Link-Analisis y PageRank

Organización de Datos 75.06

Marzo 2014

Este apunte describe el algoritmo PageRank utilizado por Google para ranquear los resultados de las consultas realizadas por los usuarios y variantes del mismo.

Introducción: La necesidad de PageRank

En un principio los motores dedicados a buscar páginas web trabajaban analizando pura y exclusivamente el contenido de estas páginas. De acuerdo a la consulta del usuario se asignaba un puntaje a cada página que dependía de cosas como la cantidad de veces que los términos buscados aparecían en la página, la distancia entre los mismos, el lugar en que aparecían los términos, etc.

Lamentablemente este método es sensible a que los usuarios modifiquen, maliciosamente, el contenido de sus páginas para quedar mejor posicionados en los resultados de las búsquedas. Por ejemplo para una cierta búsqueda podemos ver que página queda primera, copiar su contenido y pegarlo a nuestra página en texto oculto, luego de esto deberíamos ranquear en los primeros lugares. Otra técnica común era repetir en texto oculto una cierta palabra miles o millones de veces, por ejemplo ponemos "movies" en texto invisible 1 millón de veces en nuestra página y cuando el usuario busque "movies" nuestra página estará entre las primeras aunque ni siquiera se trate sobre ese tema.

El gran éxito de Google fue lograr ranquear los resultados de las búsquedas de forma independiente del contenido de las mismas y eso fue logrado con el algoritmo PageRank.

Concepto básico de PageRank

PageRank se basa en la estructura de links de la Web. El concepto básico es que cada página tiene una cierta "importancia" que es intrínseca y depende de los links que lleven a dicha página. Cuántos mas links nos puedan llevar a una cierta página mas importante será la misma.

Además no todos los links tienen el mismo peso sino que dependen de la importancia de la página origen. Un link desde una página muy importante tiene un peso mucho mayor que un link desde una página cualquiera. Esto quiere decir que con cada link la página propaga su importancia a las páginas linkeadas. Esto lo podemos representar mas o menos con la siguiente fórmula:

$$I(Pi) = \sum I(Pj) / Lj$$

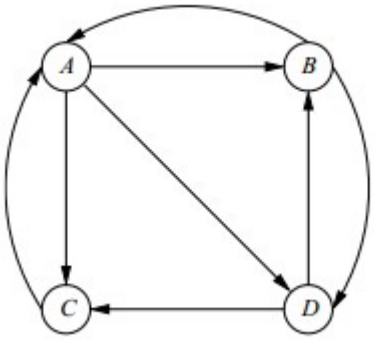
Donde Pj son todas las páginas que tienen un link hacia Pi y Lj es la cantidad de links en total que tiene cada Pj. Por ejemplo una página con importancia "10" y 20 links le propaga 1/2 de importancia a cada una de las 10 páginas linkedas.

Como podemos ver la fórmula es recursiva, para calcular el PageRank de una cierta página necesitamos el PageRank de todas las que linkean a la misma y asi sucesivamente.

Intentaremos resolver este problema dándole una interpretación matemática al mismo.

La gran matriz que representa a la Web

Imaginemos que la Web esta representada por el siguiente grafo:



Imaginemos ahora un usuario que comienza en una cierta página y luego de 1 segundo visitando la misma elige al azar un link cualquiera y pasa a otra página, pasa un segundo en esta nueva página y luego elige un link al azar y pasa a otra y asi sucesivamente.

El tiempo total que el usuario pasa luego de "n" iteraciones es entonces la importancia de la misma. Este es el concepto de "random walker" y una fuerte explicación al algoritmo de PageRank.

Para usar este modelo representaremos a cada página con un vector en donde indicaremos la probabilidad de ir a cada una de las otras páginas. Por ejemplo para el grafo de arriba desde "A" podemos ir a "B", "C" o "D" es decir que el vector de "A" es (0,1/3,1/3,1/3)

Poniendo a cada vector como columnas tenemos una matriz que representa al grafo:

$$\boldsymbol{H} = \begin{bmatrix} 0 & 1/2 & 1 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix}$$

Vamos a representar el PageRank de las páginas también con un vector que inicialmente le da a todas las páginas importancia 1/N en nuestro caso I=(1/4,1/4,1/4,1/4)

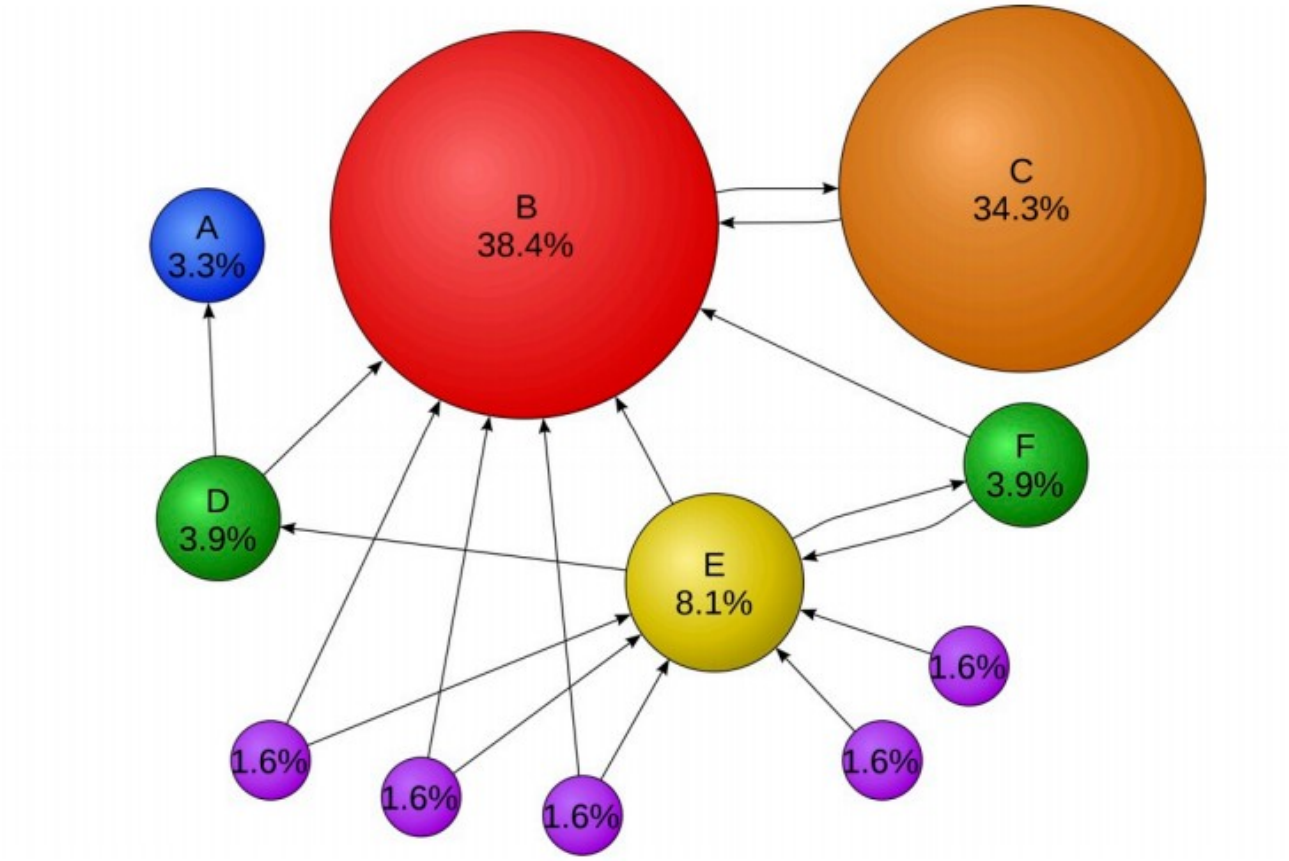
Para calcular el PageRank vamos a realizar el producto de la mátriz H por el vector I: HI y este proceso lo vamos a repetir "k" veces hasta que el vector converja. En nuestro ejemplo:

I0 = (1/4,1/4,1/4,1/4)
I1 = HI0 = (9/24,5/24,5/24,5/24)
I2 = HI1 = (15/48,11/48,11/48,11/49)
....
I = (3/9, 2/9, 2/9, 2/9)

Es decir que la página mas importante es "A" y la importancia de B,C y D es igual.

El Pagerank representa, finalmente la probabilidad de que un usuario que navega al azar termine en

una cierta página.



El gráfico representa la probabilidad de cada página mediante su diametro. Notar que hay muchos links a "B" y que B solo lleva a "C" propagándole casi toda su importancia.

Principios Matemáticos de PageRank

La matriz H tiene algunas características interesantes: todos sus valores son positivos y la suma de cualquiera de sus columnas es 1, esta es la definición de matriz estocástica.

El vector I que calculamos iterativamente es en entonces un autovector de la mátriz H que corresponde al autovalor 1. Para que el método de pagerank funcione (converja) necesitamos que la matriz cumpla con las siguientes características.

- Debe ser estocástica
- Deber ser primitiva
- Debe ser irreducible

Ya sabemos que significa que la matriz sea estocástica.

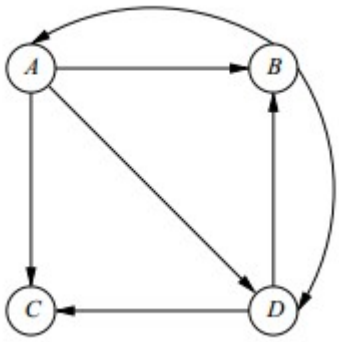
Que sea primitiva significa que para cualquier m H^m deben ser todos valores no-negativos. La intepretación de esto es que para cualquier par de páginas debe ser posible llegar desde una a la otra en CUALQUIER cantidad de links.

Que la matriz sea irreducible quiere decir que no hay una sub-matriz nula dentro de la misma (todos ceros)

Veamos como estas características que son necesarias matemáticamente tienen sentido en nuestro modelo del random surfer.

Dead Ends

Consideremos el siguiente grafo:

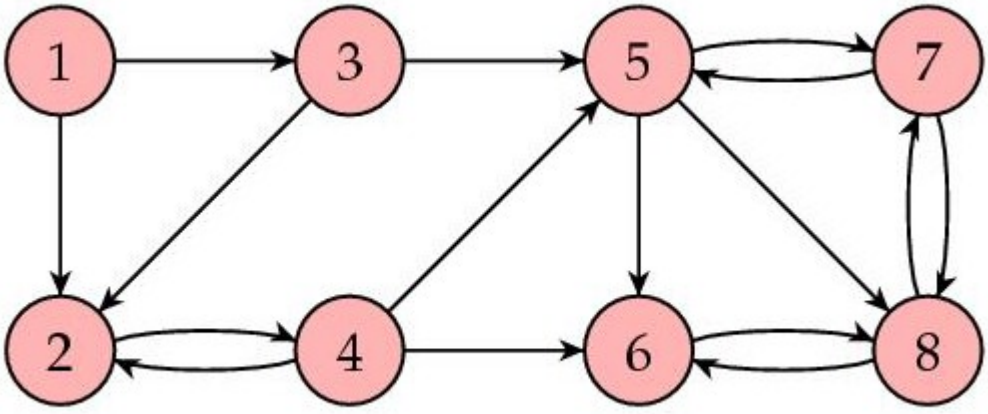


Notemos que la página "C" es un camino sin salida una vez que un usuario navegando al azar llega a "C" no puede salir. Si hacemos el PageRank iterativamente vamos a llegar a I=(0,0,0,0) lo cual no es bueno.

La matriz no es estocástica pues el vector de "C" es todos ceros.

Sub Webs

Sea el siguiente grafo:



Si calculamos el PageRank con nuestro método iterativo llegaremos a que:

I = (0,0,0,0, 0.12, 0.24, 0.24, 0.4)

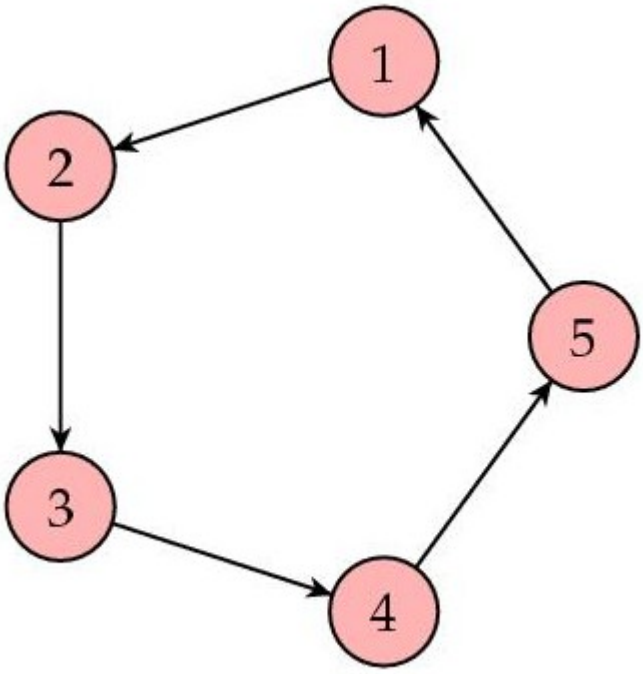
Esto ocurre porque 5,6,7 y 8 forman un sub-grafo del cual no se puede salir! Por eso el PageRank de las primeras páginas queda en cero. Esto tampoco es bueno.

Veamos la matriz:

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/2 & 0 \end{bmatrix}$$

Esta matriz no es irreducible, observemos la esquina superior derecha es una sub-matriz nula.

Otro caso problemático



Si hacemos el PageRank para este ejemplo veremos que el resultado no converge oscila de la forma (1,0,0,0,0); (0,1,0,0,0); (0,0,1,0,0)... etc..

Esto pasa porque la matriz no es primitiva. No se puede llegar de 1 a 5 en cualquier cantidad de links. Solamente se puede llegar en 4 saltos.

Teletransportación

Para solucionar todos los problemas antes mencionados y su justificación matemática Google

inventó para PageRank el concepto de "teletransportación". La idea es muy simple, cada vez que nuestro navegante al azar visita una página tiene dos opciones:

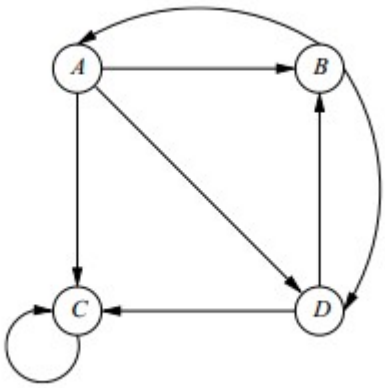
- Seguir un link al azar hacia otra página con probabilidad b
- Teletransportarse magicamente hacia cualquier página al azar con probabilidad (1-b)

El parámetro "b" en general se define entre 0.7 y 0.9 es decir que es mucho mas probable seguir un link que teletransportarse al azar.

Agregar la teletransportación a nuestro modelo es sencillo. Debemos crear una matriz de teletransportacion "T" en donde todos los elementos son 1/N entonces

SH = bH + (1-b)T

Sea el siguiente grafo:



Notamos que "C" es un dead-end, una vez que se llega a C solo se puede volver a C por lo que el PageRank sin teletransportación sería de 1 para "C" y 0 para todas las demas páginas. Es decir que con tiempo suficiente cualquier navegador al azar quedaría atrapado en "C".

Usando b=0.8 podemos construir la matriz con teletransportación como:

$$\mathbf{v}' = \begin{bmatrix} 0 & 2/5 & 0 & 0 \\ 4/15 & 0 & 0 & 2/5 \\ 4/15 & 0 & 4/5 & 2/5 \\ 4/15 & 2/5 & 0 & 0 \end{bmatrix} \mathbf{v} + \begin{bmatrix} 1/20 \\ 1/20 \\ 1/20 \\ 1/20 \end{bmatrix}$$

Notemos que cada elemento del vector de teletransportación es (1-b) * 1/4, es decir 0.2 * 1/4 = 1/20

Haciendo iteraciones con esta fórmula nos queda:

(1/4,1/4,1/4,1/4)
(9/60,13/60,25/60,13/60)
...
(15/148,19/148,95/148,19/148)

C sigue siendo la página mas importante pero el efecto se ha atenuado notablemente.

PageRank en la Práctica

En la práctica la matriz de links tiene unas 25.000 millones de filas y columnas (y creciendo), la forma en que se almacena la matriz y se calcula el PageRank es crítica.

En primer lugar debemos notar que la matriz sin teletransportación es una matriz extremadamente dispersa, la mayoría de sus elementos son ceros. En promedio se estima que una página web tiene unos 10 links por lo que cada columna de la matriz tiene 10 elementos sobre un total de 25000 millones.

Existen técnicas muy eficientes para almacenar matrices dispersas y en la práctica la matriz se almacena en bloques, submatrices de un tamaño k fijo. Es decir que almacenamos muchos bloques de kxk elementos donde cada bloque es disperso.

La matriz de teletransportación tiene todos sus elementos iguales por lo que no es necesario almacenarla realmente. Es posible calcular, en cada iteración, el producto de la matriz de links por el vector de páginas y sumar el factor de teletransportación como un vector en donde cada elemento del vector es.

(1-b) * 1/N siendo N el total de páginas indexadas

Es decir que en lugar de hacer (M + T) x V = I
Hacemos M x V = Io + T = I

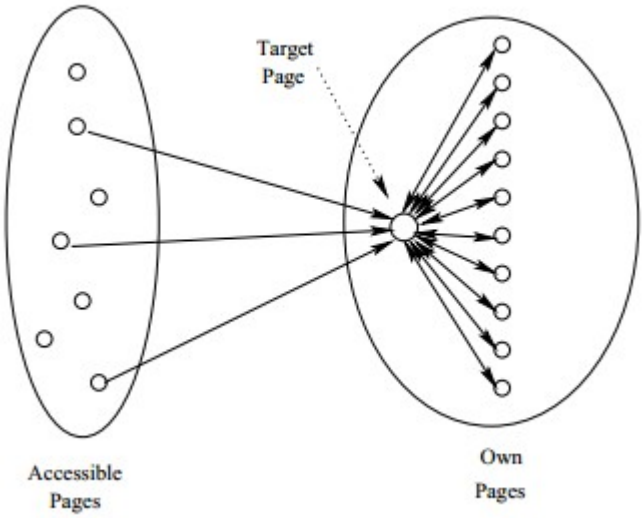
El producto de la matriz de links por el vector que almacena el PageRank se hace con MapReduce, en el curso hemos visto métodos eficientes para calcular MxV cuando ni M ni V entran en memoria.

En la práctica Google utiliza b=0.85 y el cálculo de PageRank con 50-100 iteraciones tarda aproximadamente un día o dos, el PageRank se mantiene en actualización constante.

Link Spam

La teletransportación soluciona los problemas por los cuales la matriz de links puede no cumplir los requisitos necesarios para que el valor del PageRank converja. Pese a ello es posible construir artificialmente estructuras de páginas web que tengan como objetivo aumentar el PageRank de una determinada página. Esto se conoce como "Link Spam" o tambiéen "PageRank Farms".

Veamos uan estructura típica de un Link Spam



Tenemos una página a la cual queremos aumentarle el PageRank. Por un lado creamos links hacia nuestra página desde sitios accesibles por los usuarios. En general esto implica el agregado de un link en comentarios de foros, facebook, twitter, diarios, etc.

Por otro lado creamos una estructura de miles, millones de páginas web y ponemos un link desde nuestra página a cada una de estas y vice-versa.

Para mitigar este problema es necesario distinguir de alguna forma páginas que son confiables de aquellas que no lo son. Supongamos por el momento que saber que páginas son confiables es sencillo.

Es muy raro que una página que es confiable linkee a una que es spam por lo tanto el objetivo es darle un mayor peso a los links salientes desde páginas confiables.

Una forma sencilla de hacer esto en PageRank es crear un vector de teletransportación confiable en la cual solamente tienen valor las páginas que son confiables. Es decir que nuestro navegante azaroso o bien sigue un link con probabilidad "b" o bien se teletransporta a una página confiable con probabilidad 1-b.

Por ejemplo un vector de la forma (0,1/10,0,1/10,0,0,0)

Podemos entonces calcular el PageRank usando ambos vectores y de esta forma calcular el PageRank de la página y el PageRankConfiable. La diferencia entre ambos valores es el "Indice de Spam" de la página.

Luego una vez calculado el índice de spam de cada página es posible eliminar las páginas que son Spam simplemente estableciendo un cierto umbral.

Nos quedaría simplemente poder establecer cuáles páginas son confiables, en la actualidad este proceso se hace por un lado manualmente y por otro lado estableciendo ciertos dominios que son confiables. Por ejemplo las páginas en dominios .gov o .edu no suelen ser spam y se consideran confiables.

PageRank Temático

El método descripto para combatir el link-spam sirve también para realizar un PageRank temático. La idea es que a partir de una consulta sobre un determinado tema le demos mayor peso a las páginas que tratan sobre dicho tema.

Por ejemplo: "Jacksonville Jaguars" es una consulta que tiene que ver con deportes las páginas que tengan que ver con deportes deberían ranquear mejor que páginas que tengan que ver con la ciudad de Jacksonville o los autos Jaguar o incluso los Jaguares del Amazonas.

Para realizar un pagerank temático es necesario tener pre-establecidos vectores de teletransportación por tema, luego determinar el tema a partir de la consulta y finalmente realizar el cálculo del PageRank usando el vector indicado.

Inferir un tema a partir de una serie de términos es un tema bastante interesante y que escapa al alcance de este apunte por lo que lo vamos a dejar aqui, como una forma de darle y no darle un cierre al tema al mismo tiempo ;-)