz de adyacencia para el conjunto vecindad. Es decir, $\mathbf{L}_{ij} = 1$ sii la pag. *i* apunta a la pag. *j*.

La definición muestra que:

$$a_i^{(k)} = \Sigma_{j:\, eji \in E} h_j^{(k-1)} \qquad \text{y} \qquad h_i^{(k)} = \Sigma_{j:\, eij \in E} a_j^{(k)}$$

es equivalente a

$$\mathbf{a}^{(k)} = \mathbf{L}^t\mathbf{h}^{(k-1)} \qquad \text{y} \qquad \mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

HITS [cont.]

$$\mathbf{a}^{(k)} = \mathbf{L}^{T}\mathbf{h}^{(k-1)} \qquad \text{y} \qquad \mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

Usando un poco de álgebra tenemos:

$$\mathbf{a}^{(k)} = \mathbf{L}^{T}\mathbf{L}\mathbf{a}^{(k-1)}$$

$$\mathbf{h}^{(k)} = \mathbf{L}\mathbf{L}^{T}\mathbf{h}^{(k)}$$

HITS [cont.]

$$\mathbf{a}^{(k)} = \mathbf{L}^T\mathbf{L}\mathbf{a}^{(k-1)}$$

$$\mathbf{h}^{(k)} = \mathbf{L}\mathbf{L}^T\mathbf{h}^{(k)}$$

Estas ecuaciones dejan claro que el algoritmo de Kleinberg es realmente el **método de las potencias** aplicado a las matrices positivas $\mathbf{L}^T\mathbf{L}$ y $\mathbf{L}\mathbf{L}^T$.

$\mathbf{L}^T\mathbf{L}$ se llama la matriz de *hub* y $\mathbf{L}\mathbf{L}^T$ es la matriz de *autoridad*.

HITS [cont.]

Iteramos nuestras formulas hasta que los valores converjan:

hasta convergencia, hacer

$$\mathbf{a}^{(k)} = \mathbf{L}^t\mathbf{h}^{(k-1)}$$

$$\mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

$$k = k+1$$

normaliza $\mathbf{a}^{(k)}$ y $\mathbf{h}^{(k)}$

HITS [cont.]

Iteramos nuestras formulas hasta que los valores converjan:

hasta convergencia, hacer

$$\mathbf{a}^{(k)} = \mathbf{L}^{t}\mathbf{h}^{(k-1)}$$

$$\mathbf{h}^{(k)} = \mathbf{L}\mathbf{a}^{(k)}$$

$$k = k+1$$

normaliza $\mathbf{a}^{(k)}$ y $\mathbf{h}^{(k)}$

Método de las potencias

Es un algoritmo iterativo para calcular el *eigenvector* del *eigenvalor* más grande de una matriz.

Dada una matriz **A**, un eigenvalor λ y su vector asociado **v** son tales que:

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

Es decir, la dirección de **v** no se altera, y el eigenvalor es el factor por el cual **v** se multiplica.

Método de las potencias

Es un algoritmo iterativo para calcular el *eigenvector* del *eigenvalor* más grande de una matriz.

Dada una matriz **A**, un eigenvalor **λ** y su vector asociado **v** son tales que:

$$\mathbf{Av} = \lambda\mathbf{v}$$

Es decir, la dirección de **v** no se altera, y el eigenvalor es el factor por el cual **v** se multiplica.

Método de las potencias

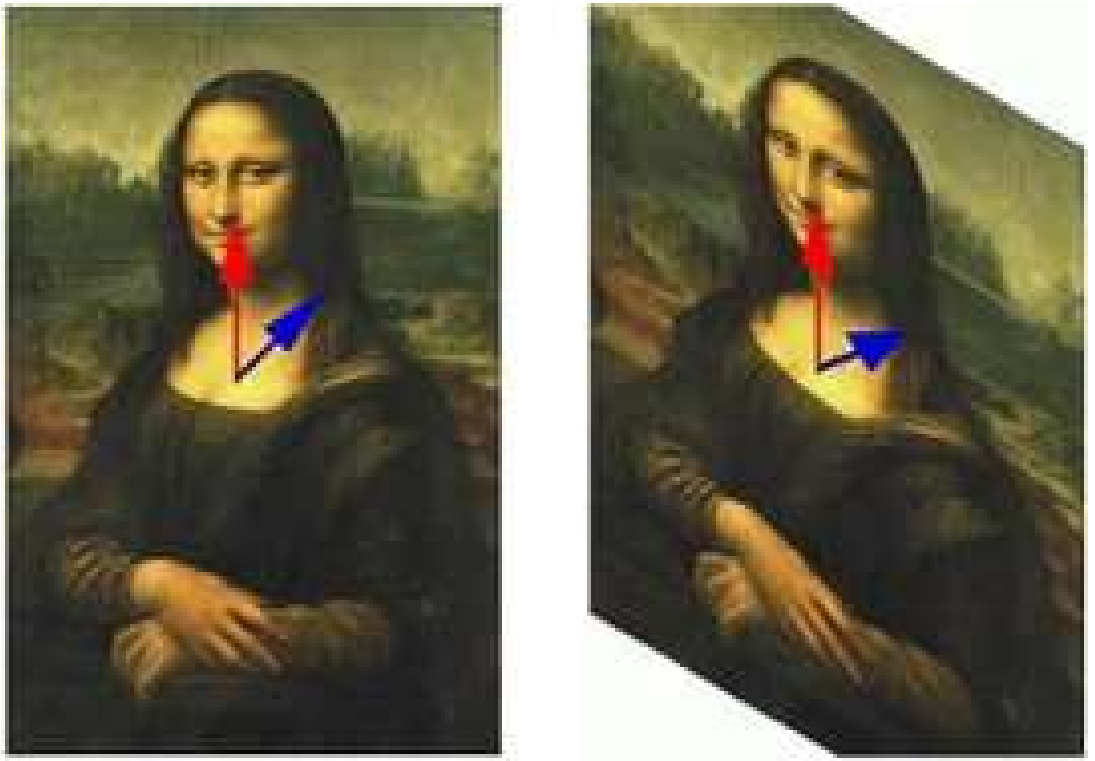
Es un algoritmo iterativo para calcular el *eigenvector* del *eigenvalor* más grande de una matriz.

Dada una matriz **A**, un eigenvalor **λ** y su vector asociado **v** son tales que:

$$\mathbf{Av} = \boldsymbol{\lambda}\mathbf{v}$$

Es decir, la dirección de **v** no se altera, y el eigenvalor es el factor por el cual **v** se multiplica.

Método de las potencias [cont.]



Ejemplo: Bajo una transformación lineal, el vector rojo no cambio su dirección, pero el azul sí. En este caso el vector rojo es un eigenvector de la T.L.

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Para calcular analiticamente los eigenvalores de una matriz A de tamaño n*n, hay que resolver el *polinomio característico* de A.

$$\mathbf{Av} = \lambda\mathbf{v} \text{ sii}$$

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \text{ sii}$$

$$\mathbf{v}(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0} \text{ sii}$$

$$\det(\mathbf{A}-\lambda\mathbf{I}) = \mathbf{0}$$

Esta última ecuación induce un polinomio de grado = n.

¡Imposible para n > 4!

Método de las potencias [cont.]

Entonces, para aproximar los eigenvalores de una matriz grande, necesitamos usar técnicas numericas.

El método de las potencias consiste en multiplicar un vector inicial $\mathbf{v}_0$ con $\mathbf{A}$, y a este producto llamarlo $\mathbf{v}_1$. Procedemos inductivamente haciendo $\mathbf{v}_{i+1} = \mathbf{A}\mathbf{v}_i$ hasta que converja a una solución estable.

Método de las potencias [cont.]

Si A cumple ciertas caracteristicas, entonces $\mathbf{v}_i$ tiende a un eigenvector de un eigenvalor dominante.

Es fácil hacer que las matrices de *hub* y de *autoridad* cumplan esos requisitos.

Método de las potencias [cont.]

Si A cumple ciertas caracteristicas, entonces $\mathbf{v}_i$ tiende a un eigenvector de un eigenvalor dominante.

Es fácil hacer que las matrices de *hub* y de *autoridad* cumplan esos requisitos.

Referencias de esta subsección: [13,14]

HITS [cont]

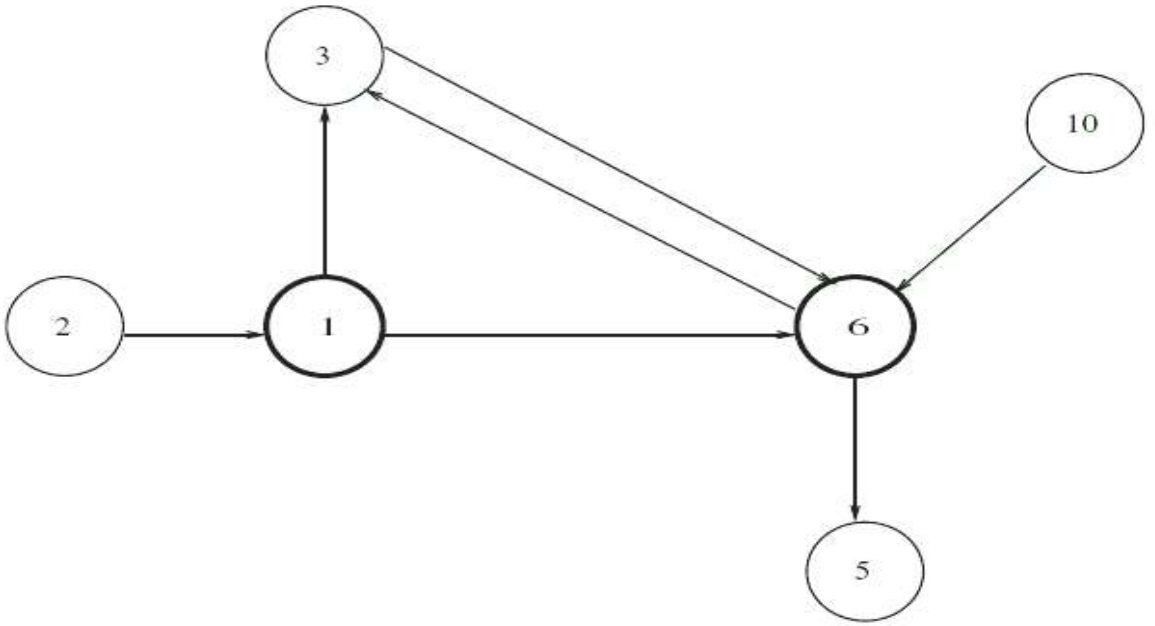
Una vez que hemos aproximado los vectores de hub y autoridad a la precisión deseada, HITS regresa al usuario las páginas de la vecindad con mejores puntuaciones en ambas categorias.

Asi, el usuario puede decidir si visitar las páginas con mayor autoridad, o las páginas que sirven mejor como portal.

HITS [Ejemplo]

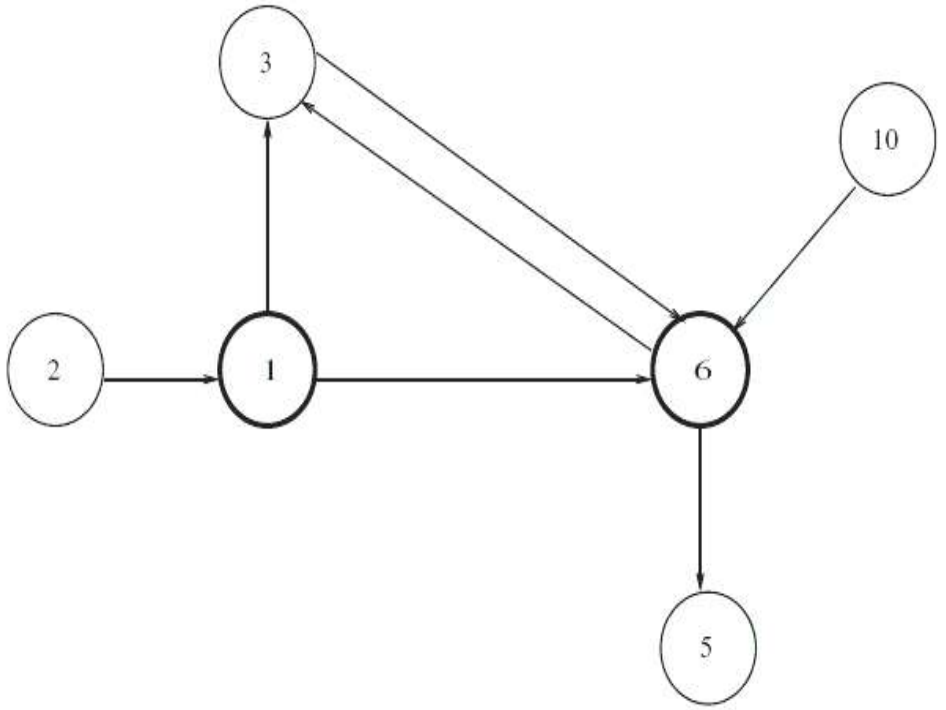
Supongamos que tenemos una consulta.

Primero identificamos a los documentos que contienen los términos de búsqueda. Supongamos para este ejemplo que son los docs. 1 y 6.



HITS [Ejemplo, cont.]

Calculamos la vecindad alrededor de los nodos 1 y 6.



$$\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}.$$

HITS [Ejemplo, cont.]

$$\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}.$$

Las matrices de hubs y autoridad son resp.

$$\mathbf{L}^T\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad \mathbf{L}\mathbf{L}^T = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \end{array}.$$

HITS [Ejemplo, cont.]

$$\mathbf{L}^T\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad \mathbf{L}\mathbf{L}^T = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \end{array}.$$

Los eigenvectores principales normalizados de ambas matrices son:

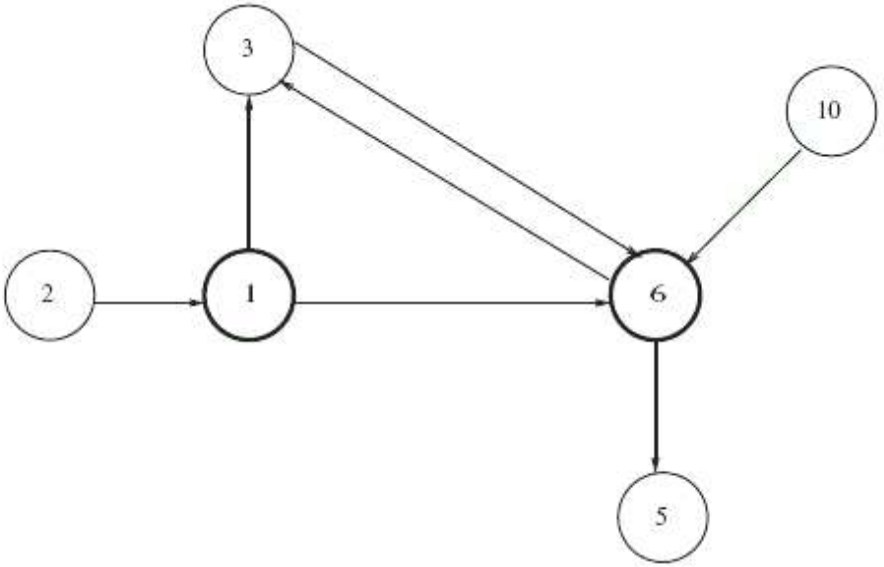
A = (0 0 .3660 .1340 .5 0)

H = (.3660 0 .2113 0 .2113 .2113)

HITS [Ejemplo, cont.]

A = (0 0 .3660 .1340 .5 0)

H = (.3660 0 .2113 0 .2113 .2113)



Entonces:

Ranking de autoridad = (6 3 5 1 2 10)

Ranking de hubs = (1 3 6 10 2 5)

HITS [cont.]

Ventajas de HITS:

- Ranking dual.
- El análisis se reduce a encontrar el eigenvector dominante de matrices pequeñas. El tamaño de estas matrices es muy pequeño comparado con la de toda la web.

HITS [cont.]

Ventajas de HITS:

- Ranking dual.
- El análisis se reduce a encontrar el eigenvector dominante de matrices pequeñas. El tamaño de estas matrices es muy pequeño comparado con la de toda la web.

HITS [cont.]

Desventajas de HITS:

- Dependencia a la consulta.
- Se debe resolver un eigenproblema por cada consulta.
- Sensible a spamming, insertando vinculos mutuos entre dos sitios relacionados.
- Desvio de tema.

HITS [cont.]

Desventajas de HITS:

- Dependencia a la consulta.
- Se debe resolver un eigenproblema por cada consulta.
- Sensible a spamming, insertando vinculos mutuos entre dos sitios relacionados.
- Desvio de tema.

HITS [cont.]

Desventajas de HITS:

- Dependencia a la consulta.
- Se debe resolver un eigenproblema por cada consulta.
- Sensible a spamming, insertando vinculos mutuos entre dos sitios relacionados.
- Desvio de tema.

HITS [cont.]

Desventajas de HITS:

- Dependencia a la consulta.
- Se debe resolver un eigenproblema por cada consulta.
- Sensible a spamming, insertando vinculos mutuos entre dos sitios relacionados.
- Desvio de tema.

Referencias sobre HITS: [1,2,3]

PageRank

Propuesto por Larry Page y Sergei Brin para RI en web mientras eran estudiantes de doctorado en Stanford (1998).

PageRank [cont.]

Idea original de Geller, N. en 1978 para su uso en bibliometría[3].

Curiosamente ese articulo no es citado por Page-Brin en sus artículos [4,5].

PageRank [cont.]

- Es un método para asignar una calificación a cada página, basado en la estructura de la gráfica de la web.
- Tiene aplicaciones en búsqueda, navegación y estimación de trafico.
- “El corazón de nuestro software es PageRank... provee la base para todas nuestras herramientas de busqueda” [Google]
- Ha sido responsable de posicionar a Google como el buscador más usado del mundo.

PageRank [cont.]

- Es un método para asignar una calificación a cada página, basado en la estructura de la gráfica de la web.
- Tiene aplicaciones en búsqueda, navegación y estimación de trafico.
- "El corazón de nuestro software es PageRank... provee la base para todas nuestras herramientas de busqueda" [Google]
- Ha sido responsable de posicionar a Google como el buscador más usado del mundo.

PageRank [cont.]

- Es un método para asignar una calificación a cada página, basado en la estructura de la gráfica de la web.
- Tiene aplicaciones en búsqueda, navegación y estimación de trafico.
- “El corazón de nuestro software es PageRank... provee la base para todas nuestras herramientas de busqueda” [Google]
- Ha sido responsable de posicionar a Google como el buscador más usado del mundo.

PageRank [cont.]

- Es un método para asignar una calificación **a cada página**, basado en la estructura de la gráfica de la web.
- Tiene aplicaciones en búsqueda, navegación y estimación de trafico.
- "El corazón de nuestro software es PageRank... provee la base para todas nuestras herramientas de busqueda" [Google]
- Ha sido responsable de posicionar a Google como el buscador más usado del mundo.

PageRank [cont.]

- Idea parecida a la de HITS..
- La idea es que un enlace de A a B se considera una recomendación de A hacia B.
- “Una página es importante si es referenciada por páginas importantes”.
- Definición recursiva...

PageRank [cont.]

- Idea parecida a la de HITS..
- La idea es que un enlace de A a B se considera una recomendación de A hacia B.
- "Una página es importante si es referenciada por páginas importantes".
- Definición recursiva...

PageRank [cont.]

- Idea parecida a la de HITS..
- La idea es que un enlace de A a B se considera una recomendación de A hacia B.
- “Una página es importante si es referenciada por páginas importantes”.
- Definición recursiva...

PageRank [cont.]

- Idea parecida a la de HITS..
- La idea es que un enlace de A a B se considera una recomendación de A hacia B.
- “Una página es importante si es referenciada por páginas importantes”.
- Definición recursiva...

PageRank [cont.]

No todas las recomendaciones son igual de importantes. Tiene más valor una recomendación de Mozilla.org que una de mi página web.

Pero si Mozilla tiene un enlace hacia mi página no se debe considerar que mi página es igual de importante!.

El valor del voto de A hacia B es la importancia de A, dividida entre el numero de enlaces de A hacia otras páginas... La importancia de A se distribuye entre todas las páginas a las que recomienda.

PageRank [cont.]

No todas las recomendaciones son igual de importantes. Tiene más valor una recomendación de Mozilla.org que una de mi página web.

Pero si Mozilla tiene un enlace hacia mi página no se debe considerar que mi página es igual de importante!.

El valor del voto de A hacia B es la importancia de A, dividida entre el numero de enlaces de A hacia otras páginas... La importancia de A se distribuye entre todas las páginas a las que recomienda.

PageRank [cont.]

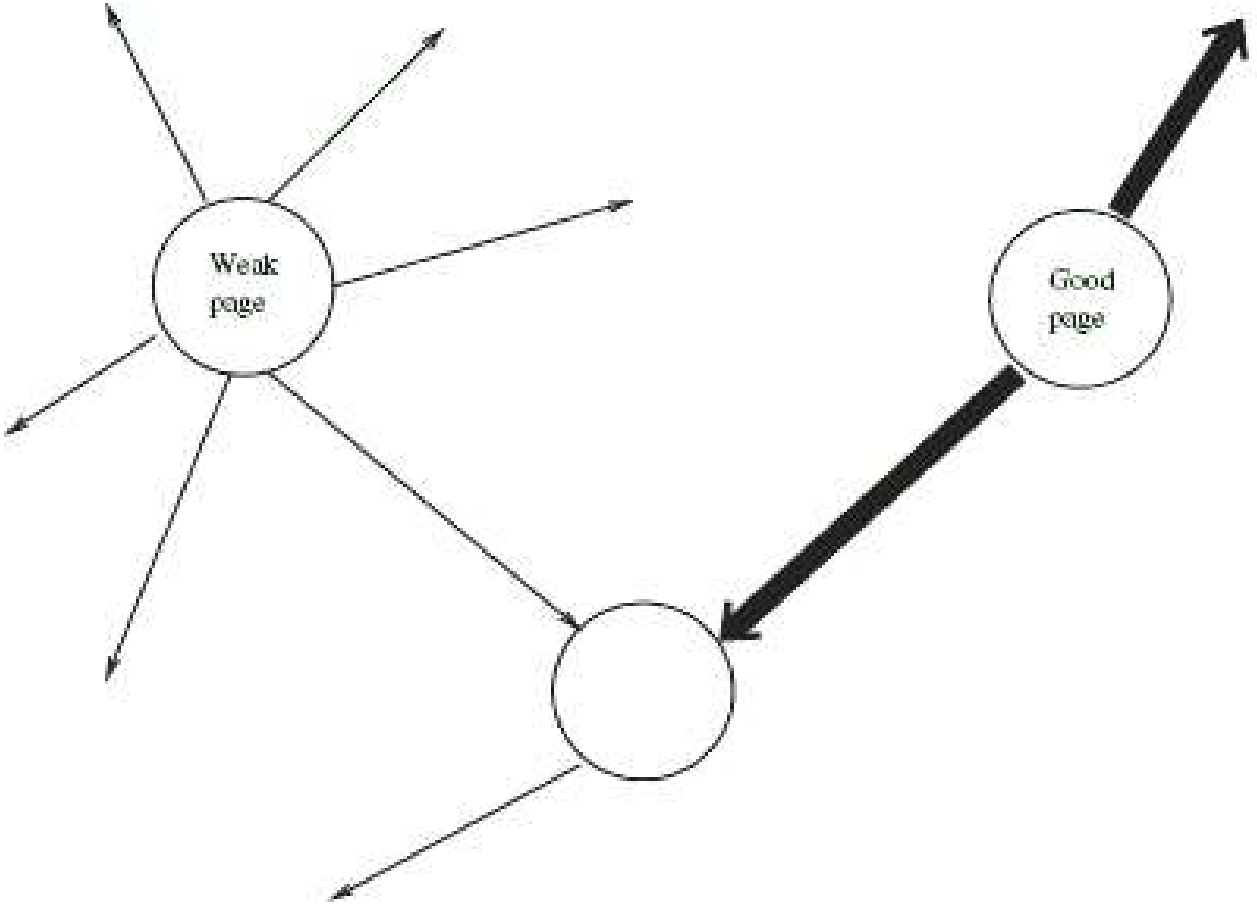
No todas las recomendaciones son igual de importantes. Tiene más valor una recomendación de Mozilla.org que una de mi página web.

Pero si Mozilla tiene un enlace hacia mi página no se debe considerar que mi página es igual de importante!.

El valor del voto de A hacia B es la importancia de A, dividida entre el numero de enlaces de A hacia otras páginas... La importancia de A se distribuye entre todas las páginas a las que recomienda.

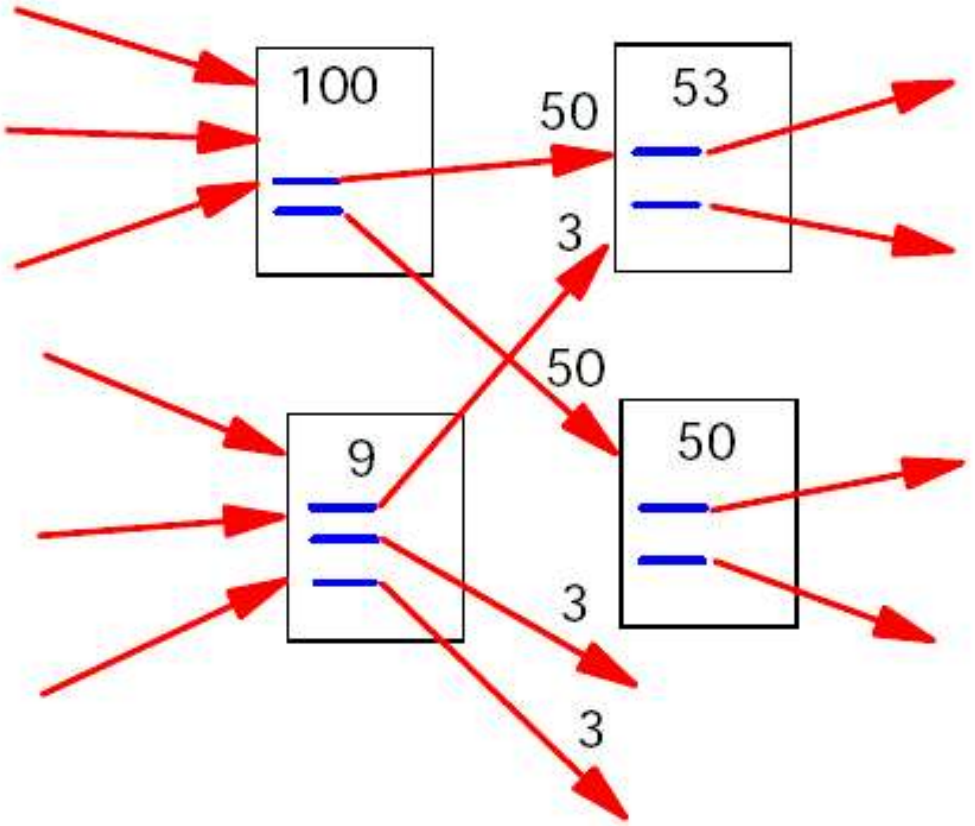
PageRank [cont.]

Ejemplo de los pesos ponderados.



PageRank [cont.]

El ranking se propaga a travez de las páginas.



PageRank [cont.]

Una formulación simplificada de PageRank es:

- Sea u una página web.
- F_u el conjunto de páginas a las que u apunta.
- B_u el conjunto de páginas que apuntan a u.
- $N_u = |F_u|$ el número de enlaces desde u.
- $c < 1$ un factor para normalización.

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}$$

PageRank [cont.]

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}$$

Aplicamos esta expresión a todas las páginas para, de forma iterativa, encontrar un valor estable.

Esto se logra haciendo $\pi_j^T = (r_j(P_1), r_j(P_2), \ldots, r_j(P_n))$ e iterativamente calculando

$$\pi_j^T = \pi_{j-1}^T \mathbf{P}$$

Donde P es la matriz con $p_{ij} = \begin{cases} 1/|P_i| \text{ si } P_i \text{ enlaza a } P_j, \\ 0 \text{ de otra forma.} \end{cases}$

PageRank [cont.]

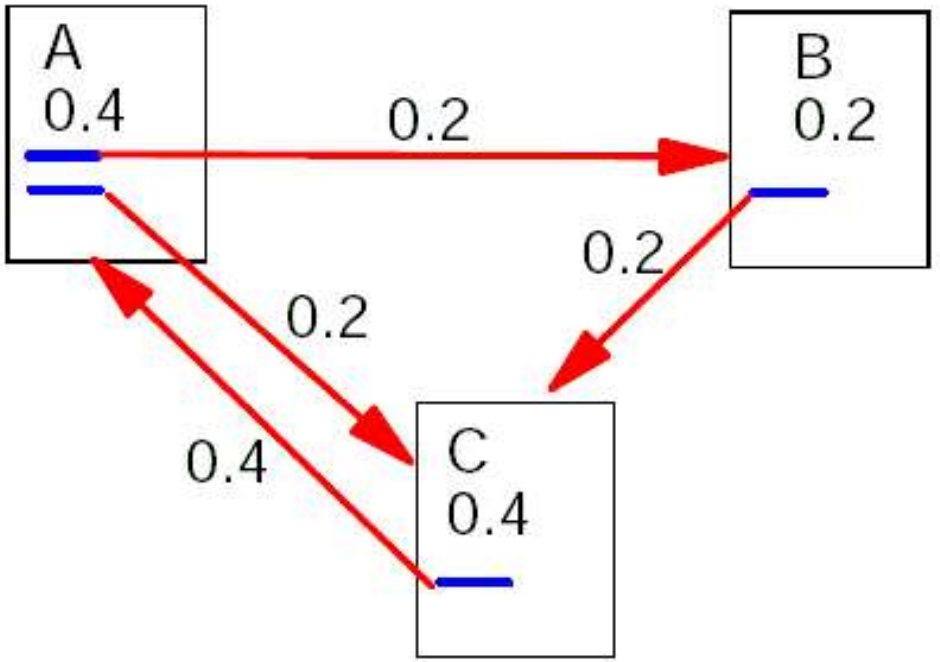
$$\pi_j^T = \pi_{j-1}^T \mathbf{P}$$

¡El método de las potencias nuevamente!

P se conoce como la matriz de Google.

PageRank [cont.]

Ejemplo de R(u) estable.



PageRank [cont.]

Hay algunos problemas con esa definición sencilla.

- c es menor que 1 por que hay algunas páginas que no tienen enlaces de salida, y su peso se pierde del sistema (nodos colgantes). Una cuarta parte del total de nodos en la web son de este tipo[1].
- Consideremos dos páginas que se enlazan entre si, pero que no tienen enlaces hacia una tercera página. Aún más, que hay una página externa con un enlace hacia una de ellas. Entonces durante la iteración, este lazo acumula calificación, pero no la distribuye por que no hay enlaces hacia afuera. A estos lazos se les conoce como *desagüe* de calificación.

PageRank [cont.]

Hay algunos problemas con esa definición sencilla.

- c es menor que 1 por que hay algunas páginas que no tienen enlaces de salida, y su peso se pierde del sistema (nodos colgantes). Una cuarta parte del total de nodos en la web son de este tipo[1].
- Consideremos dos páginas que se enlazan entre si, pero que no tienen enlaces hacia una tercera página. Aún más, que hay una página externa con un enlace hacia una de ellas. Entonces durante la iteración, este lazo acumula calificación, pero no la distribuye por que no hay enlaces hacia afuera. A estos lazos se les conoce como *desagüe* de calificación.

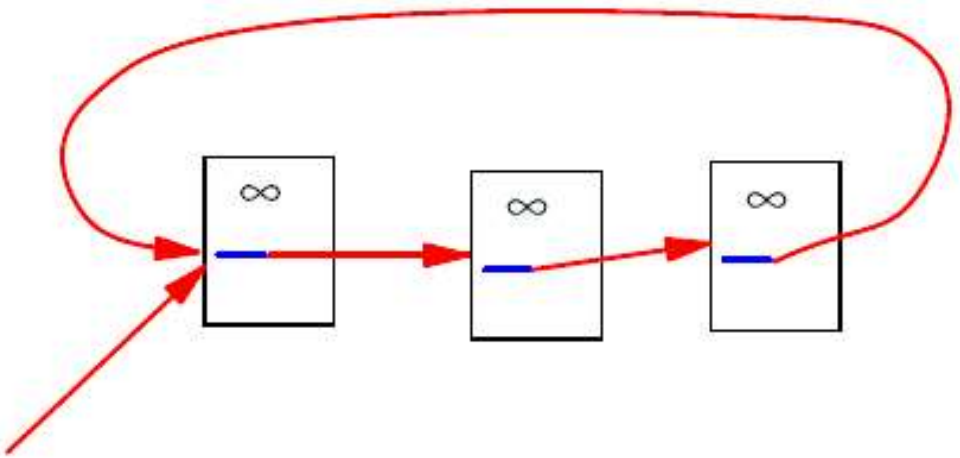
PageRank [cont.]

Hay algunos problemas con esa definición sencilla.

- c es menor que 1 por que hay algunas páginas que no tienen enlaces de salida, y su peso se pierde del sistema (nodos colgantes). Una cuarta parte del total de nodos en la web son de este tipo[1].
- Consideremos dos páginas que se enlazan entre si, pero que no tienen enlaces hacia una tercera página. Aún más, que hay una página externa con un enlace hacia una de ellas. Entonces durante la iteración, este lazo acumula calificación, pero no la distribuye por que no hay enlaces hacia afuera. A estos lazos se les conoce como *desagüe* de calificación.

PageRank [cont.]

Ejemplo de un lazo que actua como desagüe de calificación.



PageRank [cont.]

Modelo de Markov de la web

Si suponemos por un momento que no hay nodos colgados (o hacemos un arreglo artificial para evitarlos), entonces P es una matriz estocástica, lo que significa que la iteración de PageRank representa la evolución de una cadena de Markov.

Más precisamente, esta cadena de Markov es una caminata aleatoria en la gráfica definida por la estructura de los vínculos de la red.

"We compare PageRank to an idealized web surfer"[5]

PageRank [cont.]

Modelo de Markov de la web

Si suponemos por un momento que no hay nodos colgados (o hacemos un arreglo artificial para evitarlos), entonces P es una matriz estocástica, lo que significa que la iteración de PageRank representa la evolución de una cadena de Markov.

Más precisamente, esta cadena de Markov es una caminata aleatoria en la gráfica definida por la estructura de los vínculos de la red.

"We compare PageRank to an idealized web surfer"[5]

PageRank [cont.]

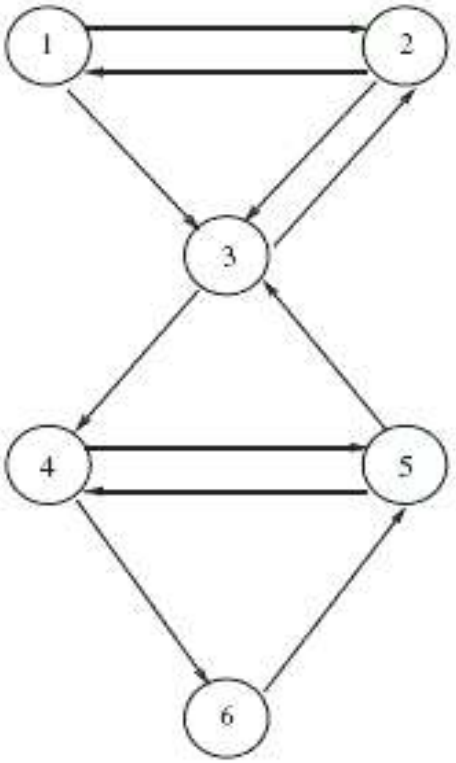
Modelo de Markov de la web

Si suponemos por un momento que no hay nodos colgados (o hacemos un arreglo artificial para evitarlos), entonces P es una matriz estocástica, lo que significa que la iteración de PageRank representa la evolución de una cadena de Markov.

Más precisamente, esta cadena de Markov es una caminata aleatoria en la gráfica definida por la estructura de los vínculos de la red.

"We compare PageRank to an idealized web surfer"[5]

PageRank [cont.]

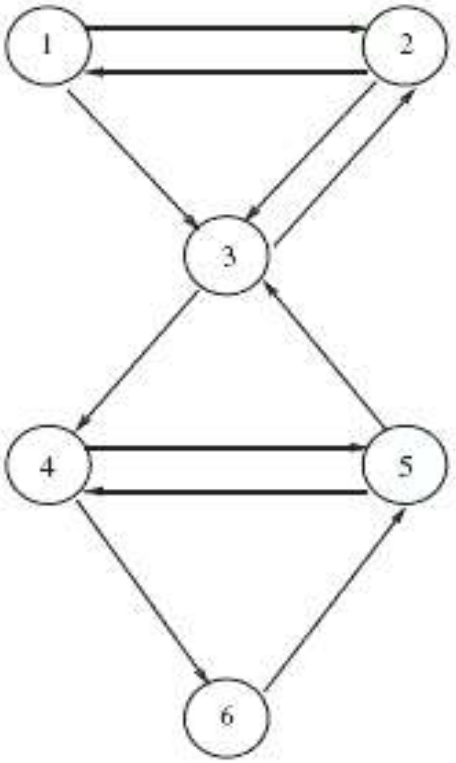


Ejemplo: Consideremos esta estructura de enlaces.

El modelo de Markov representa la gráfica dirigida como una matriz cuadrada de probabilidades de transición cuyo elemento p_{ij} es la probabilidad de moverse del estado i (pag. i) al estado j (pag. j) en un paso (click).

$$\mathbf{P} = \begin{pmatrix} 0 & .5 & .5 & 0 & 0 & 0 \\ .5 & 0 & .5 & 0 & 0 & 0 \\ 0 & .5 & 0 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & .5 & .5 \\ 0 & 0 & .5 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

PageRank [cont.]



Otra manera de interpretar el PageRank de una página es como la fracción promedio de tiempo que el navegador aleatorio estará visitando esa pagina, dado un tiempo infinito.

$$\mathbf{P} = \begin{pmatrix} 0 & .5 & .5 & 0 & 0 & 0 \\ .5 & 0 & .5 & 0 & 0 & 0 \\ 0 & .5 & 0 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & .5 & .5 \\ 0 & 0 & .5 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

PageRank [cont.]

En general, el eigenvalor dominante para cada matriz estocastica es $\lambda = 1$.

Si el vector PageRank converge, lo hara al eigenvector normalizado que satisface:

$$\pi^T = \pi^T \mathbf{P}, \quad \pi^T e = \mathbf{1} \text{ (e es una columna de unos)}$$

El problema de calcular PageRank se reduce entonces al de encontrar el eigenvector dominante, o equivalentemente resolver el sistema lineal homogeneo

$$\pi^T (\mathbf{I-P}) = 0 \text{ con } \pi^T e = \mathbf{1}$$

PageRank [cont.]

En general, el eigenvalor dominante para cada matriz estocastica es $\lambda = 1$.

Si el vector PageRank converge, lo hara al eigenvector normalizado que satisface:

$$\pi^T = \pi^T \mathbf{P}, \quad \pi^T e = \mathbf{1} \text{ (e es una columna de unos)}$$

El problema de calcular PageRank se reduce entonces al de encontrar el eigenvector dominante, o equivalentemente resolver el sistema lineal homogeneo

$$\pi^T(\mathbf{I}-\mathbf{P})=0 \text{ con } \pi^T e = \mathbf{1}$$

PageRank [cont.]

En general, el eigenvalor dominante para cada matriz estocastica es $\lambda = 1$.

Si el vector PageRank converge, lo hara al eigenvector normalizado que satisface:

$$\pi^T = \pi^T \mathbf{P}, \quad \pi^T e = \mathbf{1} \text{ (e es una columna de unos)}$$

El problema de calcular PageRank se reduce entonces al de encontrar el eigenvector dominante, o equivalentemente resolver el sistema lineal homogeneo

$$\pi^T(\mathbf{I-P})=0 \text{ con } \pi^T e = \mathbf{1}$$

PageRank [cont.]

El sistema anterior es muy sencillo, pero su solución no es trivial por el tamaño de las matrices..

Actualmente P es una matriz de mas de 64,000,000,000,000,000 términos!

De hecho se dice que calcular PageRank es “La mayor computación matricial del mundo”.

El hecho de que sea una matriz extremadamente rala es una razón para usar el metodo de las potencias.

PageRank [cont.]

El sistema anterior es muy sencillo, pero su solución no es trivial por el tamaño de las matrices..

Actualmente P es una matriz de mas de 64,000,000,000,000,000,000 términos!

De hecho se dice que calcular PageRank es “La mayor computación matricial del mundo”.

El hecho de que sea una matriz extremadamente rala es una razón para usar el metodo de las potencias.

PageRank [cont.]

El sistema anterior es muy sencillo, pero su solución no es trivial por el tamaño de las matrices..

Actualmente P es una matriz de mas de 64,000,000,000,000,000,000 términos!

De hecho se dice que calcular PageRank es “La mayor computación matricial del mundo”.

El hecho de que sea una matriz extremadamente rala es una razón para usar el metodo de las potencias.

PageRank [cont.]

El sistema anterior es muy sencillo, pero su solución no es trivial por el tamaño de las matrices..

Actualmente P es una matriz de mas de 64,000,000,000,000,000,000 términos!

De hecho se dice que calcular PageRank es “La mayor computación matricial del mundo”.

El hecho de que sea una matriz extremadamente rala es una razón para usar el metodo de las potencias.

PageRank [cont.]

Ajuste de P

La matriz de Google (P) asi definida tiene dos graves problemas:

- Existen muchas filas de solo 0's (nodos colgantes).
- Hay desagües de calificación.

Para solucionarlos hay que modificar un poco a P...

PageRank [cont.]

Ajuste de P

La matriz de Google (P) asi definida tiene dos graves problemas:

- Existen muchas filas de solo 0's (nodos colgantes).
- Hay desagües de calificación.

Para solucionarlos hay que modificar un poco a P...

PageRank [cont.]

Ajuste de P

La matriz de Google (P) asi definida tiene dos graves problemas:

- Existen muchas filas de solo 0's (nodos colgantes).
- Hay desagües de calificación.

Para solucionarlos hay que modificar un poco a P...

PageRank [cont.]

Ajuste de P

La matriz de Google (P) asi definida tiene dos graves problemas:

- Existen muchas filas de solo 0's (nodos colgantes).
- Hay desagües de calificación.

Para solucionarlos hay que modificar un poco a P...

PageRank [cont.]

- Existen muchas filas de solo 0's (nodos colgantes).

Esto se soluciona facilmente cambiando cada renglón de P que contiene solo 0's, con el renglon (1/n,1/n,...,1/n).

Entonces trabajamos ahora con la matriz P' que tiene dichos cambios.

PageRank [cont.]

- Existen muchas filas de solo 0's (nodos colgantes).

Esto se soluciona facilmente cambiando cada renglón de P que contiene solo 0's, con el renglon (1/n,1/n,...,1/n).

Entonces trabajamos ahora con la matriz P' que tiene dichos cambios.

PageRank [cont.]

- Hay desagües de calificación.

Esto se soluciona haciendo a cada página alcanzable desde cualquier otra.

La justificación intuitiva de hacerlo es la tendencia de un usuario de brincar de manera aleatoria de una pagina a cualquier otra de la red, aunque no haya un enlace de la primera a la segunda. Esto sucede p.e. cuando un usuario introduce una dirección en la barra del navegador.

Al hacer cualquier pag. alcanzable desde cualquier otra, hacemos irreducible a la matriz estocastica.

PageRank [cont.]

- Hay desagües de calificación.

Esto se soluciona haciendo a cada página alcanzable desde cualquier otra.

La justificación intuitiva de hacerlo es la tendencia de un usuario de brincar de manera aleatoria de una pagina a cualquier otra de la red, aunque no haya un enlace de la primera a la segunda. Esto sucede p.e. cuando un usuario introduce una dirección en la barra del navegador.

Al hacer cualquier pag. alcanzable desde cualquier otra, hacemos irreducible a la matriz estocastica.

PageRank [cont.]

- Hay desagües de calificación.

Esto se soluciona haciendo a cada página alcanzable desde cualquier otra.

La justificación intuitiva de hacerlo es la tendencia de un usuario de brincar de manera aleatoria de una pagina a cualquier otra de la red, aunque no haya un enlace de la primera a la segunda. Esto sucede p.e. cuando un usuario introduce una dirección en la barra del navegador.

Al hacer cualquier pag. alcanzable desde cualquier otra, hacemos irreducible a la matriz estocastica.

PageRank [cont.]

La solución a los problema anterior lleva al replanteamiento de la matriz de google.

$$P'' = \alpha P' + (1-\alpha)E$$

Donde α es un escalar entre 0 y 1 y representa la probabilidad de que un navegador siga un enlace de la página actual.

$(1-\alpha)$ es entonces la probabilidad de "teletransportarse" de la página actual a cualquier otra usando la barra de direcciones. E es una matriz del mismo tamaño que P' que representa la distribucion de probabilidades para este salto aleatorio.

PageRank [cont.]

La solución a los problema anterior lleva al replanteamiento de la matriz de google.

$$P'' = \alpha P' + (1-\alpha)E$$

Donde α es un escalar entre 0 y 1 y representa la probabilidad de que un navegador siga un enlace de la página actual.

$(1-\alpha)$ es entonces la probabilidad de "teletransportarse" de la página actual a cualquier otra usando la barra de direcciones. E es una matriz del mismo tamaño que P' que representa la distribucion de probabilidades para este salto aleatorio.

PageRank [cont.]

La solución a los problema anterior lleva al replanteamiento de la matriz de google.

$$P'' = \alpha P' + (1-\alpha)E$$

Donde α es un escalar entre 0 y 1 y representa la probabilidad de que un navegador siga un enlace de la página actual.

$(1-\alpha)$ es entonces la probabilidad de "teletransportarse" de la página actual a cualquier otra usando la barra de direcciones. E es una matriz del mismo tamaño que P' que representa la distribucion de probabilidades para este salto aleatorio.

PageRank [cont.]

Convergencia del cálculo de PageRank[1]

Ajustando α se puede controlar la velocidad de convergencia del cálculo.

Valores pequeños hacen que converja rapidamente, pero entonces se trabaja con una matriz P" muy diferente de la estructura de la web real. Diferentes valores de α pueden producir PageRank's muy diferentes.

Como se forza a la matriz a ser irreducible con la introducción de E, no hay problemas con la uniquinez de la solución. Cualquier vector de probabilidad positiva puede usarse como valor inicial.

PageRank [cont.]

Convergencia del cálculo de PageRank[1]

Ajustando α se puede controlar la velocidad de convergencia del cálculo.

Valores pequeños hacen que converja rapidamente, pero entonces se trabaja con una matriz P'' muy diferente de la estructura de la web real. Diferentes valores de α pueden producir PageRank's muy diferentes.

Como se forza a la matriz a ser irreducible con la introducción de E, no hay problemas con la uniquinez de la solución. Cualquier vector de probabilidad positiva puede usarse como valor inicial.

PageRank [cont.]

Convergencia del cálculo de PageRank[1]

Ajustando α se puede controlar la velocidad de convergencia del cálculo.

Valores pequeños hacen que converja rapidamente, pero entonces se trabaja con una matriz P" muy diferente de la estructura de la web real. Diferentes valores de α pueden producir PageRank's muy diferentes.

Como se forza a la matriz a ser irreducible con la introducción de E, no hay problemas con la uniquinez de la solución. Cualquier vector de probabilidad positiva puede usarse como valor inicial.

PageRank [cont.]

Convergencia del cálculo de PageRank[1]

- Brin y Page reportan una convergencia a niveles“razonablemente tolerables” en aprox. 52 iteraciones para una base de datos de 332 millones de enlaces.[5]
- Sugieren tambien que el factor de escala es casi lineal a log n.

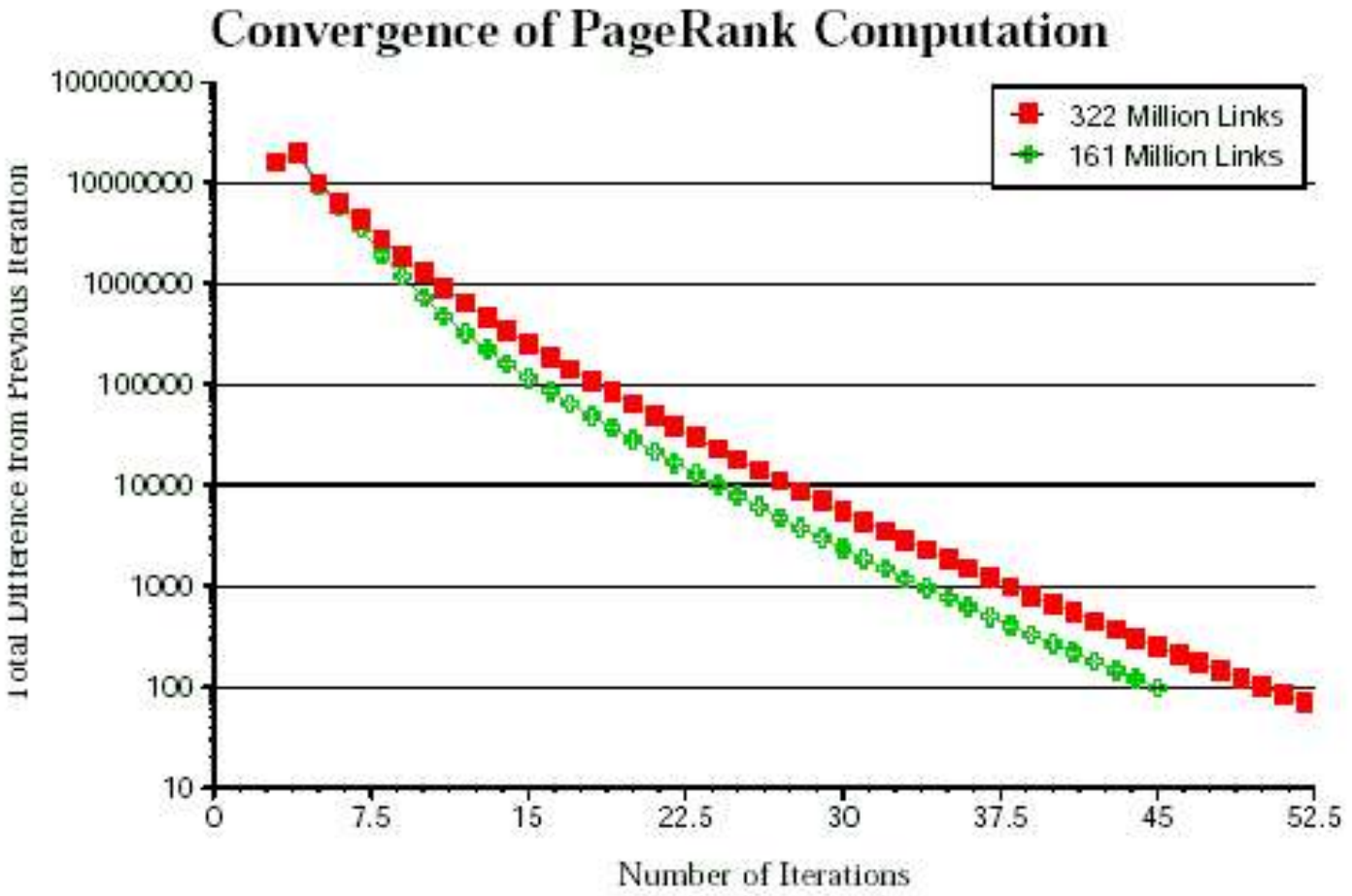
PageRank [cont.]

Convergencia del cálculo de PageRank[1]

- Brin y Page reportan una convergencia a niveles“razonablemente tolerables” en aprox. 52 iteraciones para una base de datos de 332 millones de enlaces.[5]
- Sugieren tambien que el factor de escala es casi lineal a log n.

PageRank [cont.]

[5]



PageRank [cont.]

Actualización de PageRank [1]

- La actualización del vector PageRank es muy costosa.
- Google ha reportado hacerlo cada ciertas semanas.
- El vector anterior es practicamente inutil para cálcular el nuevo.
- Google comienza practicamente de cero cada vez que actualiza PageRank

PageRank [cont.]

Actualización de PageRank [1]

- La actualización del vector PageRank es muy costosa.
- Google ha reportado hacerlo cada ciertas semanas.
- El vector anterior es practicamente inutil para cálcular el nuevo.
- Google comienza practicamente de cero cada vez que actualiza PageRank

PageRank [cont.]

Actualización de PageRank [1]

- La actualización del vector PageRank es muy costosa.
- Google ha reportado hacerlo cada ciertas semanas.
- El vector anterior es practicamente inutil para cálcular el nuevo.
- Google comienza practicamente de cero cada vez que actualiza PageRank

PageRank [cont.]

Actualización de PageRank [1]

- La actualización del vector PageRank es muy costosa.
- Google ha reportado hacerlo cada ciertas semanas.
- El vector anterior es practicamente inutil para cálcular el nuevo.
- Google comienza practicamente de cero cada vez que actualiza PageRank

PageRank [cont.]

Uso de PageRank en Google

- PageRank califica la *importancia* de una página. Pero no dice nada respecto a la relevancia de esa página respecto a alguna consulta. La calificación que PageRank da a una página es estatica.

- PageRank es solo una parte de Google, de hecho PageRank se combina con otras euristicas para formar una calificación de una página respecto a una consulta.

PageRank [cont.]

Uso de PageRank en Google

- PageRank califica la *importancia* de una página. Pero no dice nada respecto a la relevancia de esa página respecto a alguna consulta. La calificación que PageRank da a una página es estatica.

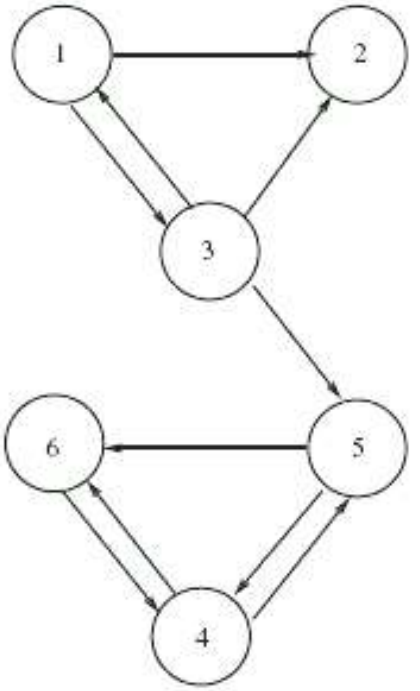
- PageRank es solo una parte de Google, de hecho PageRank se combina con otras euristicas para formar una calificación de una página respecto a una consulta.

PageRank [cont.]

Un escenario (muy simple) de uso de PageRank es el siguiente.

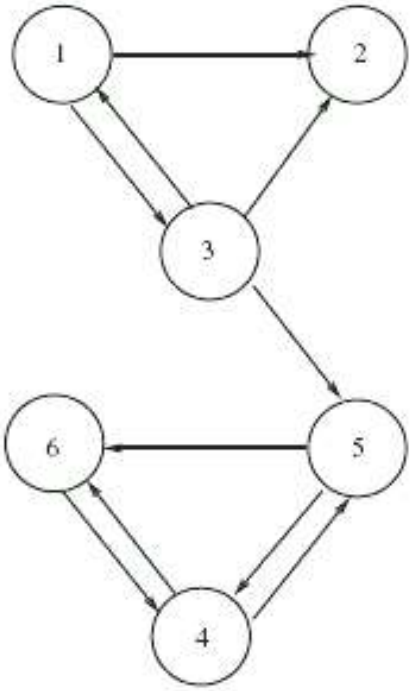
Consideremos la microred de la figura.

Primero calculamos la matriz *cruda* de Google....



PageRank [cont.]

$$\mathbf{P} = \begin{array}{c} \\ d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \end{array} \begin{array}{c} \begin{array}{cccccc} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

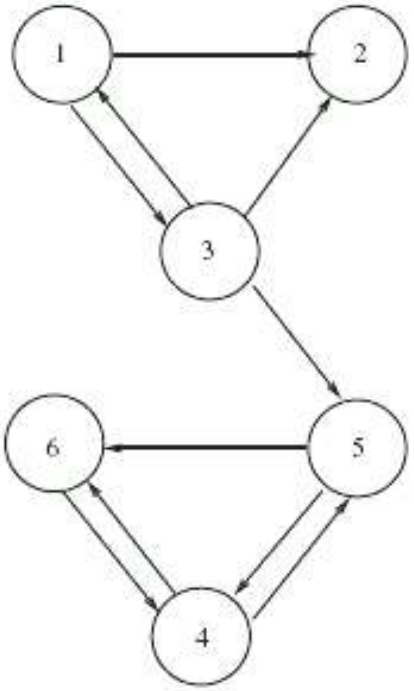


La segunda fila solo contiene ceros por que no hay enlaces de salida desde la segunda página.

Lo arreglamos añadiendo 1/6 a cada elemento de esa fila...

PageRank [cont.]

$$\bar{\mathbf{P}} = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

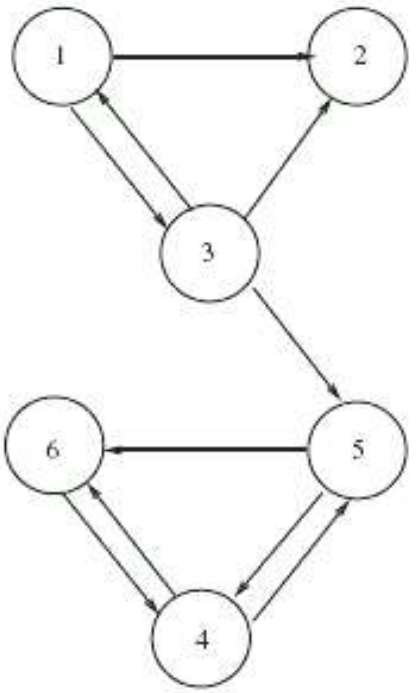


Esta matriz es estocástica, pero no irreducible.
Para forzar irreducibilidad escogemos $\alpha = 0.9$ y formamos P"...

PageRank [cont.]

$$\bar{\bar{\mathbf{P}}} = \alpha \bar{\mathbf{P}} + (1-\alpha)\mathbf{e}\mathbf{e}^T/n$$

$$= \begin{pmatrix} 1/60 & 7/15 & 7/15 & 1/60 & 1/60 & 1/60 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 19/60 & 19/60 & 1/60 & 1/60 & 19/60 & 1/60 \\ 1/60 & 1/60 & 1/60 & 1/60 & 7/15 & 7/15 \\ 1/60 & 1/60 & 1/60 & 7/15 & 1/60 & 7/15 \\ 1/60 & 1/60 & 1/60 & 11/12 & 1/60 & 1/60 \end{pmatrix}$$



Esta matriz es estocástica e irreducible, y su vector estacionario (y el vector de PageRank) es:

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

PageRank [cont.]

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

Estos PageRank's son independientes de la consulta.

Supongamos que se hace una consulta conteniendo los términos 1 y 2.

Accedemos a un indice invertido término-documento con las sig. entradas:

term1 -> doc 1, doc 4, doc 6
term2 .> doc 1, doc 3

....

....

PageRank [cont.]

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

Estos PageRank's son independientes de la consulta.

Supongamos que se hace una consulta conteniendo los términos 1 y 2.

Accedemos a un indice invertido término-documento con las sig. entradas:

term1 -> doc 1, doc 4, doc 6
term2 .> doc 1, doc 3

....

....

PageRank [cont.]

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

Estos PageRank's son independientes de la consulta.

Supongamos que se hace una consulta conteniendo los términos 1 y 2.

Accedemos a un indice invertido término-documento con las sig. entradas:

term1 -> doc 1, doc 4, doc 6
term2 .> doc 1, doc 3

....

....

PageRank [cont.]

Consulta: términos 1 y 2

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

term1 -> doc 1, doc 4, doc 6
term2 .> doc 1, doc 3
....

El conjunto de documentos relevantes es {1,3,4,6}.
Comparamos los PageRank's de esos documentos para determinar cuales de estos cuatro documentos relevantes son los más importantes, lo que da:

$\pi_4 = .3751,\ \pi_6 = .2862,\ \pi_3 = 0.4151,\ \pi_1 = .03721$

PageRank [cont.]

Consulta: términos 1 y 2

$\pi^T = (.03721\ .05396\ .04151\ .3751\ .206\ .2862\ )$

term1 -> doc 1, doc 4, doc 6
term2 .> doc 1, doc 3
....

El conjunto de documentos relevantes es {1,3,4,6}. Comparamos los PageRank's de esos documentos para determinar cuales de estos cuatro documentos relevantes son los más importantes, lo que da:

$\pi_4 = .3751,\ \pi_6 = .2862,\ \pi_3 = 0.4151,\ \pi_1 = .03721$

PageRank [cont.]

$$\pi_4 = .3751, \pi_6 = .2862, \pi_3 = 0.4151, \pi_1 = .03721$$

En consecuencia el documento 4 es el más relevante de los documentos relevantes, seguido de los documentos 6, 3, y 1.

PageRank [cont.]

$\pi_4 = .3751, \pi_6 = .2862, \pi_3 = 0.4151, \pi_1 = .03721$

En consecuencia el documento 4 es el más relevante de los documentos relevantes, seguido de los documentos 6, 3, y 1.

PageRank [cont.]

Ventajas de PageRank

- El uso de importancia, más que relevancia para calificar una página, es la clave del exito de Google. Midiendo la importancia, la dependencia de la consulta (el principal problema de HITS) no es ya un asunto de importancia.
- En tiempo de consulta, solo es necesario hacer una rápida busqueda de documentos relevantes, y ordenarlos de aucerdo a su PageRank.
- Es virtualmente inmune a spamming.

PageRank [cont.]

Ventajas de PageRank

- El uso de importancia, más que relevancia para calificar una página, es la clave del exito de Google. Midiendo la importancia, la dependencia de la consulta (el principal problema de HITS) no es ya un asunto de importancia.
- En tiempo de consulta, solo es necesario hacer una rápida busqueda de documentos relevantes, y ordenarlos de aucerdo a su PageRank.
- Es virtualmente inmune a spamming.

PageRank [cont.]

Ventajas de PageRank

- El uso de importancia, más que relevancia para calificar una página, es la clave del exito de Google. Midiendo la importancia, la dependencia de la consulta (el principal problema de HITS) no es ya un asunto de importancia.
- En tiempo de consulta, solo es necesario hacer una rápida busqueda de documentos relevantes, y ordenarlos de aucerdo a su PageRank.
- Es virtualmente inmune a spamming.

PageRank [cont.]

Ventajas de PageRank

- Flexibilidad en la selección de E, el factor de desvio. Con esto, Google puede alterar los rankings de manera predecible. Otorgando de esta manera ventajas a algunos sitios o “castigar” a quienes tratan de engañar a Google.

PageRank [cont.]

Desventajas de PageRank

- El problema de desvio de tema, debido a la necesidad de encontrar un conjunto de páginas relevantes a la consulta.
- La necesidad de aplicar extensivamente euristicas extras para determinar páginas realmente relevantes, de otra manera regresar páginas importantes no sirve de nada si estan fuera del tema. "Como PageRank es independiente de la consulta, no puede distinguir entre páginas que son autoritativas en general, y las que los son en el tema particular de la consulta".

PageRank [cont.]

Desventajas de PageRank

- El problema de desvio de tema, debido a la necesidad de encontrar un conjunto de páginas relevantes a la consulta.
- La necesidad de aplicar extensivamente euristicas extras para determinar páginas realmente relevantes, de otra manera regresar páginas importantes no sirve de nada si estan fuera del tema. "Como PageRank es independiente de la consulta, no puede distinguir entre páginas que son autoritativas en general, y las que los son en el tema particular de la consulta".

Referencias

[1] A Survey of Eigenvector Methods for Web Information Ratrieval. Amy N. Langville, Carl D. Meyer. 2005. SIAM Review Vo. 47, No. 1. pp. 135-161

[2]The Use of the Linear Algebra by Web Search Engines. Amy N. Langville, Carl D. Meyer. 2004.

[3]Authoritative Sources in a Hiperlinked Environment. Jon M. Kleinberg. Journal of the ACM, vol. 46, No. 5, 1999, pp. 604-632

[4]The Anatomy of a Large-Scale Hypertextual Web Search Engine. S Brin, L Page - WWW7 / Computer Networks, 1998 - kulturinformatik.uni-lueneburg.de

[5]The pagerank citation ranking: Bringing order to the web. L Page, S Brin, R Motwani, T Winograd - 1998 – dbpubs.stanford.edu

[6]Web Information Resource Discovery: Past, Present, and Future. G Oezsoyoglu, A Al-Hamdani - Yazici, Adnan – art.cwru.edu

[7]Web search engines. C Schwartz - Journal of the American Society for Information Science 49(11), 1998.

[8]Hyperlink analysis for the Web. MR Henzinger - IEEE Internet Computing, 2001 – ieeexplore.ieee.org

[9]The indexable web is more than 11.5 billion pages. A Gulli, A Signorini – Proceedings of the 14th international conference on World Wide Web. ACM 2005.

[10]Breadth-first search crawling yields high-quality pages. M Najork, JL Wiener - Proceedings of the 10th International World Wide Web …, 2001

[11] [Ding et al., 2004] Li Ding, Tim Finin, and Anupam Joshi. Swoogle: A search and metadata engine for the semantic web. In Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management, pages 58–61, Washington DC, November 2004.

[12]Aplicación de las factorizaciones QR y SVD a motores de búsqueda. Humberto M. de la Vega, I. Delia G. Calvillo. Universidad Autónoma de Coahuila.

[13]Eigenvalue algorithm. http://en.wikipedia.org/wiki/Eigenvalue_algorithm

[14]Eigenvalue, eigenvector and eigenspace. http://en.wikipedia.org/wiki/Eigenvalue

¡Gracias de nuevo!