

TEMA 6. BÚSQUEDA EN LA WEB

Contenidos basados en el material del curso de Manning



Contenidos

1. Introducción a la RI en la web
2. Publicidad en los buscadores
3. Detección de contenidos duplicados
4. Web Crawler
5. La web como un grafo dirigido
6. Uso del texto de los enlaces
7. Page-Rank
8. HITS

Bibliografía

A Introduction to Information Retrieval:

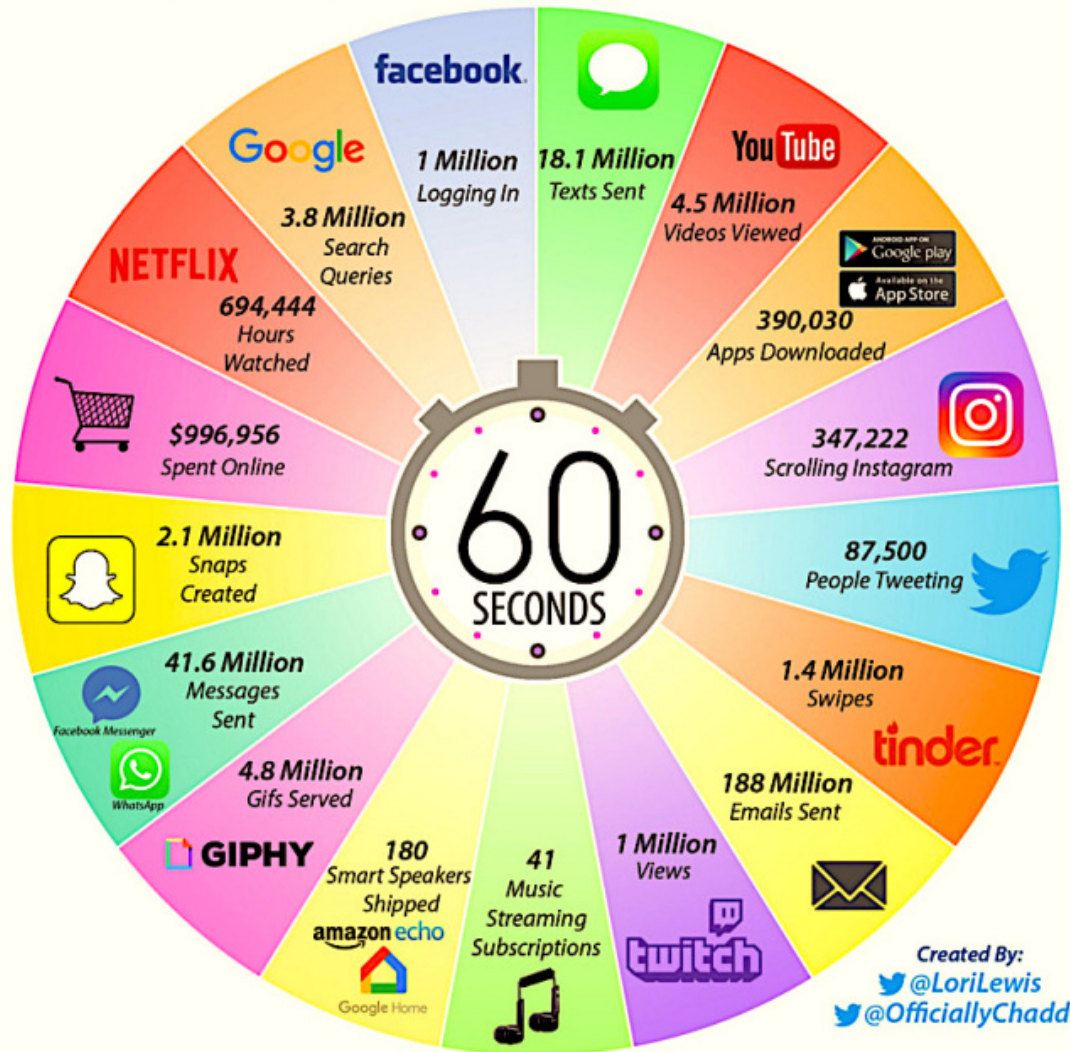
Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze.
Cambridge University Press, **2009**.

Capítulos 19, 20 y 21

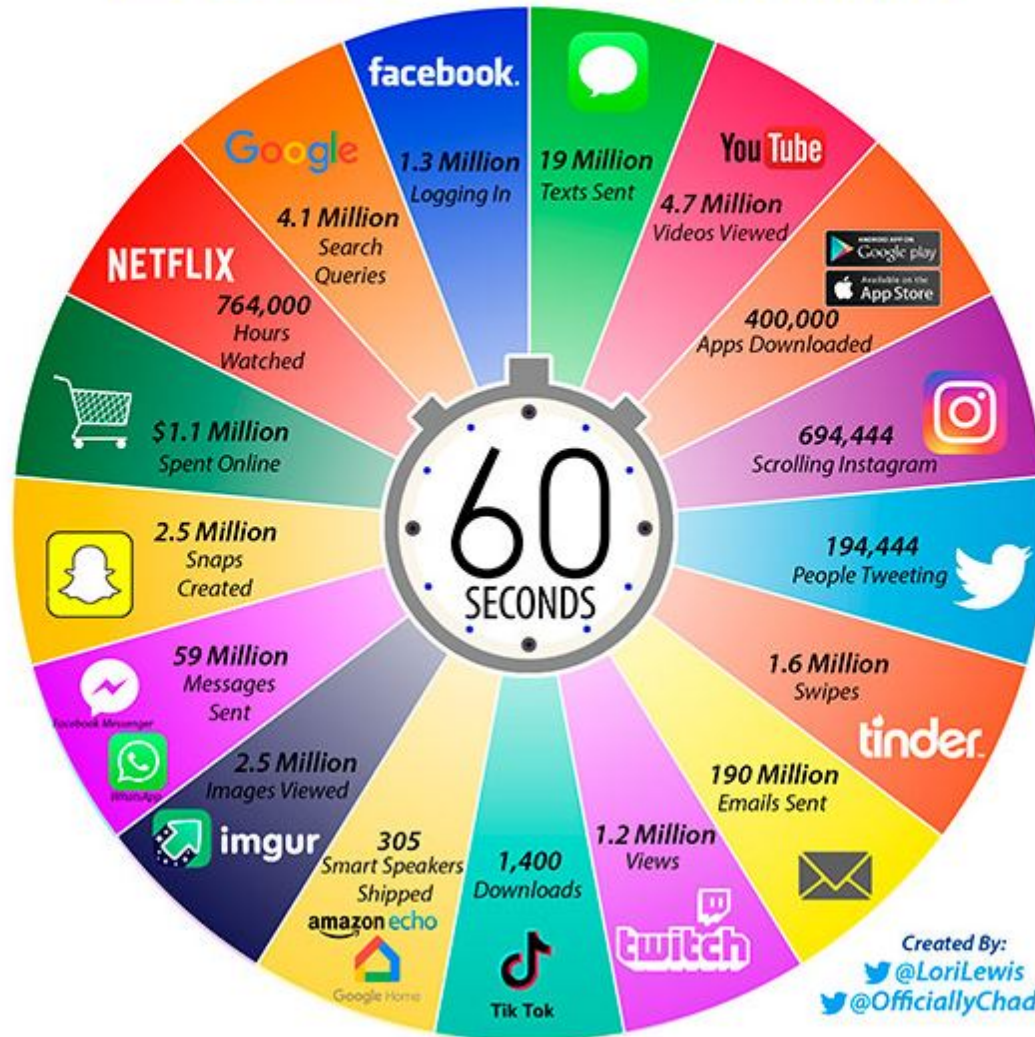


1. INTRODUCCIÓN A LA RI EN LA WEB

2019 *This Is What Happens In An Internet Minute*



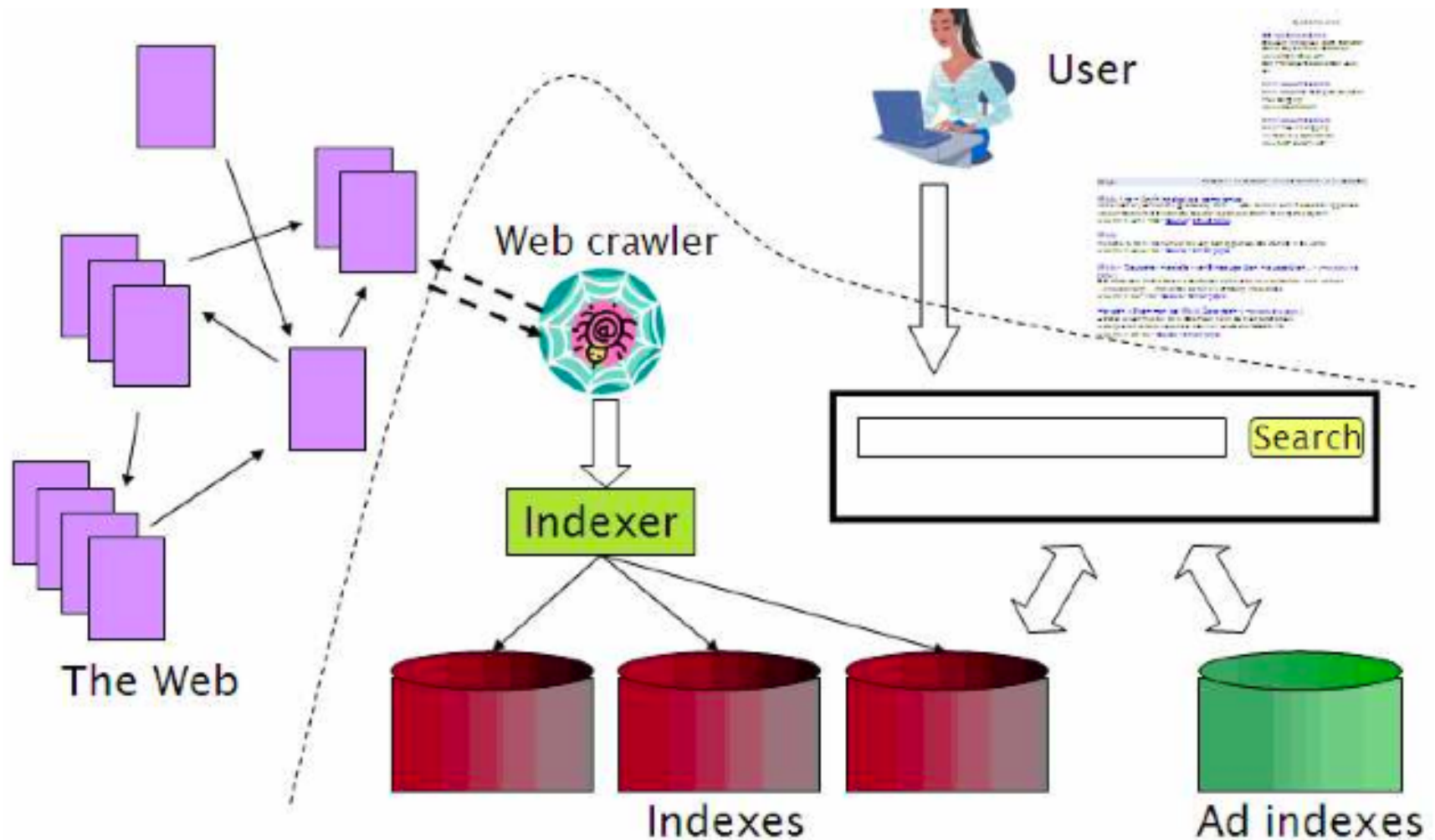
2020 *This Is What Happens In An Internet Minute*



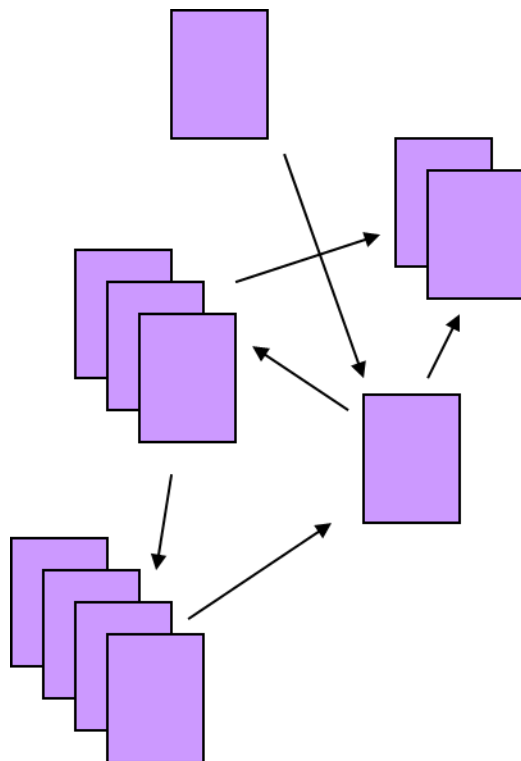
2021 *This Is What Happens In An Internet Minute*



Esquema de la RI en la web



La colección de documentos web



The Web

- No hay ninguna coordinación en el diseño.
- Creación de contenido distribuido, todo el mundo puede crear contenido.
- Presencia de cientos de lenguas (ausencia de muchas otras)
- El contenido incluye verdades, mentiras, información obsoleta, contradicciones, ...
- Información no estructurada (text, html, ...), semi-estructurada (XML, annotated photos), estructurada (Databases)...
- Una escala mucho mayor que las colecciones anteriores...
- En continua expansión
- El contenido puede ser generado dinámicamente.

La web necesita buscadores

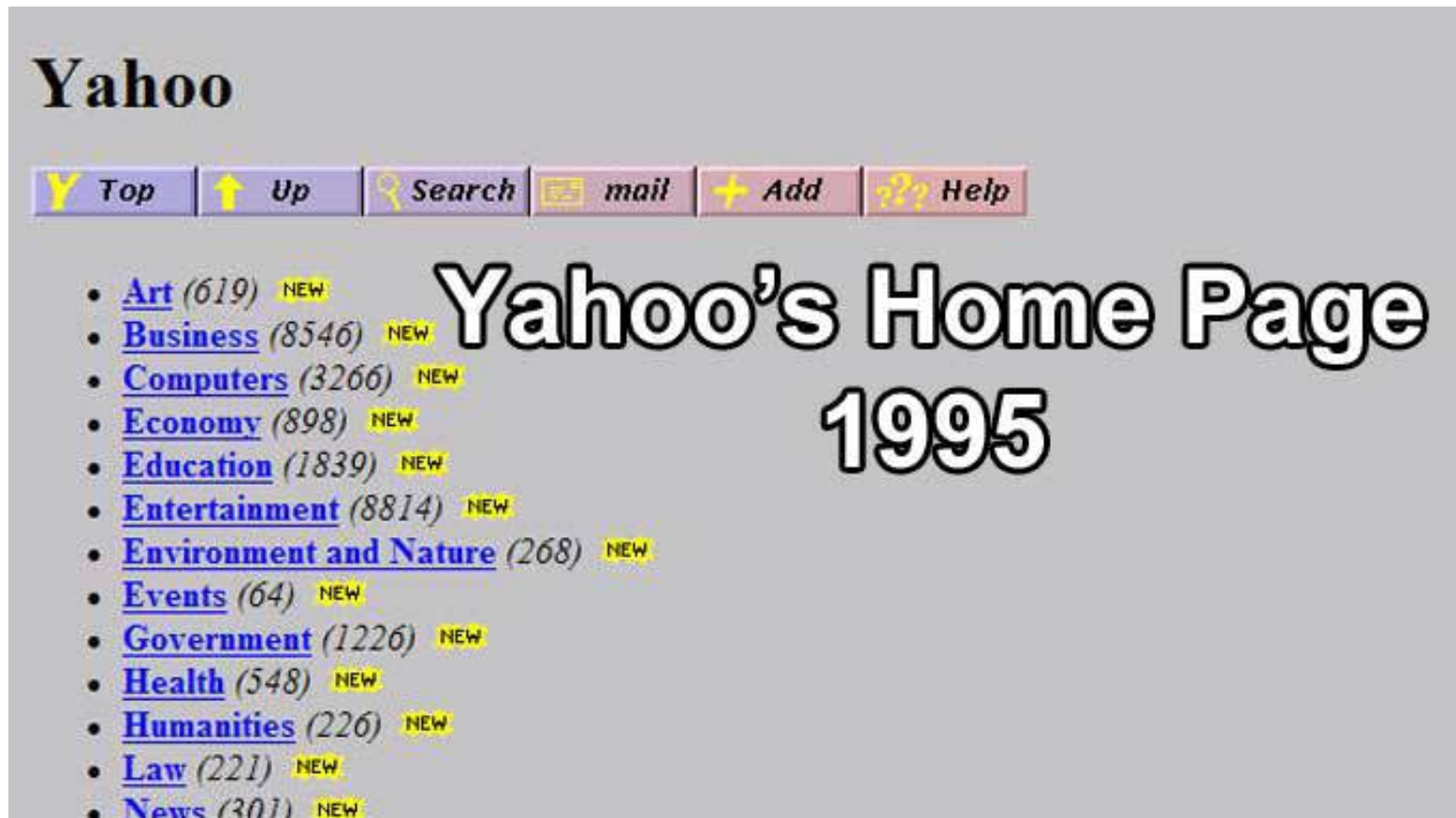
- Sin un buscador sería muy difícil encontrar “nuevas” páginas web.
- Si las webs no se encuentran no hay incentivo para crear contenido.
 - ¿Por qué publicar algo si nadie va a leerlo?
 - ¿Por qué publicar algo si no obtendremos ingresos (por publicidad) por ello?
- La web es gratis pero cuesta mucho dinero. Alguien tiene que pagar.
 - Servidores, infraestructura web, creación de contenido, ...
- La publicidad en las búsquedas (y en las páginas web) está pagando gran parte de la web.



Primeros intentos de hacer visible la web a los usuarios

- Basados en el uso de taxonomías.
 - Agrupaban las webs en categorías.
 - **Yahoo! (dir.yahoo.com), about.com, [Open Directory Project](#).**
 - No pretendían clasificar toda la web. Sólo las “mejores” web de cada categoría.
- Basados en técnicas clásicas de recuperación de información.
 - **Altavista, Excite, Infoseek.**
 - Intenta indexar toda la web.

Primeros intentos de hacer visible la web a los usuarios

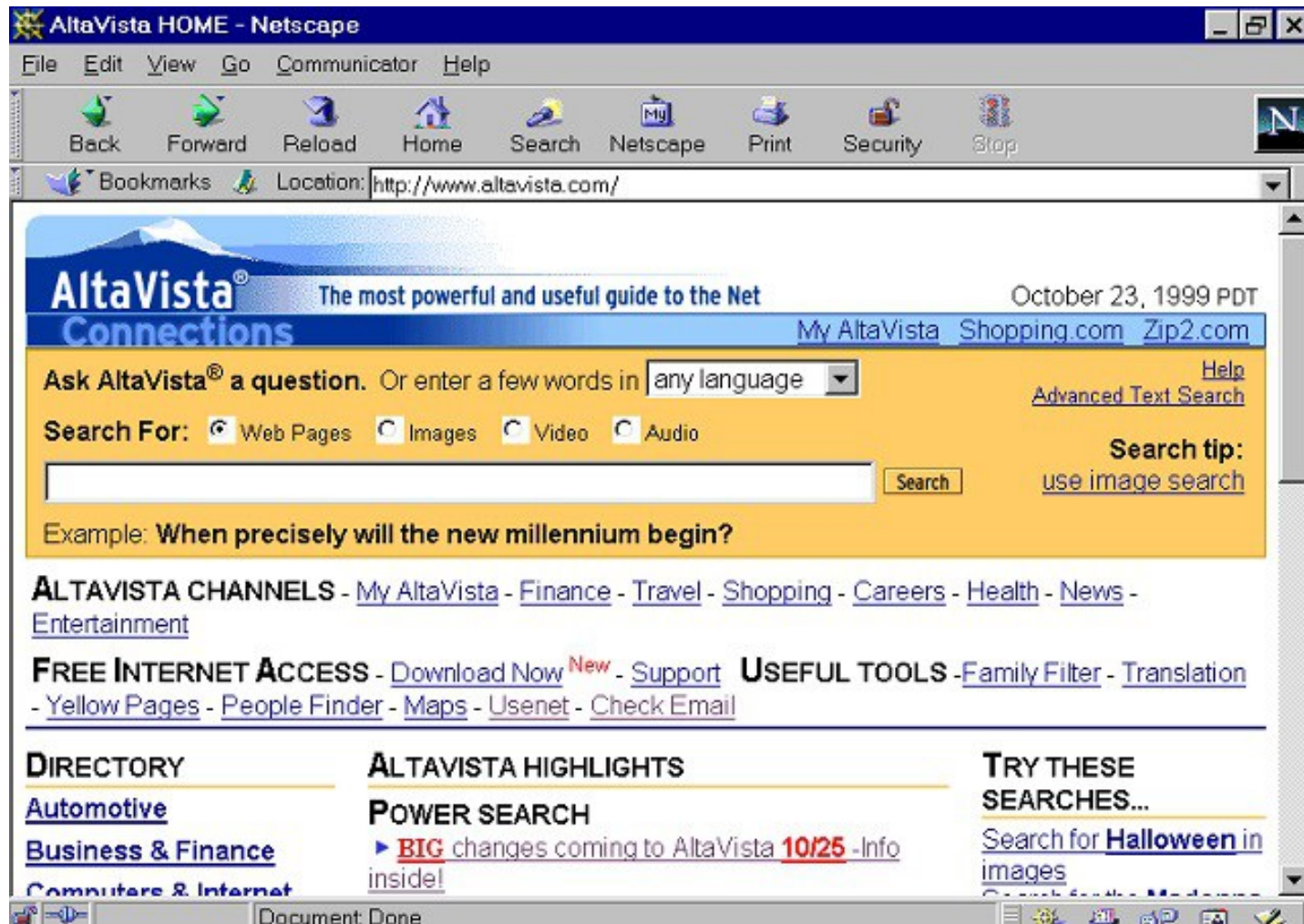




Primeros intentos de hacer visible la web a los usuarios

- Basados en el uso de taxonomías.
 - **Yahoo!**, **about.com**, [Open Directory Project](#).
 - Agrupaban las webs en categorías.
 - No pretendían clasificar toda la web. Sólo las “mejores” web de cada categoría. **Yahoo!** tenía más de 10.000 categorías distintas.
 - Necesitaban un proceso manual de edición: “descubrimiento” de contenidos y elección de la categoría dentro de la taxonomía.
 - El usuario sólo “encontraba” la web si la buscaba en la categoría en la que el editor decidía clasificarla.

Primeros intentos de hacer visible la web a los usuarios





Primeros intentos de hacer visible la web a los usuarios

- Basados en técnicas clásicas de recuperación de información.
 - **Altavista, Excite, Infoseek.**
 - Intenta indexar toda la web.
 - Centrar los esfuerzos en solucionar problemas de escala: millones de documentos a indexar.
 - Los resultados de las búsquedas no eran buenos.
 - Necesidad de nuevas medidas de ranking de los resultados.



2. LA PUBLICIDAD EN LOS BUSCADORES

1ª generación de buscadores: resultados guiados por la publicidad

www.goto.com/d/search/?sessionid\$AC42T4AAAH0R5QFIEF3QPUQ?type=home&tr=1&Keywords=Wilmington+

Wilmington real estate.

Access 75% of all users now!
Premium Listings reach 75% of all
Internet users. [Sign up](#) for Premium
Listings today!

1. [Wilmington Real Estate - Buddy Blake](#)
Wilmington's information and real estate guide. This is your on
anything to do with Wilmington.
[www.buddyblake.com](#) (Cost to advertiser: **\$10.28**)
2. [Coldwell Banker Sea Coast Realty](#)
Wilmington's number one real estate company.
[www.cbseacoast.com](#) (Cost to advertiser: **\$10.37**)
3. [Wilmington, NC Real Estate Becky Bullard](#)
Everything you need to know about buying or selling a home c
on my Web site!
[www.iwwc.net](#) (Cost to advertiser: **\$10.35**)

www.goto.com (1996)

1ª generación de buscadores: resultados guiados por la publicidad



- Buddy Blake hizo la puja más alta (\$0,38) por esta búsqueda.
- Pagaba \$0,38 cada vez que alguien hacía clic en el enlace.
- Páginas se clasificaban según lo que los anunciantes pagaban a **goto**.
- No hay separación entre los resultados y la publicidad
- No hay ranking de relevancia. Pero **goto** era honesto.



2ª generación de buscadores: separación entre resultados y publicidad

- Los buscadores separan en diferentes listas:
 - Resultados de las búsquedas, *algorithmic search results*
 - La publicidad de los anunciantes, *sponsored search results*

El ser anunciante de un buscador no proporciona mayor prioridad a la hora de aparecer más alto en los resultados del buscador.

2ª generación de buscadores: separación entre resultados y publicidad

Google search results for "hoteles en valencia". The page shows organic search results on the left and sponsored advertisements on the right. A red diagonal line separates the two sections. Blue arrows point from the text on the right to specific elements in the search results.

Organic Results (Left):

- 70 Hoteles a València - Reserva el teu Hotel a València
www.booking.com/Valencia-Hoteles
Cerca i reserva hotels online.
Booking.com té 1.066.080 seguidors a Google+
- Hotels més populars
Hotels econòmics
- Hotels millor puntuats
Hotels de luxe
- 475 Hoteles en Valencia - trivago.es
www.trivago.es/Hoteles-Valencia
trivago™ Compara Hoteles hasta-78%. Hoteles Baratos, Compara y Ahorra.
Hotel? trivago té 16.633 seguidors a Google+
- Hoteles en Valencia - Reserva Hoteles. 78% Dto
hotel.edreams.es/valencia
Date Prisa: Oferta Últimas Plazas!
Hoteles en Barcelona - Hoteles en Girona - Hoteles en Santander
- Hoteles en Valencia - Atrapalo.com
www.atrapalo.com/hoteles/.../valencia/valencia/ Tradueix aquesta pàgina
Reserva los mejores hoteles en Valencia, encuentra nuestras ofertas en Valencia usando el buscador de hoteles. Aprovecha las opiniones de nuestros ...
Hoteles en Valencia provincia - Hoteles de 4 estrellas - Gándia - 5 Estrellas
- Hoteles en Valencia desde 34€ - Rumbo
www.rumbo.es/.../Valencia Tradueix aquesta pàgina
Hoteles en Valencia. Encuentra tu alojamiento en Valencia entre más de 65.000 hoteles. Tu hotel siempre al mejor precio. Y pon Rumbo a un viaje perfecto!.
- Hoteles recomendados en Valencia - Booking.com
www.booking.com/city/es/valencia.es.html Tradueix aquesta pàgina
Reserva online y consigue fantásticos descuentos en hoteles de Valencia, España. Buena disponibilidad, excelentes precios. Lee comentarios de clientes y ...
Confortel Aqua 4 - Port Saplaya - Apartamentos en Valencia - Puerto de Valencia

Sponsored Results (Right):

- Hotel en Valencia 34€
www.rumbo.es/hoteles-Valencia
Ofertas de hotel insustitibles.
Reserva Online. Precios Exclusivos
Rumbo té 122 seguidors a Google+
- Hoteles en Valencia 43€
www.lastminute.com/Hotel-Valencia
Los Mejores Hoteles 4* en Valencia desde 43€ ¡Reserva Ya y Ahorra!
- Hoteles centro Valencia
www.melia.com/Hoteles-Centro-Valencia
Hoteles bien comunicados en pleno centro al mejor precio garantizado!
- Hotel Barceló en Valencia
www.barcelo.com/Valencia
Disfrute de unos días increíbles en Valencia desde 55€. Barceló.com
Barceló Hotels & Resorts té 151 seguidors a Google+
- Hoteles en Valencia
www.hoteles-catalonia.com/Valencia
Descubre nuestras ofertas a precios muy económicos
C/ Barcelonina 5, Valencia 963 51 46 12 - Indicacions
- Hotel Gran Valencia 4*

booking y rumbo aparece en los resultados de la búsqueda

booking y rumbo aparecen en la publicidad

¿Ranquean mejor los buscadores a los anunciantes?

Todos los buscadores dicen que **NO**



¿Cómo se ordena la publicidad?

- Los anunciantes pujan por términos de búsqueda.
- Es un sistema abierto: Cualquier persona puede “comprar” de forma no exclusiva un término.
- Diferentes formas de pago:
 - **Cost Per Click (CPC)**: El anunciante paga sólo cuando los usuarios hacen clic en el enlace.
 - **Cost Per Mille (CPM)**: El anunciante paga una cantidad por cada 1000 impresiones de su anuncio en la página de resultados (SERP).
 - **Cost Per Action (CPA)**: El anunciante paga en función de los usuarios que realmente finalizan una acción: comprar, registrarse, ...

adwords.google.com
bingads.microsoft.com



¿Cómo se ordena la publicidad?

- En cada búsqueda de los usuarios el buscador decide qué anuncios aparecerán y en qué orden.
- La elección no depende sólo de la puja del anunciante.
- Se tiene en cuenta la **relevancia** del anuncio.
- Factores que intervienen en la relevancia: la query, la zona horaria, la ubicación de usuario, la velocidad de carga de la página, ...
- Una medida de la relevancia de un anuncio, el ratio de cliqueo **Click-Through Ratio** (CTR).

$$CTR = \frac{\text{número de clics}}{\text{número de impresiones}} \cdot 100$$

- El CTR varía por muchos factores (0.1% o 0.3% se puede considerar normal).



¿Cómo se ordena la publicidad?

- En cada búsqueda de los usuarios el buscador decide qué anuncios aparecerán y en qué orden.
- Para los anunciantes, entender cómo los buscadores deciden este ranking, y qué pujas hacer sobre las diferentes keywords y a los diferentes sponsored search engines, ha pasado a ser una profesión conocida como *Search Engine Marketing* (SEM).
- De forma paralela se define *Search Engine Optimization* (SEO) como la disciplina que se encarga del posicionamiento de un sitio web en los buscadores, con el objetivo de mejorar su visibilidad. En este caso hablamos de los resultados de la búsqueda algorítmica en los índices de las páginas web.

Publicidad en los buscadores

- Los buscadores reciben la gran mayoría de ingresos por publicidad.
- En el caso más general el anunciante sólo paga cada vez que el usuario hace clic a su enlace.
- El usuario sólo hace clic si está interesado en el anuncio.
- Los buscadores castigan anuncios fraudulentos e irrelevantes y presentan los más atractivos en las primeras posiciones.
- El usuario normalmente queda satisfechos al hacer clic en un anuncio.
- El anunciante encuentra a nuevos clientes de manera rentable.
- Todos contentos...

3. DETECCIÓN DE CONTENIDOS DUPLICADOS



Detección de duplicados

La web esta llena de contenidos duplicados.

Muchos más duplicados que en cualquier otra colección de documentos.

- Duplicados exactos.
 - Fáciles de detectar.
 - Hash o fingerprints
- Documentos casi iguales (near-duplicates).
 - Muy abundantes en la web.
 - Difíciles de detectar.

Es importante que documentos casi idénticos no aparezcan juntos en los resultados de una búsqueda.

Se deben detectar los documentos casi iguales.

Detección de contenidos casi iguales

The image shows a side-by-side comparison of two web pages in a browser window. The left page is the official Wikipedia article for Michael Jackson, with the standard Wikipedia logo and navigation menu. The right page is a parody site called "wapedia," which mimics the layout of Wikipedia but has a pink header and slightly different text. Both pages feature a photo of Michael Jackson in a black military-style jacket with a red armband. The browser's address bar and search bar are visible at the bottom of both pages.

Left Page (Wikipedia):

- Header: **Michael Jackson**
- Text: From Wikipedia, the free encyclopedia
- Text: For other persons named Michael Jackson, see [Michael Jackson \(disambiguation\)](#).
- Text: **Michael Joseph Jackson** (August 29, 1958 – June 25, 2009) was an American recording artist, entertainer and businessman. The seventh child of the [Jackson family](#), he made his debut as an entertainer in 1968 as a member of [The](#)
- Image:

Right Page (wapedia):

- Header: **wapedia.**
- Text: **Wiki: Michael Jackson (1/6)**
- Text: For other persons named Michael Jackson, see [Michael Jackson \(disambiguation\)](#).
- Text: **Michael Joseph Jackson** (August 29, 1958 - June 25, 2009) was an American recording artist, entertainer and businessman. The seventh child of the [Jackson family](#), he made his debut as an entertainer in 1968 as a member of [The Jackson 5](#). He then began a solo

¿Cómo detectar contenido casi idénticos?



Detección de contenidos casi iguales

- Cuando hablamos de contenidos casi iguales nos referimos a **similitud sintáctica**.
- La **similitud semántica**, que dos webs digan lo mismo con palabras distintas, es mucho más difícil de detectar.
- Una primera aproximación a la detección de la similitud sintáctica podría ser el uso de una **distancia de edición**.
 - Podemos considerar que dos documentos son casi iguales si la similitud entre ellos es mayor que un cierto umbral θ (por ejemplo, 80%).



Similitud basada en shingles

- Un shingle es un **n-grama** del documento: una secuencia de n palabras que aparecen juntas en el documento.
- Dada una **n** los shingles de un documento es el conjunto de n -gramas de ese documento.
- **Ejemplo:** dado el texto “a rose is a rose is a rose” obtener el conjunto de shingles para $n = 2$, $n = 3$ y $n = 4$.
 - 2-gramas: { “a rose”, “rose is” , “is a” }
 - 3-gramas: { “a rose is”, “rose is a” , “is a rose” }
 - 4-gramas: { “a rose is a”, “rose is a rose” , “is a rose is” }

Similitud basada en shingles

- Los shingles se pueden utilizar para medir la **similitud sintáctica** entre dos documentos.
- Si dos documentos tienen el mismo conjunto de shingles podemos decidir que son iguales, o casi.
- Dados dos documentos d_1 y d_2 , una medida de similitud entre los dos documentos: **coeficiente de Jaccard** de sus conjuntos de shingles para un determinado valor de n .



Coeficiente de Jaccard

El coeficiente de Jaccard es una **medida de solapamiento de conjuntos**.

Dados dos conjuntos no vacíos A y B, se define el coeficiente de Jaccard entre ambos conjuntos $Jaccard(A, B)$ (o simplemente $J(A, B)$) como el **cociente entre** la talla de la **intersección** de ambos conjuntos y la talla de su **unión**.

$$JACCARD(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- No es necesario que A y B tengan la misma talla.
- El coeficiente de Jaccard siempre es un número entre 0 y 1.

¿Cómo calcular el coeficiente de Jaccard?

Coeficiente de Jaccard

Ejercicio: dados los documentos:

d1: "Jack London traveled to Oakland"

d2: "Jack London traveled to the city of Oakland"

d3: "Jack traveled from Oakland to the city of London"

Calcular el coeficiente de Jaccard para $n = 2$ y $n = 3$.

$n = 2$

$S(d1) = \{\text{"Jack London", "London traveled", traveled to", "to Oakland"}\}$

$S(d2) = \{\text{"Jack London", "London traveled", "traveled to", "to the", "the city" "city of", "of Oakland"}\}$

$S(d3) = \{\text{"Jack traveled", "traveled from", "from Oakland", "Oakland to", "to the", "the city", "city of", "of London"}\}$

$J(S(d1), S(d2)) = 3/8 = 0,375$, $J(S(d1), S(d3)) = 0$, $J(S(d2), S(d3)) = 3/12 = 0,25$

Coeficiente de Jaccard

Ejercicio: dados los documentos:

d1: "Jack London traveled to Oakland"

d2: "Jack London traveled to the city of Oakland"

d3: "Jack traveled from Oakland to the city of London"

Calcular el coeficiente de Jaccard para $n = 2$ y $n = 3$.

$n = 3$

$S(d1) = \{\text{"Jack London traveled", "London traveled to", traveled to Oakland"}\}$

$S(d2) = \{\text{"Jack London traveled", "London traveled to", "traveled to the", "to the city", "the city of", "city of Oakland"}\}$

$S(d3) = \{\text{"Jack traveled from", "traveled from Oakland", "from Oakland to", "Oakland to the", "to the city", "the city of", "city of London"}\}$

$J(S(d1), S(d2)) = 2/7 = 0,286$, $J(S(d1), S(d3)) = 0$, $J(S(d2), S(d3)) = 2/11 = 0,182$

¿Cómo calcular el coeficiente de Jaccard de forma eficiente?



Calculo del coeficiente de Jaccard

Vamos a utilizar un **hashing** (por ejemplo de 64bits).

- A cada shingle le asociamos un valor de hash en un espacio de 64 bits.
- $H(d_j)$ es el conjunto de valores de hash derivado del conjunto de shingles $S(d_j)$.
- Sea π una permutación aleatoria de los enteros de 64 bits en los enteros de 64 bits.



Calculo del coeficiente de Jaccard

Denotamos con $\Pi(d_j)$ el conjunto de valores hash permutados de $H(d_j)$; cada $h \in H(d_j)$ tendrá un valor $\pi(h) \in \Pi(d_j)$ asociado.

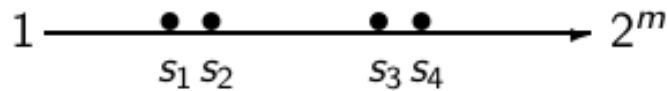
Sea x_j^π el menor entero de $\Pi(d_j)$.

Teorema:

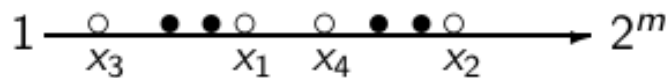
$$J(S(d_1), S(d_2)) = P(x_1^\pi = x_2^\pi)$$

Calculo del coeficiente de Jaccard

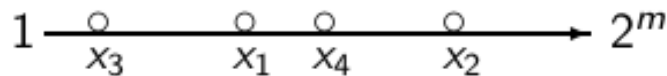
d1: S(d1)



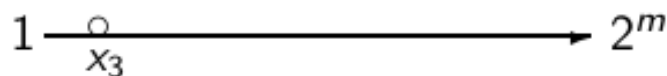
$$x_k = \pi(s_k)$$



x_k



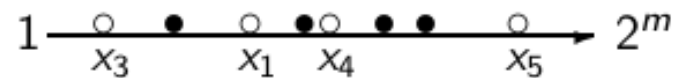
$$\min_{s_k} \pi(s_k)$$



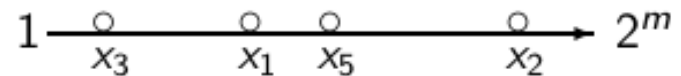
d2: S(d2)



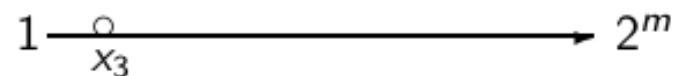
$$x_k = \pi(s_k)$$



x_k



$$\min_{s_k} \pi(s_k)$$



Para la permutación π los documentos $d1$ y $d2$ son casi idénticos



Demostración del teorema

$S(d1)$	$S(d2)$
0	1
1	0
1	1
0	0
1	1
0	1

Sea la siguiente tabla una representación de los shingles de dos documentos:

- cada columna es un documento
- cada fila un shingle.
- cada celda tiene valor 1 o 0 en función de que el shingle pertenezca o no al conjunto de shingles del documento.



Demostración del teorema

S(d1)	S(d2)
0	1
1	0
1	1
0	0
1	1
0	1

Sea:

- C00: el número de filas donde en ambas columnas hay un 0.
- C01: el número de filas donde en la primera columna hay un 0 y en la segunda un 1.
- C10: el número de filas donde en la primera columna hay un 1 y en la segunda un 0.
- C11: el número de filas donde en las dos columnas hay un 1.

Demostración del teorema

S(d1)	S(d2)
0	1
1	0
1	1
0	0
1	1
0	1

¿Cuál es el coeficiente de Jaccard de los dos documentos?:

$$J(S(d1), S(d2)) = \frac{|S(d1) \cap S(d2)|}{|S(d1) \cup S(d2)|} = \frac{C11}{C01 + C10 + C11}$$

¿Cuál es la probabilidad de que en cualquier permutación aleatoria de filas el primer 1 aparezca simultáneamente en ambas columnas?

$$P(x_1^\pi = x_2^\pi) = \frac{C11}{C01 + C10 + C11}$$

$$J(S(d_1), S(d_2)) = P(x_1^\pi = x_2^\pi)$$



Calculo del coeficiente de Jaccard

- Generamos **200 permutaciones aleatorias**, pueden ser 200 funciones de hash distintas.
- Para cada permutación π y cada documento d_i calculamos x_i^π . Llamamos ψ_i al conjunto de 200 valores de x_i^π para el documento d_i .

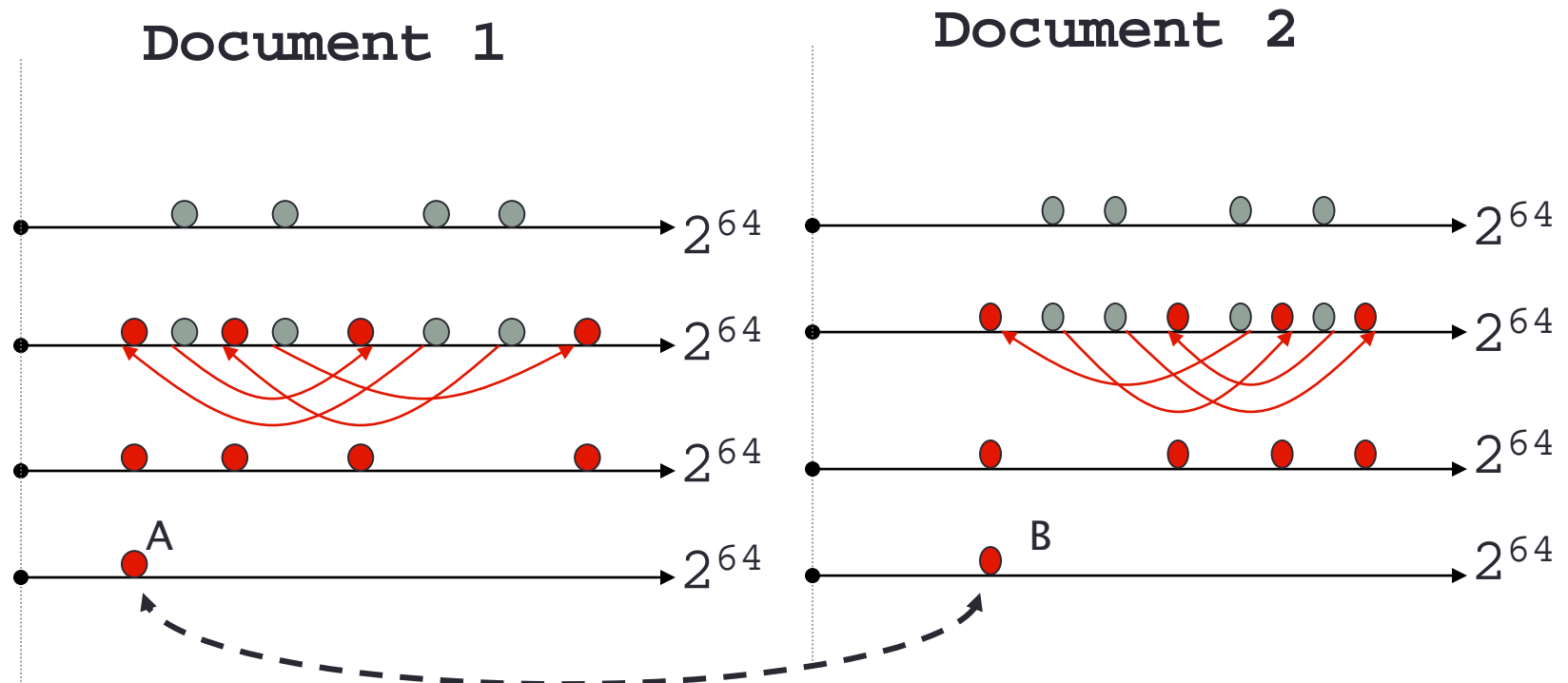
Aproximamos el coeficiente de Jaccard como:

$$J(S(d_i), S(d_j)) \approx \frac{|\psi_i \cap \psi_j|}{200}$$

Si el valor obtenido supera un umbral determinado podemos afirmar que los documentos d_i y d_j son similares.



Calculo del coeficiente de Jaccard



Testear para 200 permutaciones random: $\pi_1, \pi_2, \dots, \pi_{200}$



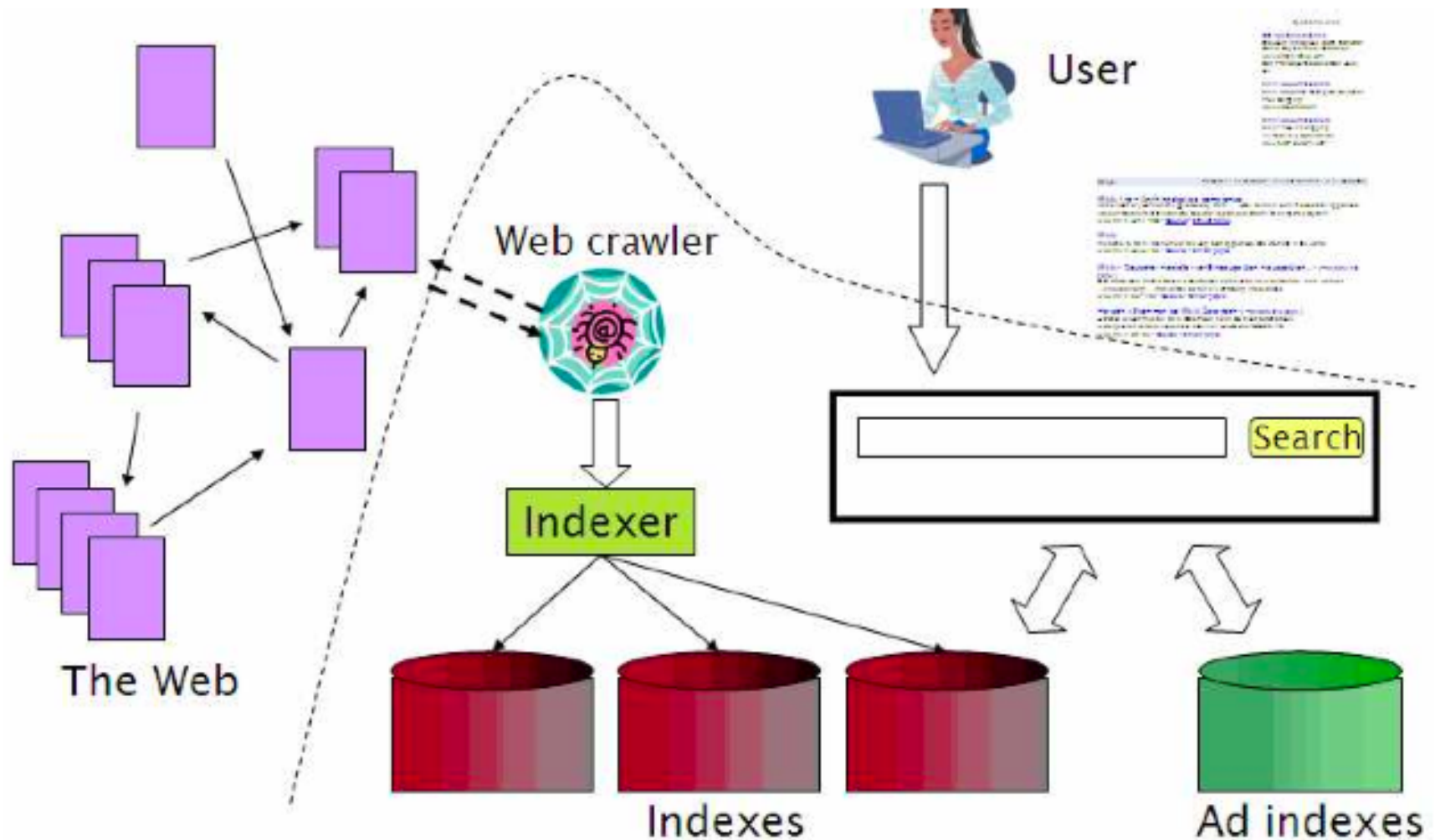
Calculo del coeficiente de Jaccard

Una última optimización heurística para reducir el número de pares de documentos a los que calcular el coeficiente de similitud:

- Ordenar los elementos de ψ_i y calcular su conjunto de shingles (**super-shingles**).
- Sólo compararemos (utilizando el coeficiente de Jaccard) aquellos documentos que tengan algún super-shingle en común.

4. WEB CRAWLER

Esquema de la RI en la web





Web crawler

Los sistemas de recuperación de información deben acceder al contenido de los documentos para indexarlos.

Sistema de recuperación de información **clásico**:

- **Fácil y rápido.**
- Indexar una colección de documentos en un directorio de un disco local.
- Hacer un recorrido recursivo por el sistema de ficheros.



Web crawler

Los sistemas de recuperación de información deben acceder al contenido de los documentos para indexarlos.

Sistema de recuperación de información para la web:

- Más **complicado** y mayor **coste temporal** (**latencia**).
- Indexar una colección de documentos repartidos por múltiples servidores.
- Analizar los enlaces de los documentos para recuperar nuevos documentos.



¿Qué debe hacer un web crawler?

Inicializar la cola con URLs de páginas conocidas

Repetir:

- Coger una URL de cola

- Recuperar y analizar la página

- Extraer las URLs de la página

- Añadir las URLs a la cola

Asunción: la web es un grafo fuertemente conexo.

Con más detalle ...

```
urlqueue := (urls iniciales bien seleccionadas)
while urlqueue is not empty:
    myurl := urlqueue.getlastanddelete()
    mypage := myurl.fetch()
    fetchedurls.add(myurl)
    newurls := mypage.extracturls()
    for myurl in newurls:
        if myurl not in fetchedurls
            and not in urlqueue:
                urlqueue.add(myurl)
    addtoinvertedindex(mypage)
```




¿Qué le falta al web crawler básico?

- **Escalabilidad**: Es necesario distribuir el proceso.
- **Selección**: No se puede indexar toda la web, hay que seleccionar lo que se va a indexar.
- **Control de Duplicados**. La detección de duplicados debe formar parte del proceso de crawling.
- **Detección** de la páginas de **spam** y “**spider traps**”. Se deben detectar este tipo de páginas para no indexarlas y “liberar” el web crawler.



¿Qué le falta al web crawler básico?

- **Cortesía:** No se deben sobrecargar los servidores que alojan las páginas. Las peticiones a un mismo servidor deben espaciarse.
- **Refresco:** El contenido de las páginas se va actualizando. Necesitamos repetir el proceso de crawling periódicamente.
 - Sólo podemos hacer refrescos frecuentes de algunas páginas.
 - Problema de selección y priorización.

Ejemplo: Para obtener 25,000,000,000 páginas en un mes se deben procesar casi **10,000 páginas por segundo**.



robots.txt

- Definido en 1994.
- Un fichero de texto que se pone en la **raíz del sitio web**.
- Da indicaciones al crawler (también llamado robot) sobre qué parte de la propia web **no debe ser indexada**.
- Es decisión del crawler respetar las directrices de robots.txt.
- Los web crawler usados por los buscadores web respetan las directrices del robots.txt.



Un ejemplo real: www.upv.es/robots.txt

```
User-agent: *
Disallow: /upvrenew
Disallow: /webpreview
Disallow: /wniujom
Disallow: /pls/
Disallow: /pls/soalu
Disallow: /pls/oalu
Disallow: /pls/sobib
Disallow: /pls/soarc
Disallow: /obibproxy
Disallow: /oaluproxy
Disallow: /policonsulta/c/*
Disallow: /policonsulta/v/*
Disallow: /policonsulta/i/*
Disallow: /contenidos/BIBTEST*
Disallow: /bin2/tipoacc/*
Allow: /ical/*
Allow: /pls/ical/sic_ical_crypt.getCal*
Allow: /pls/obib/sic_bibpublicador.listas*
Allow: /pls/oreg/rtv_web.*
Allow: /pls/oalu/sic_per.info_persona*
Allow: /pls/oalu/sic_person.info*
Allow: /ficha-personal/*
```



content="noindex"

- content="noindex" es un atributo que se añade a la etiqueta **<meta>** en la cabecera de un documento HTML.
- Indica a los web crawlers que no se indexe la página.

```
<html>
<head>
  <meta name="robots" content="noindex">
  <title>Esta página no se indexa</title>
</head>
```



rel="nofollow"

- Definido en 2005 por Google y Blogger.
- rel="nofollow" es un atributo que se añade a la etiqueta de enlace <a> de HTML.
- El crawler sí sigue estos enlaces pero ...

Los enlaces con este atributo no son tenidos en cuenta por los buscadores al calcular los resultados de una consulta (en el cálculo del pagerank por ejemplo).

- Evitar el uso de spammers robots en los comentarios.
- Los spammers robots se siguen utilizando.

```
<a href="http://spam.com" rel="nofollow">texto</a>
```



Contenido ya visto

Para cada página descargada:

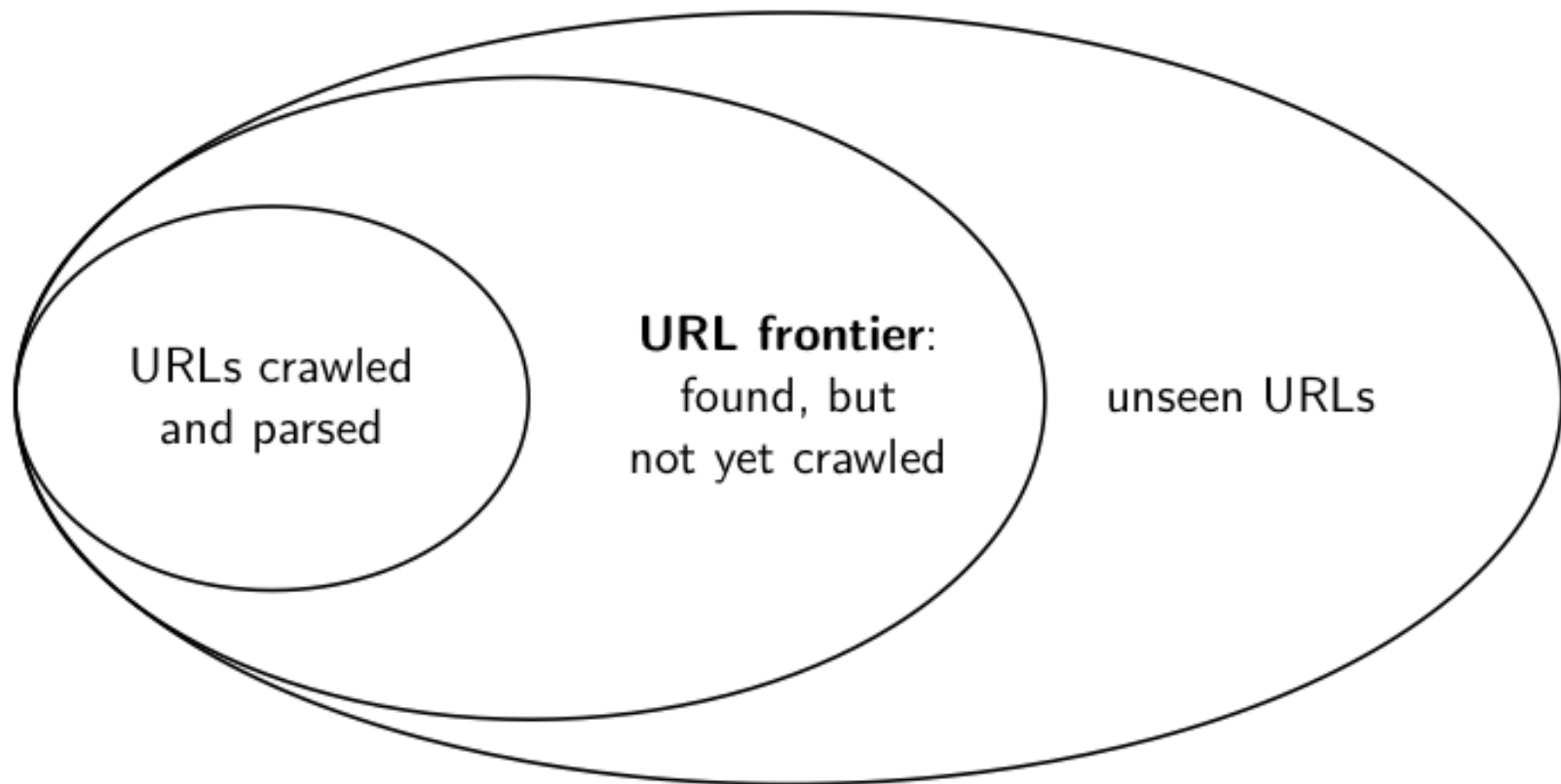
- Comprobar si el contenido es **idéntico** al de algún documento ya indexado (**fingerprint**).
- Comprobar si el contenido es **casi igual** al de algún documento ya indexado (**shingles** y coeficiente de **Jaccard**).
- Las páginas con contenido ya indexado deben descartarse.

Normalización de URL

- Algunas URLs extraídas de un documento son las direcciones URL relativas.
- El enlace a **info.html** dentro de www.pagina.com es equivalente a un enlace a www.pagina.com/info.html.
- Todas las URLs relativas deben convertirse en absolutas durante el análisis.



URL frontier



5. LA WEB COMO UN GRAFO DIRIGIDO



La web como un grafo dirigido

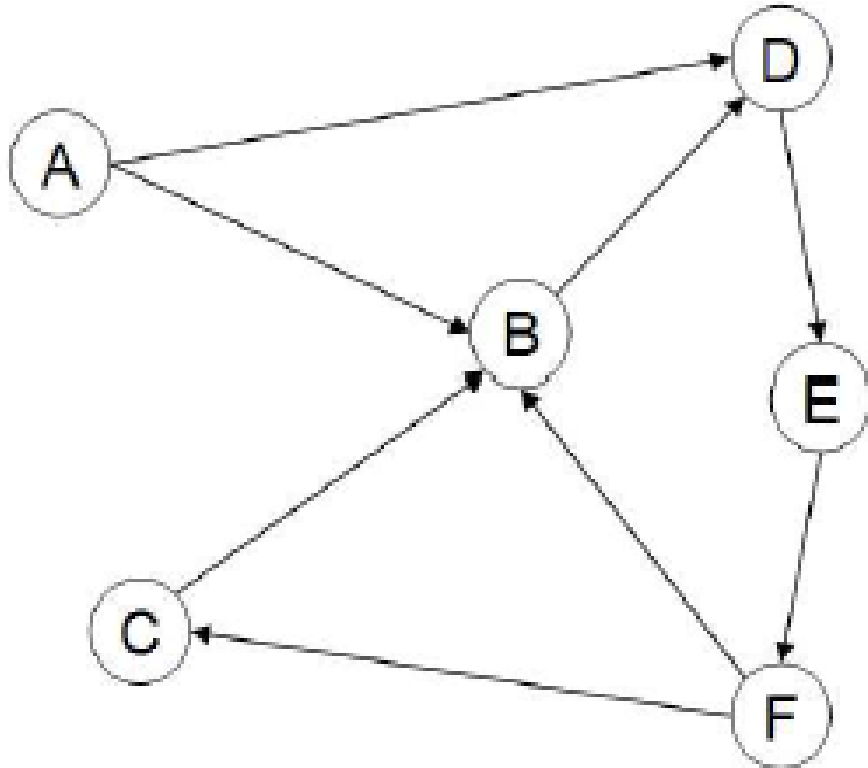
Las páginas web utilizan la etiqueta `<a>` (anchor) de HTML para enlazar a otras páginas mediante hipervínculos.

```
<a href="http://www.dirección_del_enlace.com">
```

```
texto del hipervínculo</a>
```

Podemos ver las páginas web estáticas junto a los hipervínculos entre ellas como un grafo dirigido en el que cada página web es un nodo y cada hipervínculo un arco dirigido.

La web como un grafo dirigido



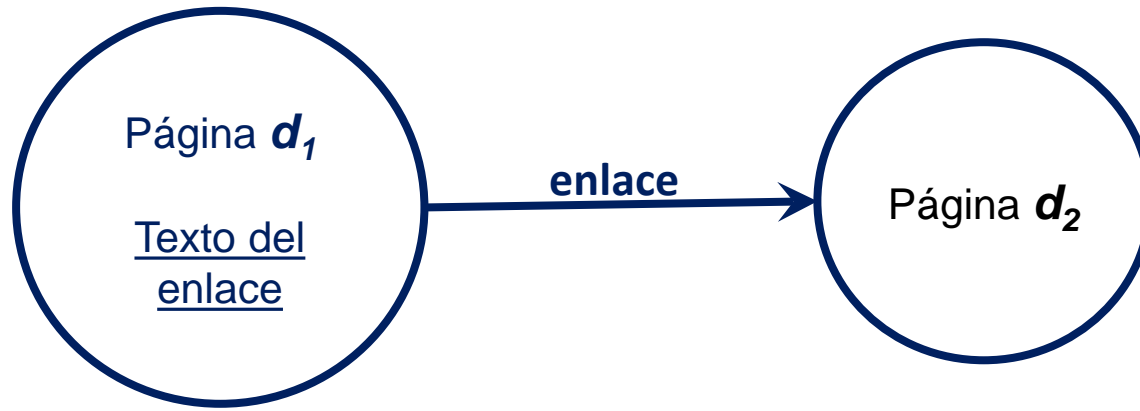
- Los nodos (**A**, **B**, **C**, **D**, **E**, **F**) representan páginas web.
- Los arcos representan hipervínculos entre las páginas.
- Ejemplo, **B** tiene:
 - grado de salida 1
 - grado de entrada 3

¿Este grafo es fuertemente conexo?

¿Es la web un grafo fuertemente conexo?

6. USO DEL TEXTO DEL ENLACE

La web es un grafo dirigido

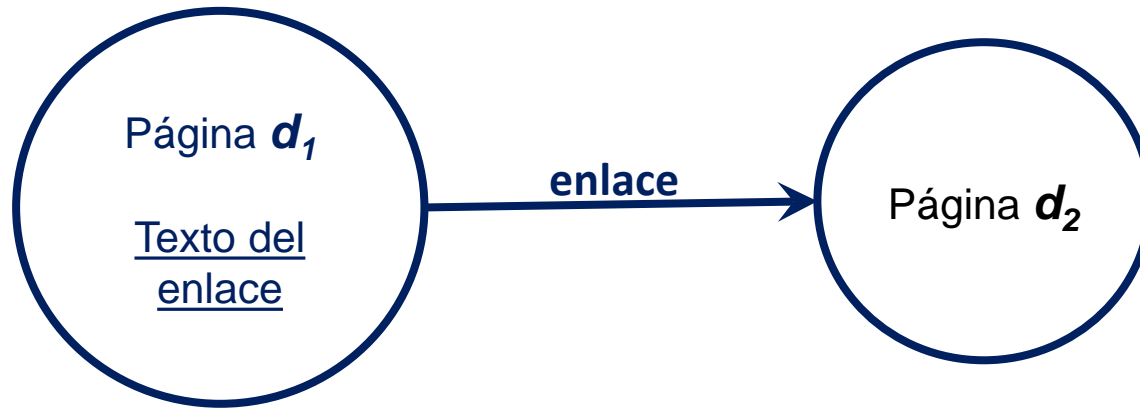


Primera hipótesis: **Un enlace es una señal de calidad.**

- El enlace de d_1 hacia d_2 indica que el autor de d_1 considera que d_2 es relevante y de alta calidad.

¿Es esto cierto habitualmente?

La web es un grafo dirigido

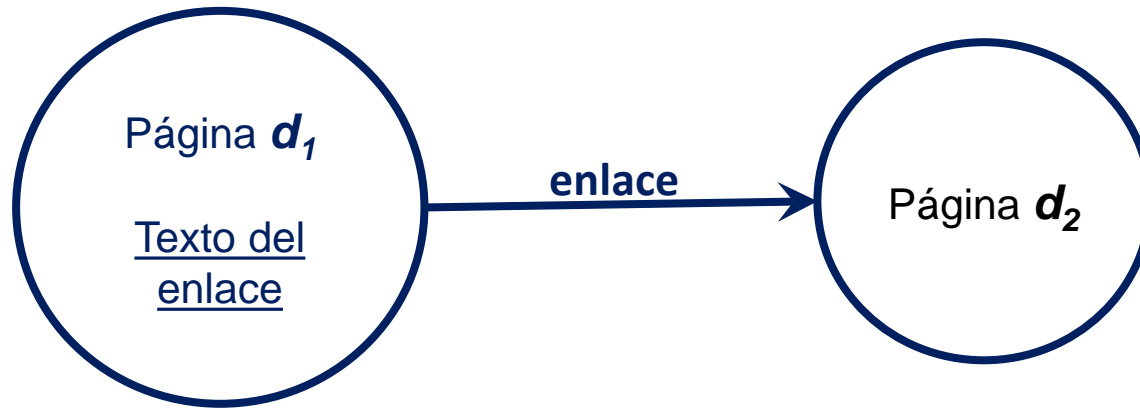


Segunda hipótesis: El texto del enlace de d_1 a d_2 es un buen descriptor del contenido de d_2 .

- Aquí ampliamos el texto del enlace a texto “alrededor” del enlace.

¿Es esto cierto habitualmente?

La web es un grafo dirigido



Ejemplo:

¿Quieres cambiar de ordenador?. Los mejores ordenadores al mejor precio aquí

- Texto del enlace: Los mejores ordenadores al mejor precio aquí



Uso del texto del enlace

- Un enlace representa una señal de calidad (**primera hipótesis**).
- El texto de los enlaces a una página web suele contener términos que describen adecuadamente la página. (**segunda hipótesis**).
- Muchas veces los términos de los enlaces no aparecen en la página.
- Pero muchas veces la describen mejor que el contenido de la propia página.



Uso del texto del enlace

Ejemplo:

- Durante mucho tiempo la página oficial de IBM (www.ibm.com) no contenía la palabra **IBM**.
- La página oficial de IBM no contiene la palabra **computer**.
- Muchos de los enlaces a www.ibm.com contienen las palabras IBM o computer.



Uso del texto del enlace

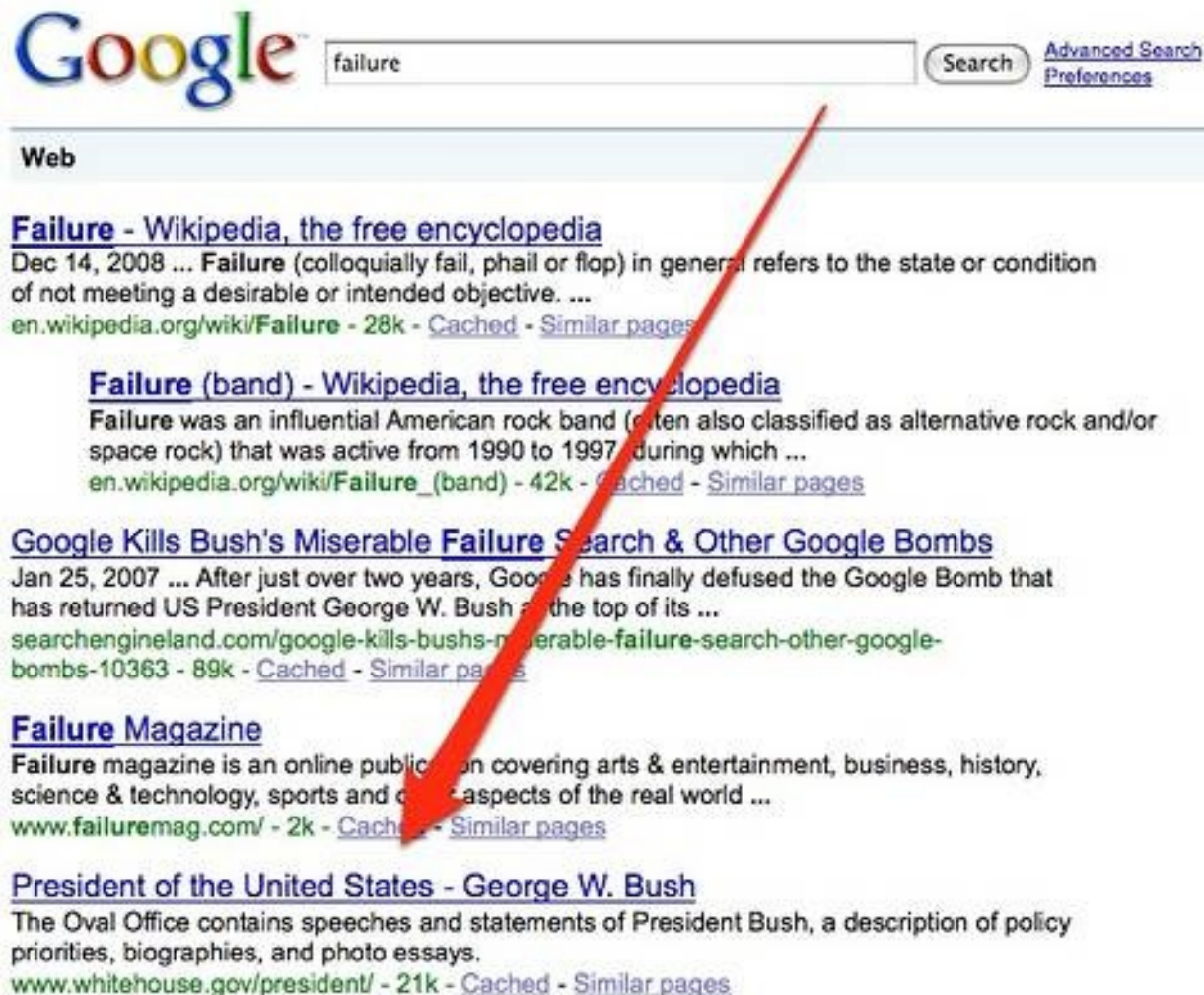
- Es una buena idea utilizar los términos del texto de los enlaces a una página para indexar esa página.
- Los términos de los enlaces son, muchas veces, **tenidos más en cuenta** por los buscadores que los términos de la página.
- Podemos encontrar una página buscando por términos que no aparecen en la página. Términos “**no controlables**” por el creador de la página.



Google bombs

- Intento de **manipular los resultados** de un buscador aprovechando que el buscador tiene muy en cuenta el texto de los enlaces a páginas web.
- **Muchos enlaces** a una misma página web desde múltiples sitios con un **mismo texto**.
- Al buscar por ese texto se consigue que se obtenga como resultado la página.
- En 2007 Google hizo cambios para minimizar el efecto.

Google bombs



Google

failure

Search

[Advanced Search](#)
[Preferences](#)

Web

Failure - Wikipedia, the free encyclopedia
Dec 14, 2008 ... **Failure** (colloquially fail, phail or flop) in general refers to the state or condition of not meeting a desirable or intended objective. ...
en.wikipedia.org/wiki/Failure - 28k - [Cached](#) - [Similar pages](#)

Failure (band) - Wikipedia, the free encyclopedia
Failure was an influential American rock band (often also classified as alternative rock and/or space rock) that was active from 1990 to 1997, during which ...
[en.wikipedia.org/wiki/Failure_\(band\)](http://en.wikipedia.org/wiki/Failure_(band)) - 42k - [Cached](#) - [Similar pages](#)

Google Kills Bush's Miserable Failure Search & Other Google Bombs
Jan 25, 2007 ... After just over two years, Google has finally defused the Google Bomb that has returned US President George W. Bush at the top of its ...
searchengineland.com/google-kills-bushs-miserable-failure-search-other-google-bombs-10363 - 89k - [Cached](#) - [Similar pages](#)

Failure Magazine
Failure magazine is an online publication covering arts & entertainment, business, history, science & technology, sports and other aspects of the real world ...
www.failuremag.com/ - 2k - [Cached](#) - [Similar pages](#)

President of the United States - George W. Bush
The Oval Office contains speeches and statements of President Bush, a description of policy priorities, biographies, and photo essays.
www.whitehouse.gov/president/ - 21k - [Cached](#) - [Similar pages](#)

Google bombs



Web

[5 results stored on your computer](#) - [Hide](#) - [About](#)



[About Blether](#) - a gentle term for a **liar** or fibber, someone who is telling
[BBC NEWS | Programmes | F..](#) - of being a **liar**, a dictator, of robbing

[Tony Blair - Biography](#)

Read the full biography of Prime Minister Tony Blair.

www.pm.gov.uk/output/Page4.asp - 12k - [Cached](#) - [Similar pages](#) - [Remove result](#)

Google bombs



La Web [Imágenes](#) [Grupos](#) [Noticias](#) [Más »](#)

ladrones

Buscar

[Búsqueda avanzada](#)
[Preferencias](#)

Búsqueda: ☒ la Web ☐ páginas en español ☐ páginas de España

La Web

Resultados **1 - 100** de aproximadamente **5.050.000** de **ladrones**. (0,29 segundos)

[Sociedad General de Autores y Editores](#)

SOCIEDAD GENERAL DE AUTORES Y EDITORES. SGAE Responde, slogan. ¿Qué somos? Dónde Estamos · Grupo SGAE. Idioma. Castellano, Català, Chinese, English, Euskera ...

www.sgae.es/?ladrones - 17k - [En caché](#) - [Páginas similares](#)

[Sociedad General de Autores y Editores](#)

No se han encontrado registros que contengan la palabra "**ladrones**". (c) Copyright Sociedad General de Autores y Editores (SGAE) (Fernando VI, 4 28004 Madrid ...

www.sgae.es/search/search-es.jsp?texto=%3Ca%20href=%22%22%3Eladrones%3C/a%3E
- 7k - [En caché](#) - [Páginas similares](#)

[ladrones](#)

www.ladrones.org/ - 3k - [En caché](#) - [Páginas similares](#)

[Acceder](#)

Google bombs



Web

Google won't search for **Chuck Norris** because it knows you don't find **Chuck Norris**, he finds you.

No standard web pages containing all your search terms were found.

Your search - **Chuck Norris** - did not match any documents.

Suggestions:

- Run, before he finds you
- Try a different person

<http://www.nochucknorris.com/>



7. PAGERANK



Antecedentes del PageRank

Análisis de referencias en la bibliografía científica.

Una referencia dentro de un artículo científico a otro artículo puede verse como un enlace del primer artículo al segundo.

- **1ª Propuesta:** Se puede medir la similitud entre dos artículos por el solapamiento de los artículos que los citan (cocitation similarity).

¿Tiene utilidad para los artículos científicos?

¿Tiene utilidad para los buscadores web?



Antecedentes del PageRank

Análisis de referencias en la bibliografía científica.

- **2ª Propuesta:** Las referencias a un artículo pueden considerarse como una medida del **impacto** de ese artículo.

¿Tiene utilidad para los artículos científicos?

¿Tiene utilidad para los buscadores web?

¿Todas las referencias deben valer lo mismo?



Antecedentes del PageRank

Análisis de referencias en la bibliografía científica.

- **3ª Propuesta:** Un artículo es de gran impacto si es referenciado por artículos de gran impacto.
 - En la web: una página es de gran relevancia si es indexada por páginas de gran relevancia.
 - El «peso» de cada enlace depende de la relevancia de la página que enlaza.

Para determinar la relevancia de una página se tiene en cuenta la relevancia de las páginas que la enlazan.

Un paseo aleatorio

Un internauta haciendo un paseo aleatorio por la web.

1. Comienza en una página cualquiera al azar.
2. En cada paso elige una **nueva página aleatoriamente de manera equiprobable** entre todos los enlaces salientes de la página donde está.
3. Repetir el paso 2 un número «**infinito**» de veces.



Un paseo aleatorio

Un internauta haciendo un paseo aleatorio por la web.

¿En qué pagina se encontrará el internauta?

¿Con qué probabilidad?

Cada página tendrá una probabilidad de que el internauta se encuentre en ella.

Las páginas «más importantes» tendrán una mayor probabilidad.

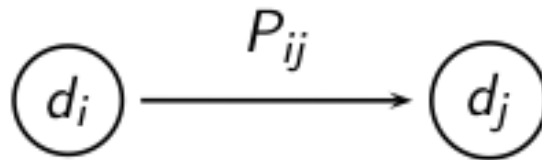
Intuitivamente: la probabilidad de que el internauta esté en una página es el PageRank de esa página.



El paseo aleatorio: una cadena de Markov

Una cadena de Markov viene definida por:

- Un conjunto de N estados.
- Una matriz $N \times N$ de probabilidad de transición, P .
- $\forall 1 \leq i, j \leq N, P_{ij}$ es la probabilidad de pasar al estado j estando en el estado i .
- $\forall 1 \leq i \leq N$, se debe cumplir que $\sum_{j=1}^N P_{ij} = 1$.





El paseo aleatorio: una cadena de Markov

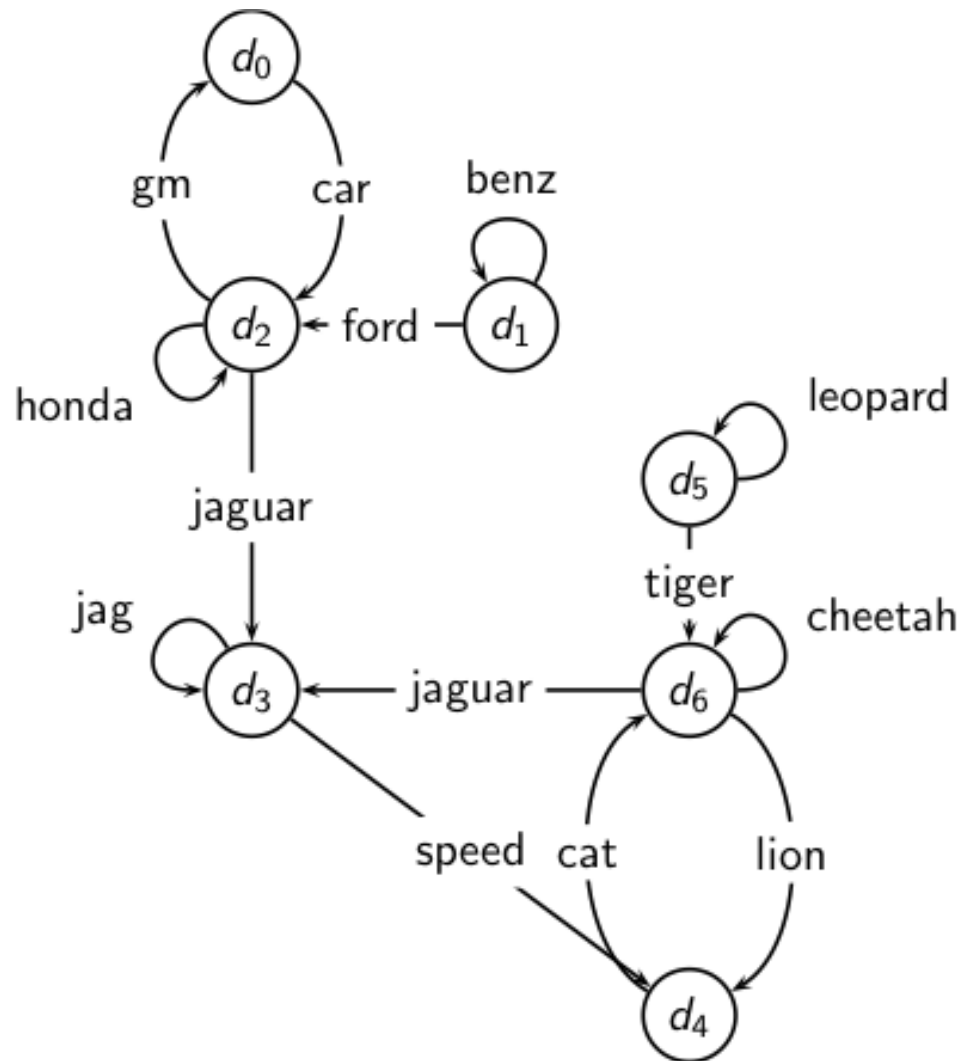
Un internauta haciendo un paseo aleatorio por la web visto como una cadena de Markov:

- Cada página web es un estado de la cadena.
- Sea $out(i)$ el conjunto de todas las páginas alcanzables desde la página i .

$$\bullet \quad \forall 1 \leq i \leq N, \quad \forall j \in out(i), \quad P_{ij} = \frac{1}{|out(i)|}$$

Ejemplo de paseo aleatorio

Grafo:





Ejemplo de paseo aleatorio

Matriz de enlaces:

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



Ejemplo de paseo aleatorio

Matriz de probabilidades de transición, **P**

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



Tasa de visitas a largo plazo

La tasa de visitas a largo plazo de una página d es la probabilidad de que un internauta realizando un paseo aleatorio esté en la página d en un momento determinado.

El **PageRank** de una página es su **tasa de visitas a largo plazo**.

Teorema 1: Para una cadena de Markov **ergódica** hay una única tasa de visitas a largo plazo.

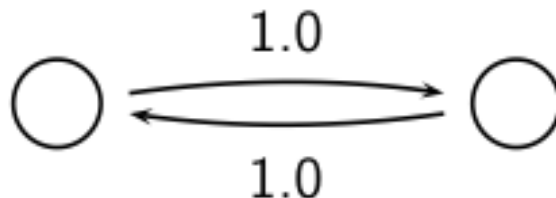
Consecuencia: para poder calcular el PageRank el grafo de la web debe ser una cadena de Markov ergódica.

Cadena de Markov ergódica

Una cadena de Markov es ergódica si es irreducible y aperiódica.

- **Irreducible**: Desde cualquier estado se puede alcanzar cualquier otro, no necesariamente en un solo salto.
- **Aperiódica**: No existe ningún estado tal que todos los caminos desde él a sí mismo tengan una longitud múltiplo de un periodo $k > 1$.

Ejemplo de cadena periódica de periodo $k = 2$





Cadena de Markov ergódica

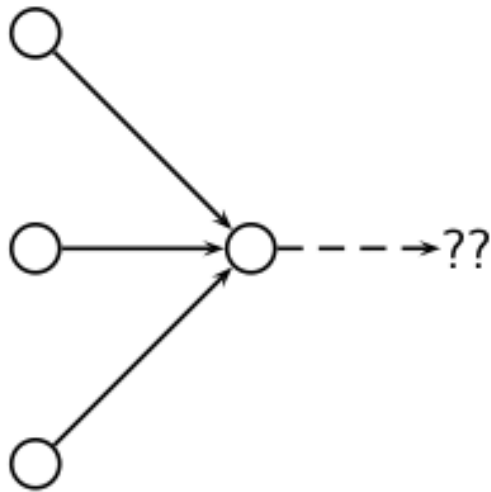
Una cadena de Markov es ergódica si es irreductible y aperiódica.

- **Irreductible**: Desde cualquier estado se puede alcanzar cualquier otro, no necesariamente en un solo salto.
- **Aperiódica**: No existe ningún estado tal que todos los caminos desde él a sí mismo tengan una longitud múltiplo de un periodo $k > 1$.

¿Es la web ergódica?



Callejones sin salida



- La web está llena de callejones sin salida.
- El internauta aleatorio se quedará atrapado.
- Con callejones sin salida la web no es ergódica.
- Tasa de visitas a largo plazo no está bien definida.



Solución a los callejones sin salida: teletransporte

- Si estamos en un callejón sin salida:
 - Saltar equiprobablemente a cualquier página web con probabilidad $\frac{1}{N}$
- Si no estamos en un callejón sin salida:
 - Con probabilidad α : Saltar a cualquier página web. Salto a cualquier página con probabilidad $\frac{\alpha}{N}$
 - Con probabilidad $1 - \alpha$: Saltar utilizando alguno de los enlaces de la página de forma equiprobable.



Solución a los callejones sin salida: teletransporte

- α es un parámetro ajustable. Típicamente $\alpha = 0,1$.
- La probabilidad de salida en los callejones no depende de α . Sólo depende del número de páginas.

Ejemplo: 1000 páginas web ($N = 1000$) y $\alpha = 0,1$.

- En los callejones sin salida elegimos cualquier página web con probabilidad $0,001$.
- En una página con 4 enlaces, podemos elegir:
 - 996 páginas con probabilidad $0,0001$ ($0,1 / 1000$).
 - 4 páginas con probabilidad $0,2251$ ($0,0001 + 0,9 / 4$).



Cadena de Markov ergódica

- **Teorema 1:** Para una cadena de Markov ergódica hay una única tasa de visitas a largo plazo.
- En el límite (un número grande de saltos) cada estado es visitado en proporción a esta tasa.
- No importa el estado de inicio del paseo.
- Esta tasa es la **distribución de probabilidad en estado estacionario**.

Cadena de Markov ergódica

- La cadena de Markov definida por la web añadiéndole el teletransporte es una cadena de Markov ergódica.
 - Se puede ir a cualquier página desde cualquier página (**Irreductibilidad**).
 - No hay estados periódicos (**aperiodicidad**).
- La web con teletransporte tiene una distribución de probabilidad en estado estacionario.
- Esta probabilidad para cada página es su PageRank.



Vector de probabilidades para el paseo aleatorio

- El vector de probabilidades \vec{x} (x_1, x_2, \dots, x_N) indica la probabilidad de estar en cada estado en el paseo aleatorio.
- El internauta aleatorio está en el estado i con probabilidad x_i .
- $\sum_{\forall i} x_i = 1$.

Ejemplo:

Al iniciar el paseo:

(0 0 0 ... 1 ... 0 0 0)

Posteriormente:

(0.05 0.01 0.02 ... 0.2 ... 0.01 0.05 0.03)



Evolución del vector de probabilidades

- En un instante de tiempo t , el vector de probabilidades es $\vec{x} = (x_1, x_2, \dots, x_N)$.
- Dada la matriz de probabilidades de transición P , la probabilidad de pasar a la página j estando en la página i es P_{ij} .
- ¿Cuál será el valor del vector de probabilidades en el instante $t+1$?
- La probabilidad de estar en el estado j en el instante $t+1$ será: $\sum_{\forall i} x_i P_{ij}$.
- El vector de probabilidades en $t+1$ se puede expresar como: $\vec{x}P$.

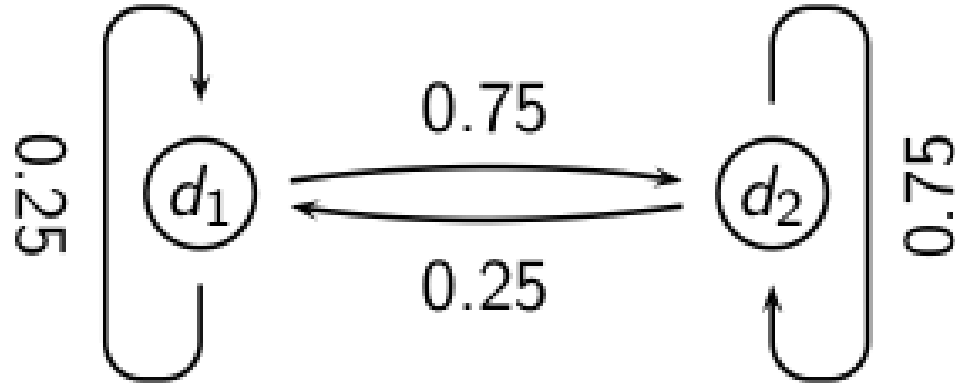
PageRank representado como vector

- El vector de probabilidades $\vec{\pi}$ ($\pi_1, \pi_2, \dots, \pi_N$) representa la distribución de probabilidad en estado estacionario. Es la evolución del vector \vec{x} cuando el tiempo crece indefinidamente.
- π_i es la tasa de visitas a largo plazo del estado i .
- π_i es el PageRank para la página i .
- El PageRank se puede representar como un vector (muy grande) con un valor por cada página web.



Ejemplo de PageRank

¿Cuál es el PageRank para este ejemplo?



Ejemplo de PageRank

	x_1 $P_t(d_1)$	x_2 $P_t(d_2)$		
			$P_{11} = 0.25$ $P_{21} = 0.25$	$P_{12} = 0.75$ $P_{22} = 0.75$
t_0	1	0	0.25	0.75
t_1	0.25	0.75	0.25	0.75
t_2	0.25	0.75	(converge)	

$$P_t(d_1) = P_{t-1}(d_1) * P_{11} + P_{t-1}(d_2) * P_{21}$$

$$P_t(d_2) = P_{t-1}(d_1) * P_{12} + P_{t-1}(d_2) * P_{22}$$

PageRank: $\vec{\pi} = (\pi_1, \pi_2) = (0.25, 0.75)$

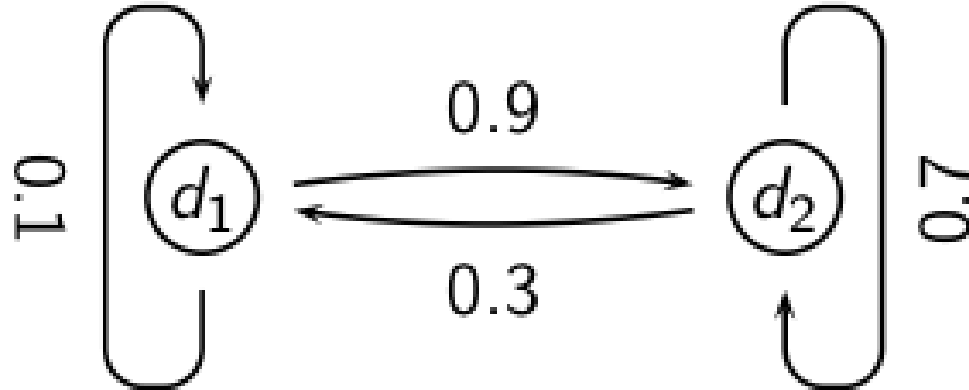


Cálculo del PageRank ($\vec{\pi}$)

- Comenzar con cualquier distribución de probabilidad inicial \vec{x} .
- Después del primer paso estamos en $\vec{x}P$.
- Después del segundo paso estamos en $\vec{x}P^2$.
- Después de k pasos estamos en $\vec{x}P^k$.
- **Método de las potencias:** multiplicar \vec{x} por potencias crecientes de P hasta que converja ($\vec{x}P^t = \vec{x}P^{t+1}$).
- Con el número suficiente de iteraciones, independientemente del valor inicial de \vec{x} se obtiene el PageRank $\vec{\pi}$.

Ejemplo del método de las potencias

¿Cuál es el PageRank para este ejemplo?



Ejemplo del método de las potencias

	x_1 $P_t(d_1)$	x_2 $P_t(d_2)$			
			$P_{11} = 0.1$ $P_{21} = 0.3$	$P_{12} = 0.9$ $P_{22} = 0.7$	
t_0	0	1	0.3	0.7	$= xP$ \rightarrow
t_1	0.3	0.7	0.24	0.76	$= xP^2$ \rightarrow
t_2	0.24	0.76	0.252	0.748	$= xP^3$ \rightarrow
t_3	0.252	0.748	0.2496	0.7504	$= xP^4$ \rightarrow
			\dots		\dots
t_∞	0.25	0.75	0.25	0.75	$= xP^\infty$ \rightarrow

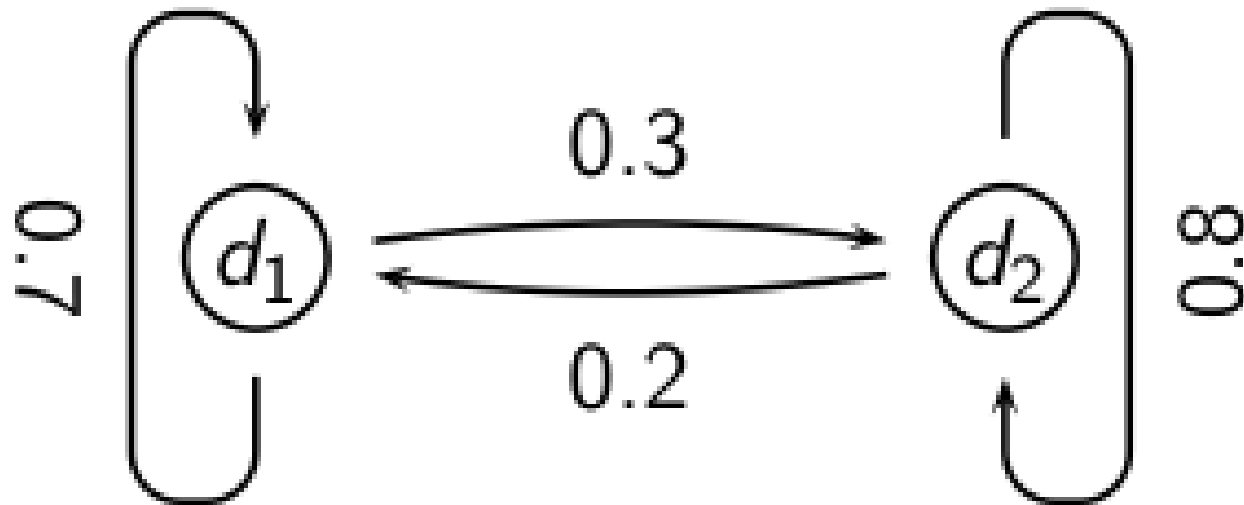
$$P_t(d_1) = P_{t-1}(d_1) * 0.1 + P_{t-1}(d_2) * 0.3$$

$$P_t(d_2) = P_{t-1}(d_1) * 0.9 + P_{t-1}(d_2) * 0.7$$

PageRank: $\vec{\pi} = (\pi_1, \pi_2) = (0.25, 0.75)$



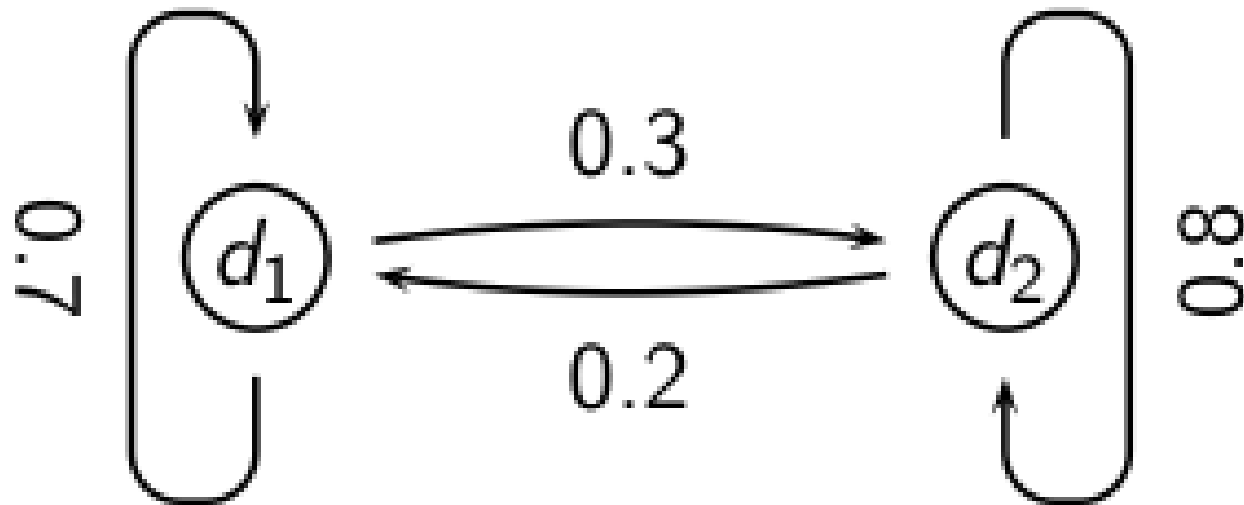
Ejercicio: Calcular el PageRank utilizando el método de las potencias



Nota: máximo 5 iteraciones



Solución



El PageRank del documento 1 es 0.4

El PageRank del documento 2 es 0.6



Solución

	x_1 $P_t(d_1)$	x_2 $P_t(d_2)$		
			$P_{11} = 0.7$ $P_{21} = 0.2$	$P_{12} = 0.3$ $P_{22} = 0.8$
t_0	0	1	0.2	0.8
t_1	0.2	0.8	0.3	0.7
t_2	0.3	0.7	0.35	0.65
t_3	0.35	0.65	0.375	0.625
			...	
t_∞	0.4	0.6	0.4	0.6



Utilización del PageRank en la recuperación de información en la web

Para calcular el PageRank:

1. A partir de la matriz de enlaces construir la matriz de probabilidades de transición P_0 .
2. Aplicar el teletransporte para obtener una nueva matriz de probabilidades de transición P .
3. Utilizando un vector de probabilidades inicial \vec{x} y aplicando el método de las potencias obtener el vector $\vec{\pi}$.

$\vec{\pi}_i$ es el PageRank de la página representada por el estado i .

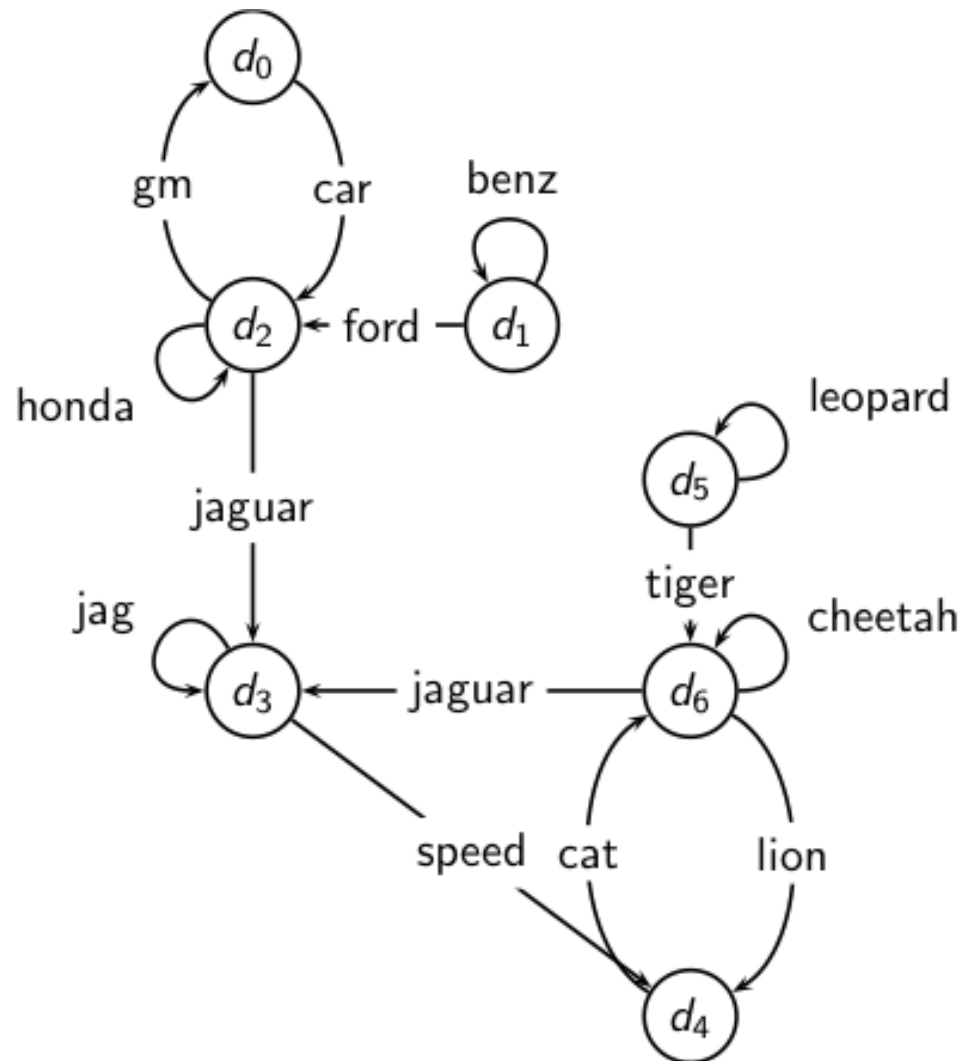
Utilización del PageRank en la recuperación de información en la web

Para responder a una consulta del usuario:

1. Recuperar todos los documentos (las páginas web) que satisfacen la consulta planteada por el usuario.
2. Utilizar el PageRank de cada página para ordenar la lista de resultados.
3. Presentarle al usuario la lista ordenada de páginas web.

Del grafo de páginas web al PageRank

El grafo:



Del grafo de páginas web al PageRank

Matriz de enlaces:

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

Del grafo de páginas web al PageRank

La matriz de probabilidades de transición inicial

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



Del grafo de páginas web al PageRank

La matriz de probabilidades de transición con teletransporte

($\alpha = 0.14$)

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

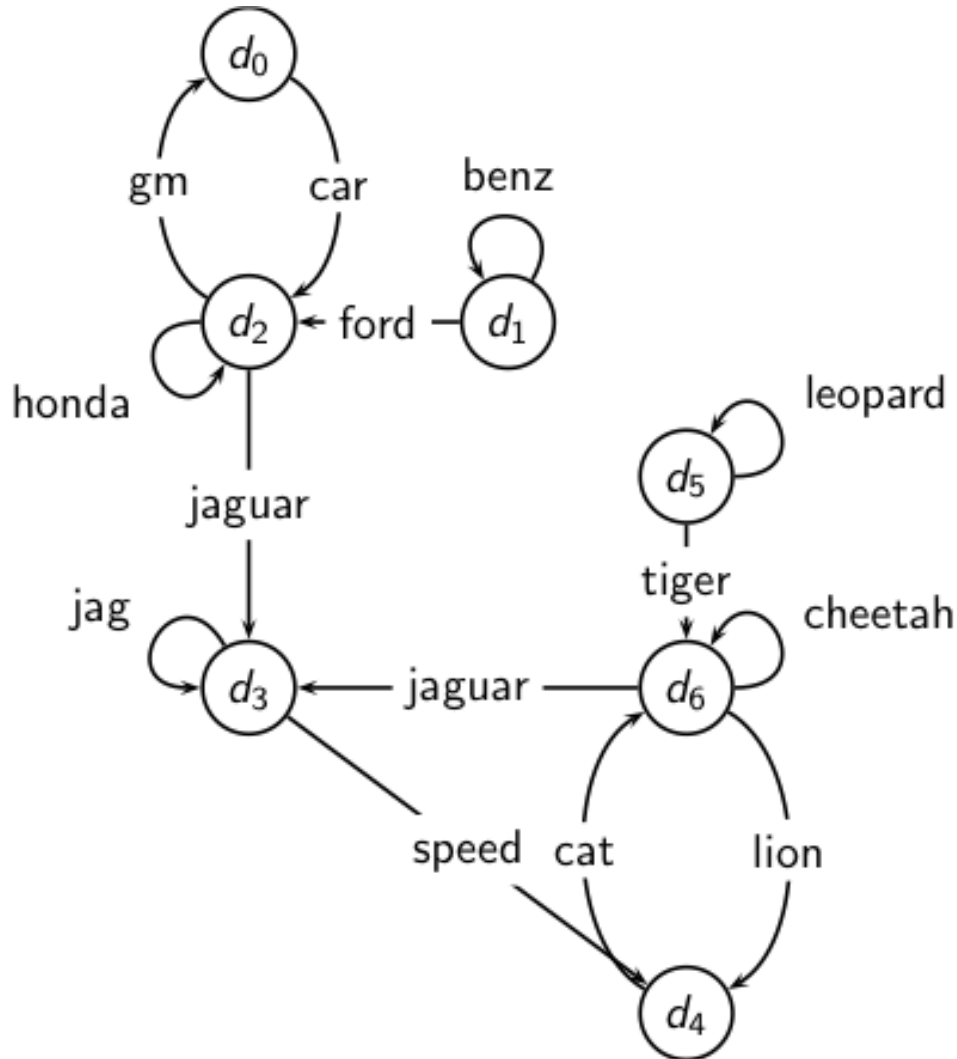


Del grafo de páginas web al PageRank

Aplicar el método de las potencias

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

Del grafo de páginas web al PageRank



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



Consideraciones sobre el PageRank

- En la actualidad **no** se utiliza **sólo el PageRank** para ordenar el resultado de una consulta en un buscador.
- Se utiliza una **combinación ponderada de muchos factores**: distancias entre página y la consulta, distancias entre el texto de los enlaces y la consulta, el PageRank de las páginas, situación geográfica, ...
- El PageRank, una versión más evolucionada, sigue siendo importante.
- Es necesario detectar el **link spam** para que el PageRank sea más fiable.



8. HITS: HUBS Y AUTHORITIES



HITS: Hyperlink Induced Topic Search

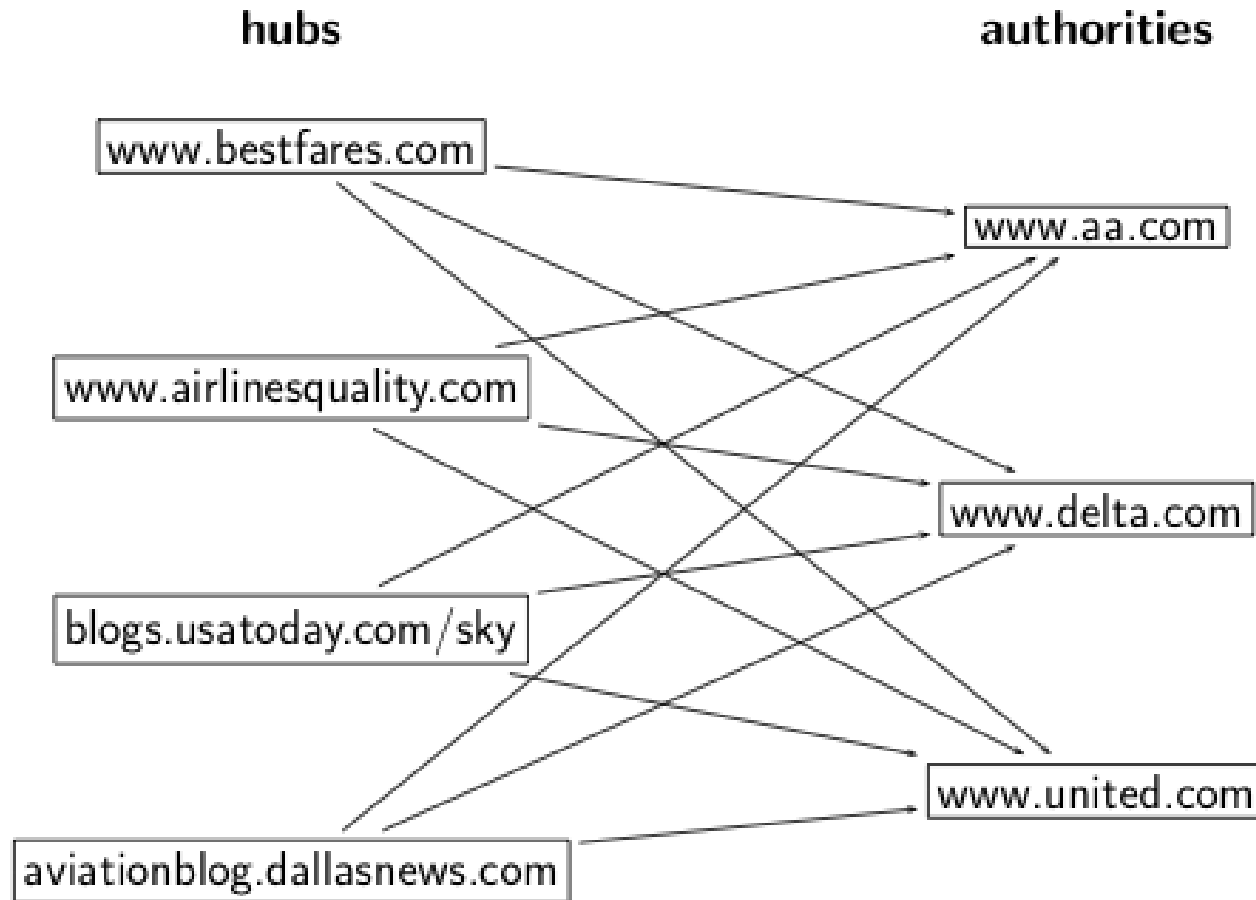
- **Premisa** en la que se basa HITS: hay dos tipos diferentes de relevancia en la web.
 - **Hub**. Es una página que contiene una buena lista de enlaces a páginas que contienen información útil.
 - **Authority**. Es una página que contiene respuesta directa a necesidades de información.
- PageRank no hace distinción entre estos dos tipos de relevancia.
- Muchas de las páginas que enlazan páginas tipo authority son hubs.



Hubs y Authorities

- Una buena página hub para un tema enlaza a muchas páginas authority para ese tema.
- Una buena página authority para un tema es enlazada por muchas páginas hub de ese tema.
- Es una definición circular que se puede transformar en un proceso iterativo.

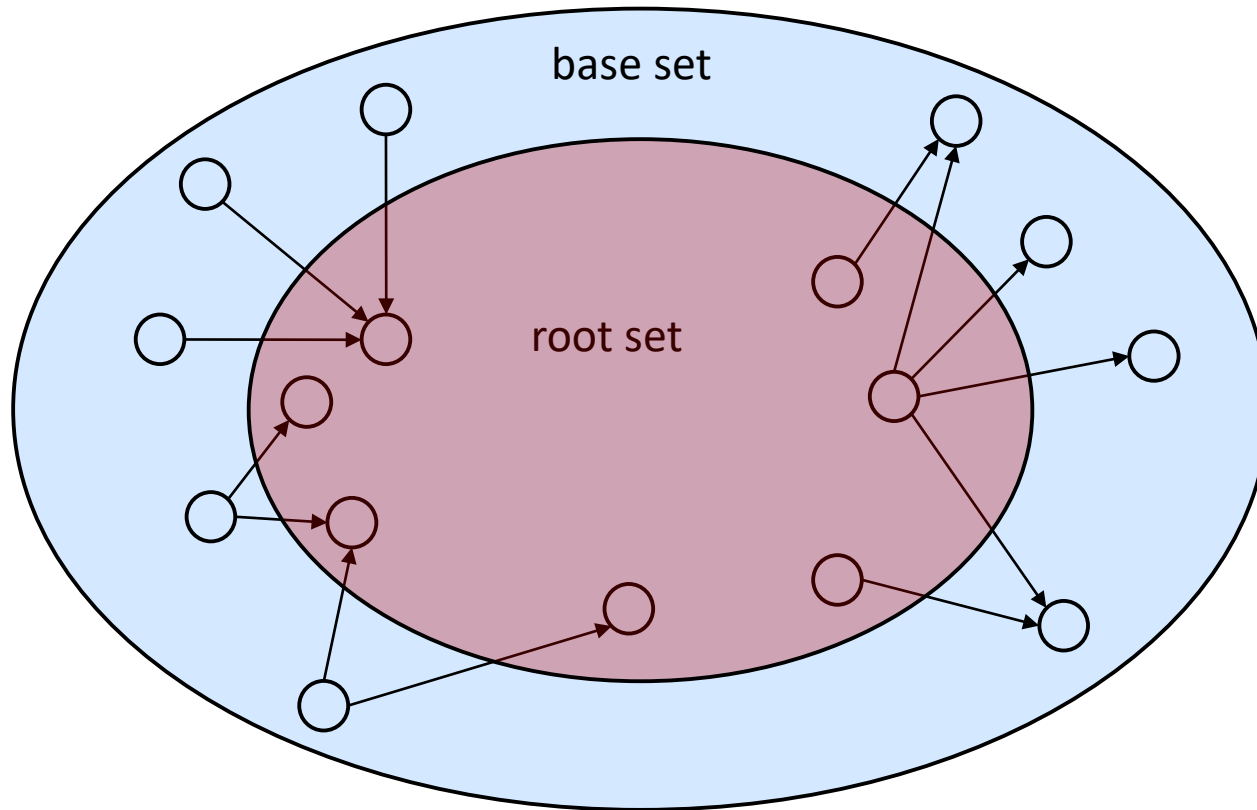
Ejemplo de Hubs y Authorities



Cálculo de hubs y authorities

- Se parte de una consulta inicial.
- Al conjunto de páginas que se obtienen como resultado de esa consulta se le llama conjunto raíz (**root set**).
- El conjunto inicial se amplía con todas las páginas que enlazan o son enlazadas a (o desde) él.
- Este conjunto ampliado de páginas se llama conjunto base (**base set**).
- El conjunto base puede representarse como un grafo.
- Los hubs y authorities se calculan sobre ese grafo.

Root set y base set



El root set tiene típicamente entre 200 y 1000 páginas mientras que el base set puede llegar a las 5000

Cálculo de hubs y authorities

- **Objetivo:** Calcular para toda página d en el conjunto base una puntuación como hub ($h(d)$) y una puntuación como authority ($a(d)$).
- **Inicialización:** Para toda página d : $h(d) = 1$, $a(d) = 1$.
- Iterativamente **recalcular** los valores h y a de cada página **hasta que converja**.
- **Resultado:** dos listas con las páginas ordenadas atendiendo a las puntuaciones como hub y como authority.

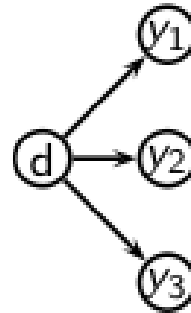


Cálculo iterativo de h y a

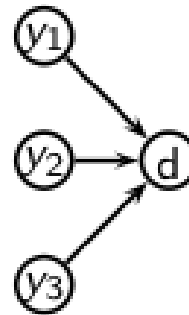
Repetir hasta que converja:

- Para cada documento d :

$$h(d) = \sum_{d \rightarrow y} a(y)$$



$$a(d) = \sum_{y \rightarrow d} h(y)$$





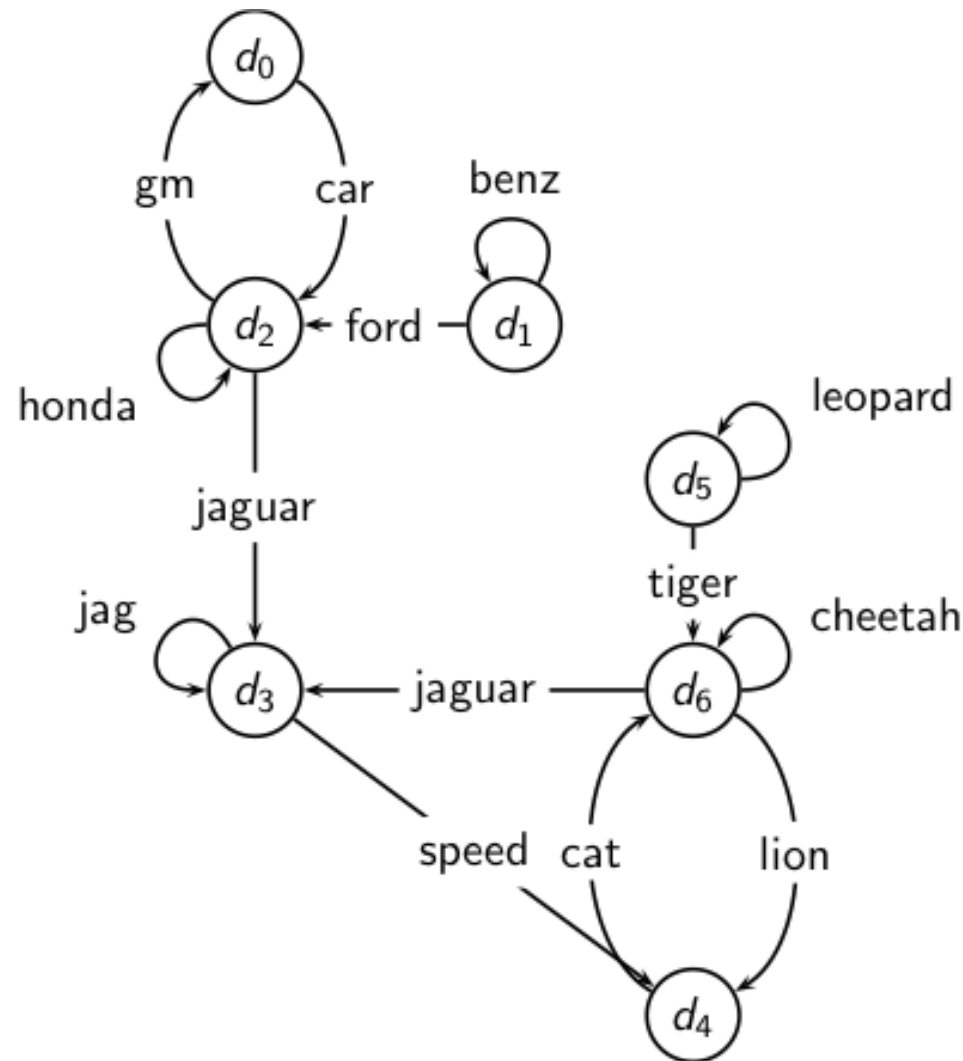
Cálculo iterativo de h y a

Una consideración adicional:

- Para evitar que los valores de h y a crezcan demasiado con cada iteración y
- Para asegurar la convergencia del algoritmo propuesto
 - Se deben escalar los valores de h y a después de cada iteración.

Independientemente del escalado, lo que interesa realmente es el orden relativo de las páginas atendiendo a la puntuación h y a .

Ejemplo de cálculo de HITS (base set)





Ejemplo de cálculo de HITS (base set)

Matriz de enlaces:

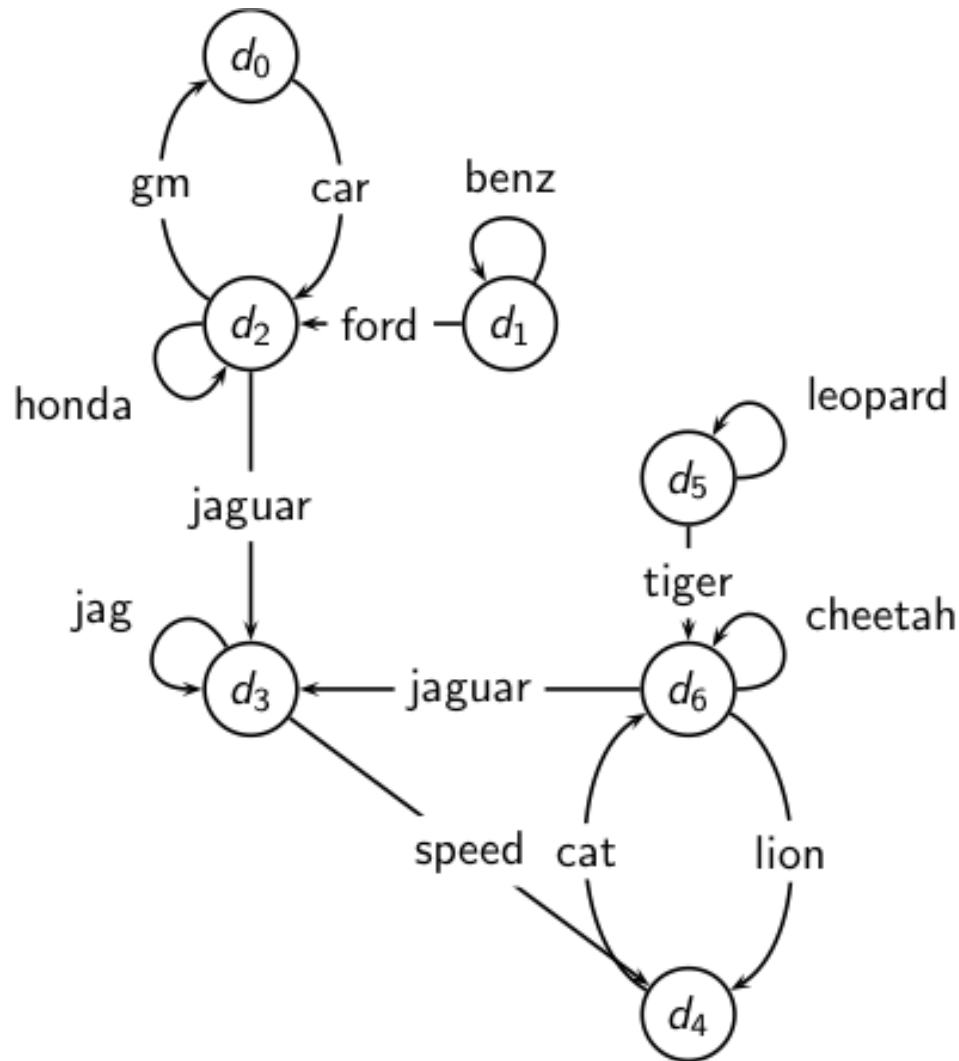
	d_0	d_1	d_2	d_3	d_4	d_5	d_6
d_0	0	0	1	0	0	0	0
d_1	0	1	1	0	0	0	0
d_2	1	0	1	1	0	0	0
d_3	0	0	0	1	1	0	0
d_4	0	0	0	0	0	0	1
d_5	0	0	0	0	0	1	1
d_6	0	0	0	1	1	0	1



Ejemplo de cálculo de HITS (base set)

	<i>Hub</i>							<i>Authority</i>						
	d_0	d_1	d_2	d_3	d_4	d_5	d_6	d_0	d_1	d_2	d_3	d_4	d_5	d_6
t_0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
t_1	1	2	3	2	1	2	3	1	1	3	3	2	1	3
t_2	3	4	7	5	3	4	8	3	2	6	8	5	2	6
t_3	6	8	17	13	6	8	19	7	4	14	20	13	4	15
t_4	14	18	41	33	15	19	48	17	8	31	49	32	8	33
t_5	31	39	97	81	33	41	114	41	18	73	122	81	19	82
<i>Norm.</i>	0.07	0.09	0.22	0.19	0.08	0.09	0.26	0.09	0.04	0.17	0.28	0.19	0.04	0.19
t_6	0.17	0.21	0.54	0.47	0.19	0.23	0.65	0.22	0.09	0.38	0.67	0.45	0.09	0.43
t_7	0.38	0.47	1.28	1.12	0.43	0.53	1.55	0.54	0.21	0.92	1.66	1.12	0.23	1.07
						
<i>Norm.</i>	0.06	0.07	0.22	0.20	0.08	0.09	0.28	0.09	0.03	0.15	0.30	0.20	0.04	0.19

Ejemplo de cálculo de HITS (base set)



	Hub	Authority
d_0	0.06	0.09
d_1	0.07	0.03
d_2	0.22	0.15
d_3	0.20	0.30
d_4	0.08	0.20
d_5	0.09	0.04
d_6	0.28	0.19



Comentarios sobre HITS

- HITS es capaz de encontrar buenas páginas independientemente de su contenido.
- A partir del conjunto base (que requiere de una consulta de usuario) sólo se hace análisis de enlaces, sin preocuparse por el contenido de las páginas.
- Muchas páginas del conjunto base pueden no contener ningún término de la consulta del usuario.
- En teoría, a partir de una consulta en un idioma se pueden recuperar páginas escritas en otros.
- El PageRank de cada página puede estar precalculado, HITS se debe calcular para cada consulta.