PageRank y HITS

Felipe Bravo Márquez

8 de noviembre de 2013

Analizando la Web como un Grafo

- La Web es una colección de documentos interconectados por hipervínculos (links).
- Se modela como un grafo dirigido donde los vértices son documentos y las aristas son links.
- Generalmente, cuando un sito **A** apunta a un sitio **B** ($A \rightarrow B$). Se asume que el autor de **A** aprueba el contenido de **B**.[Manning et al., 2008]
- Los motores de búsqueda consideran para rankear documentos para una consulta además de la similitud de contenido, la popularidad del documento dentro del grafo Web.
- Una página es considerada popular cuando es muy apuntada, lo que tiene relación con la *centralidad* del vértice en el grafo.

PageRank [1]

- Tanto la Web como una red de publicaciones y sus citas son redes de información.
- Se puede realizar la analogía de la Web hacia sus link como para los papers hacia sus citas.
- Mientras mayor sea el número de citas de un paper, mayor es su impacto y más confiable es su contenido.
- No es lo mismo ser citado por un paper que con el tiempo se vuelve muy citado a ser citado por un paper que pasa al olvido .
- **Idea circular**:el voto o citación es ponderado de acuerdo al índice de impacto.
- Este concepto fue inventado en Bibliometría en 1960 por Pinsker y Narin.

- Brin & Page lo introducen en 1998 en la Web como Pagerank como propuesta para el buscador Google. [Brin and Page, 1998]

El random surfer [1]

- Consideremos un *random surfer* que sigue un paseo aleatorio navegando a través de los links desde una página inicial.
- Cuando el *random surfer* se encuentra en la página *A*, éste puede seguir navegando por cada uno de sus *out-links* de manera **equiprobable**.

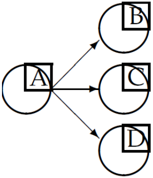

Figura: El random surfer seguirá por B,C o D con probabilidad $1/3$

- Las páginas que visite el surfista aleatorio con mayor frecuencia deben ser más importantes.
- El surfista se teletransporta cualquier sitio de manera uniforme si no existen *out-links* y puede **teletransportarse** en cualquier momento con probabilidad $0 < \alpha < 1$ a cualquier sitio del grafo.

PageRank [2]

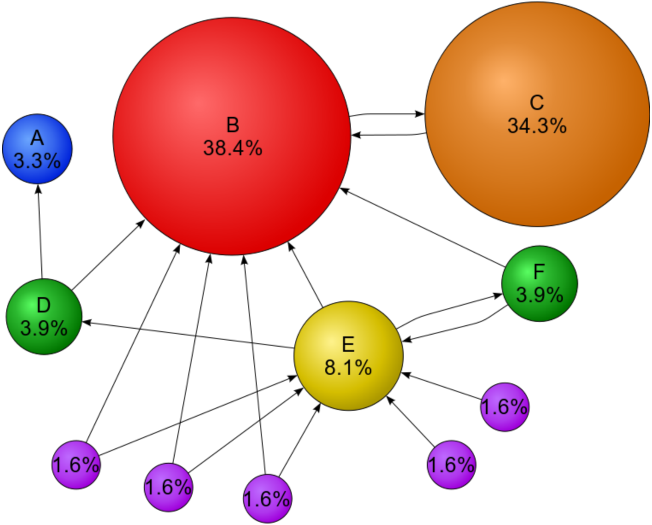

Figura: Fuente: `http://en.wikipedia.org/wiki/PageRank`

PageRank y Markov

El paseo aleatorio del random surfer se modela con cadenas de Markov discretas.

Markov

- Una cadena de Markov discreta es un **proceso estocástico** que ocurre en una serie de pasos de tiempo donde se toman decisiones aleatorias.
- Consiste en N estados (páginas) y una matriz de transiciones entre estados de $N \times N$ llamada P con valores $\in [0,1]$ además $\forall i, \sum_{j=1}^{N} P_{ij} = 1$.
- En una cadena de Markov, el próximo estado depende solamente del estado actual.
- Una matriz con entradas no negativas, que satisfaga la ecuación anterior se denomina como una **matriz estocástica**.
- En cada paso del proceso, estamos en un estado particular (una página a la vez).
- Cada elemento P_{ij} representa la **probabilidad de transición** desde el estado i al estado j.

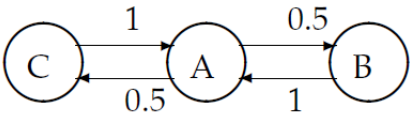

Figura: Cadena de Markov de tres estados

- La cadena de la figura tendría la siguiente matriz P de (3×3) de transiciones:

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

- Todas las filas suman 1.

PageRank y Probabilidades Estacionarias

- Cuando calculamos PageRank, buscamos un vector de probabilidades **estacionarias** sobre la matriz de transiciones P.
- La probabilidad estacionaria de un estado π_i representa la probabilidad de llegar a ese estado cuando la cantidad de transiciones tiende a infinito.
- El **PageRank** de una página, es su probabilidad estacionaria.
- Sea $\eta(i,t)$ la cantidad de veces que se ha caído en el estado i para el período t en un paseo aleatorio sobre el grafo:

$$\pi_i = \lim_{t \to \infty} \frac{\eta(i,t)}{t}$$

Vector de Probabilidades Estacionarias

Ergocidad

- Una matriz estocástica admite probabilidad estacionarias sólo si es **ergódica**.
- Donde una cadena de Markov ergódica debe ser **irreductible** y **aperiódica**.
- La **irreductibilidad** exige que haya un camino desde cualquier página a otro (lo solucionamos con la teletransportación).
- La **aperiodicidad** exige que no se caiga en ciclos debido a referencias circulares tipo $A \rightarrow B$ y $B \rightarrow A$. (Se soluciona con teletransportación y se mejora podando las referencias circulares del grafo).

Vector de Probabilidades Estacionarias [2]

Vector Propio Izquierdo

- Una matriz estocástica ergódica tiene un ***vector propio izquierdo principal*** $\vec{\pi}$ correspondiente a su *valor propio* λ de mayor valor. $\vec{\pi}P = \lambda\vec{\pi}$
- El vector propio principal es equivalente al vector de probabilidades estacionarias $\vec{\pi}$.
- Por el teorema de **Perron-Frobenius** sabemos que para matrices estocásticas el mayor valor propio λ vale siempre 1, entonces

$$\vec{\pi}P = \vec{\pi}$$

- Podemos partir con un $\vec{\pi}_0$ y recomputar $\vec{\pi}P$ hasta converger al vector de probabilidades estacionarias (Iteración de Potencias).
- **Ejemplo**: Sea el grafo Web de nodos $1, 2, 3$ con la siguiente estructura de links: $1 \rightarrow 2, 3 \rightarrow 2, 2 \rightarrow 1, 2 \rightarrow 3$.
- Comenzamos definiendo una matriz de adyacencia A tal que A_{ij} vale 1 si i apunta a j y 0 caso contrario. Para el ejemplo A es una matriz de 3×3.

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

- Debemos transformar esta matriz en una matriz de transición estocástica y ergódica.
- Si alguna fila no tiene 1's reemplazamos sus valores por $1/N$.
- Dividimos cada valor 1 en A por la cantidad de 1's en su fila, para respetar propiedad estocástica.

$$\begin{pmatrix} 0 & 1 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1 & 0 \end{pmatrix}$$

- Multiplicamos la matriz resultante por el escalar $1 - \alpha$, generalmente se usa $\alpha = 0{,}25$ (teletransportación) ahora usaremos $\alpha = \frac{1}{2}$.

$$\begin{pmatrix} 0 & 1/2 & 0 \\ 1/4 & 0 & 1/4 \\ 0 & 1/2 & 0 \end{pmatrix}$$

- Sumamos $\frac{\alpha}{N}$, $(1/6$ en el ejemplo) a todas las entradas de la matriz resultante y obtenemos P. La matriz de transición del *random surfer*.

$$P = \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

- Imaginemos que el surfista comienza en la página 1 tomamos como $\vec{\pi}_0 = [1, 0, 0]$.
- Computamos $\vec{\pi}_1 = \vec{\pi}_0 P = [1/6, 2/3, 1/6]$.
- Computamos $\vec{\pi}_2 = \vec{\pi}_1 P = [1/3, 1/3, 1/3]$.
- Iteramos: $\vec{\pi}_3 = [1/4, 1/2, 1/4]$, $\vec{\pi}_4 = [7/24, 5/12, 7/24]$
- Después de varios pasos convergemos a $\vec{\pi} = [5/18, 4/9, 5/18]$.
- Podemos usar un criterio de parada $||\vec{\pi}_{i+1} - \vec{\pi}_i|| < \epsilon$ [Velasquez and Palade, 2008].

Conclusiones de PageRank

- La actualización del *PageRank* de una página se puede representar con la siguiente expresión:

$$x_p^{i+1} = \frac{\alpha}{N} + (1-\alpha) \sum_{\forall q,p/q \rightarrow p} \frac{x_q^{(i)}}{outdeg(q)}$$

- PageRank rankea las páginas de manera independiente de una consulta.
- Para tener un PageRank elevado no basta con ser muy apuntado (in-degree) es necesario ser apuntado por páginas con alto PageRank.
- PageRank se puede calcular off-line, osea se puede tener precalculado para cuando llega una consulta. Los motores de búsqueda lo precalculan.
- Variaciones de PageRank se usan en el proceso de **crawling** para darle prioridad a sitios más relevantes en la cola de prioridad.**OPIC** (Online Page Importance Computation).

HITS

- HITS (Hyperlink-Induced Topic Search) [Kleinberg, 1998] propone que dada una consulta q cada página existente en el grafo Web tendrá dos tipos de puntajes asociados (un puntaje hub y un puntaje authority).
- Al igual que PageRank, HITS propone que los links entregan información semántica no necesariamente contenida en el **matching de texto** entre una consulta y un documento.
- Para cada consulta se construye un subgrafo de la Web, donde se computa un puntaje de **hub** y otro de **authority** para cada documento del subgrafo donde:
  1. Una página con alto puntaje de **authority** proveerá información relevante para la consulta.
  2. Una página con alto puntaje de **hub** proveerá links a sitios relevantes para la consulta.
- Entonces un buen **hub** apunta a buenas **authorities** y una buena **authority** es apuntada por buenos **hubs**.

Hubs y Autoridades

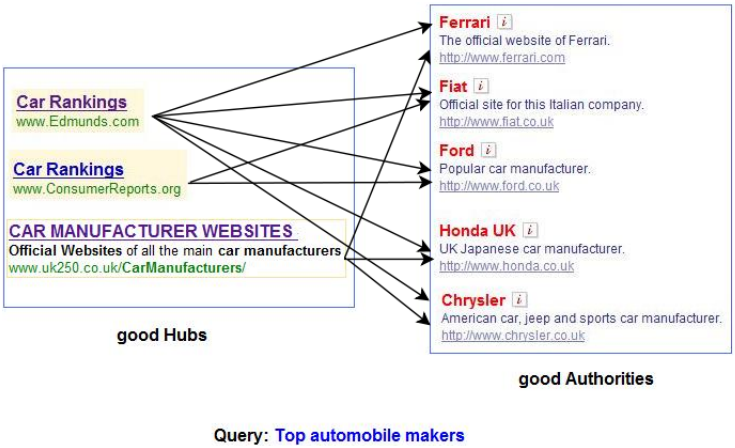

Figura: Hubs y Autoridades para una consulta s **q= Top automobile makers**

- Se modelan los puntajes hubs y authority de una página de la siguiente forma:

Definiciónes

- El puntaje hub de un sitio v $h(v)$ es la suma del puntaje de autoridad de todos los sitios que apunta:

$$h(v) \leftarrow \sum_{v \to y} a(y)$$

- El puntaje de autoridad de un sitio v $a(v)$ es suma del puntaje hub de todos los sitios que lo apuntan:

$$a(v) \leftarrow \sum_{y \to v} h(y)$$

Escogiendo un subconjunto de la Web

- Es muy importante en HITS el **subconjunto de la Web** que se procesa dada una consulta q.
- Se esperan encontrar documentos que no necesariamente son similares textualmente a la consulta pero que si son buenos satisfaciendo una necesidad de información.
- Por ejemplo para una consulta sobre **autos** una página que sólo contiene imágenes de autos no sería bien rankeada con tf-idf, pero probablemente sea apuntada por documentos que si contienen alta similitud textual.
- Se espera que las páginas hubs contengan mayor similitud textual, pero las autoridades sean mejores resolviendo la necesidad de información asociada a la consulta.
- Las autoridades aportan al ranking, mientras que los hubs no [Manning et al., 2008].
- Dada una consulta q se toman los primeros k documentos mediante alguna medida de rankeo textual (tf-idf)
- Ese conjunto S se exande agregando todos los in-links y out-links del conjunto para obtener S' como subgrafo Web a procesar.

Computando HITS

- Se tiene la matriz de adyacencia A sobre el subgrafo Web donde se computarán los puntajes $h(v)$ y $h(a)$ $\forall v \in G$.
- Donde en A, $A_{ij} = 1$ si $v_i \rightarrow v_j$ para todo i, j y 0 en caso contrario.

Relaciones Matriciales

- Matricialmente se puede representar los puntajes de **hub** y **authority** como:

$$\overrightarrow{h} \leftarrow A\overrightarrow{a}$$
$$\overrightarrow{a} \leftarrow A^T\overrightarrow{h}$$

- Si A_{ij} modela los existencia de out-links de i a j, A^T_{nm} modela la existencia de un in-link de n a m (m apunta a n).
- Reemplazamos en ambas ecuaciones para obtener definiciones recursivas:

$$\overrightarrow{h} \leftarrow AA^T\overrightarrow{h}$$
$$\overrightarrow{a} \leftarrow A^TA\overrightarrow{a}$$

Relaciones Matriciales [2]

- Las relaciones anteriores se parecen a las relaciones de vectores propios usadas en Pagerank, entonces si pasamos las relaciones de $\leftarrow$ a relaciones de igualdad, aparecen valores propios desconocidos porque nuestras matrices **no son estocásticas** y no necesariamente se cumple $\lambda = 1$ como en Pagerank :

$$\overrightarrow{h} = (1/\lambda_h)AA^T\overrightarrow{h}$$

$$\overrightarrow{a} = (1/\lambda_a)A^TA\overrightarrow{a}$$

- Para poder realizar iteración en potencias (como en PageRank), se normalizan los vectores $\overrightarrow{h}$ y $\overrightarrow{a}$ tal que la suma de sus elementos sume 1 en cada interacción.

Ejemplo

- **Ejemplo**: Supongamos que a partir de una consulta obtenemos un sub-grafo Web de nodos $1, 2, 3$ con la siguiente estructura de links: $1 \rightarrow 2, 1 \rightarrow 3, 2 \rightarrow 3, 3 \rightarrow 1$.

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

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

- Luego $\overrightarrow{h} \leftarrow AA^T\overrightarrow{h}$ con $h_0 = [1\ 1\ 1]^T$ con

$$AA^T = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- $h_1 \leftarrow AA^T\overrightarrow{h} = [3\ 2\ 1]^T$ Normalizamos y obtenemos $h_1 = [0{,}5\ 0{,}33\ 0{,}166]^T$ $h_2 = [0{,}571437\ 0{,}357148\ 0{,}07143]^T = h_3$

Ejemplo [2]

- Nos damos cuenta que el vector de hubs tiene sentido el sitio de acuerdo a la definición, 1 es el que más apunta por lo que tiene mayor peso, y el sitio 2 pesa más que el sitio 3 pues apunta al sitios con mayor autoridad.
- Ahora calculamos el vector de **autoridades** con $a_0 = [1\ 1\ 1]^T$

$$A^T A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

- $\overrightarrow{a} \leftarrow A^T A \overrightarrow{a} = [1\ 2\ 3]^T$, normalizamos y nos queda $a_1 = [0{,}1666\ 0{,}3333\ 0{,}5]$ $a_2 = [0{,}071429\ 0{,}357143\ 0{,}5471429]^T$ $a_3 = [0{,}028571\ 0{,}37149\ 0{,}6]^T = a_4$

References I

Brin, S. and Page, L. (1998).
The anatomy of a large-scale hypertextual web search engine.
Computer Networks and ISDN Systems, 30(1-7):107–117.
Proceedings of the Seventh International World Wide Web Conference.

Kleinberg, J. M. (1998).
Authoritative sources in a hyperlinked environment.
In *SODA*, pages 668–677.

Manning, C. D., Raghavan, P., and Schütze, H. (2008).
Introduction to Information Retrieval.
Cambridge University Press, New York, NY, USA.

Velasquez, J. D. and Palade, V. (2008).
Adaptive Web Sites: A Knowledge Extraction from Web Data Approach.