Algorithmique Avancée
Google PageRank Algorithm

Serigne A. Gueye

Novembre 2014, CERI

1 Introduction

PageRank est un algorithme de classification de pages web à la base du moteur de recherche Google. La méthode a été proposée en 1998 par Larry Page et Sergey Brin. Une bibliographie des sources utilisées pour écrire les explications ci-dessous est donnée dans le site du cours.

2 Processus général de recherche sur le web

Pour répondre à une requête de recherche d'informations sur internet, les moteurs de recherche procèdent généralement en trois étapes.

2.1 Phase 1 : Indexation

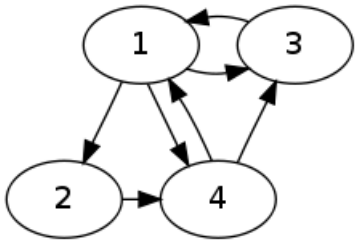
La phase dite **d'indexation** consiste à récupérer des pages webs qui seront **indexées**. L'index étant une liste de mots-clés avec l'indication des pages dans lesquelles ils se trouvent. Le nombre très important de mots-clés possibles rend la question de la structure de données de stockage de cet index important.

L'indexation est faite par des **"robot d'indexation" (en anglais web crawler ou web spider)**. Ces logiciels explorent automatiquement le Web, de façon récursive. C'est à dire en suivant récursivement les liens hypertextes rencontrés à partir d'une page initiale quelconque. Dans les travaux pratiques 3 (sur les graphes) le robot Httrack a été utilisé.

2.2 Phase 2 : Recherche dans l'index

Quand un utilisateur tape sur la page d'accueil de Google, un ou plusieurs mots clés, la second phase consiste à aller rechercher dans le fichier index stocké les pages où se trouvent ces mots clés. Cette recherche aboutit à la récupération de pages pertinentes mais également, et surtout (pour l'algorithme PageRank), des liens qu'elles entretiennent entre elles. La page A sera liée à la page B si A cite B ou inversement. Les liens entre pages pertinentes sont représentés formellement par un **graphe orienté**.

Exemple Supposons que la recherche dans l'index suite à une requête conduisent à identifier des pages liées de la façon suivante.



Toutes les pages représentées dans ce graphe répondent à la requête de recherche.

Il s'agit pour un algorithme du type de celui de PageRank de déterminer un ordre d'importance de ces pages.

□

Soit $G = (V, E)$ un 1-graphe orienté où V est l'ensemble des pages webs considérés et E l'ensemble des liens hypertextes. Désignons par μ_i une valeur exprimant l'importance (ou le poids) de la page $i \in V$. μ_i peut être déterminé par plusieurs méthodologies.

2.2.1 Classement par les degrés

Dans ce cas l'importance de chaque page est mesurée par son demi-degré intérieur.

$$\mu_i = d^-(i) \tag{1}$$

La page la plus importante sera celle de plus fort degré. Il est commode pour faire la transition avec les autres méthodologies d'interpréter chaque unité du demi-degré intérieur d'un noeud i comme un "vote" pour i. La page la plus importante étant donc celle receuillant le plus de votes.

2.2.2 Normalisation des degrés

En reprenant l'interprétation des degrés comme des votes, on peut observer qu'une page dont le demi-degré extérieur est fort (qui vote pour la quasi-totalité des autres pages) est peu informative sur l'importance des pages. Quand une page j de ce type vote pour une autre page i, il est donc pertinent de "relativiser" le vote de j par rapport à la quantité totale $d^+(j)$ de pages pour lesquelles elle a voté. La valeur du vote de j n'est plus dans ce cas 1 comme précédemment mais $\frac{1}{d^+(j)}$. Et l'importance de i est égale à la quantité totale de votes qu'il reçoit. Nous obtenons donc :

$$\mu_i = \sum_{j \in \omega^-(i)} \frac{1}{d^+(j)} \ , \ \forall \ i \in V. \tag{2}$$

où $\omega^-(i)$ désigne l'ensemble des arcs entrants en i.

2.2.3 Formule de récurrence (et "ellitisme")

Le mode de calcul précédent peut s'étendre en considérant cette fois l'importance des votants.

Rappelons que si j vote pour i le poids du vote était précédemment considéré comme étant égal à $\frac{1}{d^+(j)}$. Deux pages ayant les mêmes demi-degrés extérieurs seront donc considérés comme ayant les mêmes poids de vote. Or si l'une des pages est en réalité plus importante que l'autre, par rapport à une requête, le poids de son vote devrait aussi être plus important. Sous cet angle, on est important car cité par des pages importantes.

Cet aspect est pris en compte en multipliant le poids du vote de j par son importance. On obtient ainsi :

$$\mu_i = \sum_{j \in \omega^-(i)} \frac{\mu_j}{d^+(j)} \ \forall \ i \in V. \tag{3}$$

Soit $\mu = (\mu_i)_{i \in V}$ un vecteur, et A la matrice carrée définie par :

$$A_{ij} = \begin{cases} \frac{1}{d^+(j)} & si\ j \in \omega^-(i) \\ 0 & sinon \end{cases}$$

La formule de récurrence 3 revient, algébriquement, à résoudre le système d'équations linéaires suivant :

$$\mu = A\mu \tag{4}$$

Notons que si M désigne la matrice d'adjacence du graphe G pondérée par les valeurs $\frac{1}{d^+(i)}$ pour chaque arc (i, j) existant, alors on a :

$$M = A^T \tag{5}$$

Lemme 2.1 *Le système (4) admet une solution si et seulement si* 1 *est valeur propre de* A *et* μ *le vecteur propre associé.*

Preuve . La preuve dérive directement des définitions de valeur et vecteur propres.

∎

Définition 2.2 *Une matrice carrée est dite à colonnes stochastiques (où matrice de Markov) ssi elle ne contient que des éléments dans* $[0, 1]$ *et si la somme des éléments de chaque colonne est toujours égale à* 1*.*

Théorème 2.3 *La matrice* A *est à colonnes stochastiques.*

Preuve ∎

Théorème 2.4 *Toute matrice à colonnes stochastiques admet* 1 *comme valeur propre.* 1 *est de plus sa plus grande valeur propre en module.*

Preuve . Soit M une matrice quelconque à colonnes stochastiques.

$$M = \begin{bmatrix} m_{11} & m_{12} & ... & m_{1n} \\ m_{21} & m_{22} & ... & m_{2n} \\ ... & ... & ... & ... \\ m_{n1} & m_{n2} & ... & m_{nn} \end{bmatrix}$$

Par définition 1 est valeur propre ssi il est racine du polynôme caractéristique $det(M - I)$ où I est la matrice identité.

Or

$$M - I = \begin{bmatrix} m_{11} - 1 & m_{12} & ... & m_{1n} \\ m_{21} & m_{22} - 1 & ... & m_{2n} \\ ... & ... & ... & ... \\ m_{n1} & m_{n2} & ... & m_{nn} - 1 \end{bmatrix}$$

D'autre part, on sait qu'on ne change pas le déterminant d'une matrice en ajoutant à une quelconque une combinaison linéaire des lignes restantes. En ajoutant à la première ligne de $M - I$ la somme de toutes les autres, on obtient :

$$\begin{bmatrix} 0 & 0 & ... & 0 \\ m_{21} & m_{22} - 1 & ... & m_{2n} \\ ... & ... & ... & ... \\ m_{n1} & m_{n2} & ... & m_{nn} - 1 \end{bmatrix}$$

Ce qui implique que $det(M-I) = 0$. 1 est donc bien racine de $det(M-I)$.

∎

Ce résultat montre que le système (4) a bien une solution dont le vecteur solution fournit l'importance de chaque page. Théoriquement, la solution **exacte** pourait être déterminée par un algorithme quelconque de résolution de systèmes linéaires (Algorithme de Gauss par exemple de complexité $O(n^3)$). En pratique, on lui préférera une méthode itérative fournissant, plus rapidement, une **approximation** satisfaisante de la solution exacte.

3 Résolution itérative

Supposons qu'un internaute se trouve à la page 1 de notre exemple au temps $t = 0$ et surfe aléatoirement sur les pages à divers pas de temps $t_0 = 0$, t_1, ...

Soit $P(t)$ le vecteur dont les composantes sont les probabilités que l'internaute se retrouve sur l'une des 4 pages au temps $t = t_0, t_1, ...,$.

On a :

$$P(t) = \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \\ P_4(t) \end{pmatrix}$$

où $P_i(t)$ $(i = 1, 2, 3, 4)$ est la probabilité qu'il se trouve sur la page i au temps t.

Pour $t = t_0 = 0$, comme l'internaute se trouve sur la page 1, on a :

$$P(t_0) = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Quand l'internaute est sur une page i quelconque $(i = 1, 2, 3, 4)$, notons $P(i, j)$ la probabilité qu'il aille de i à la page j si l'arc (i, j) existe. notons M_{ij} la valeur suivante :

$$M_{ij} = \begin{cases} 0 & si\ l'arc\ (i,j)\ n'existe\ pas \\ \\ P(i,j) = \frac{1}{d^+(i)} & sinon \end{cases}$$

et $M = \{M_{ij}\}_{i,j=1,2,3,4}$ la matrice correspondante.

Observons que M est en fait la matrice d'adjacence du graphe des pages tel que décrite dans la sous-section (formule de récurrence).

$$M = \begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

On a alors :

$$P_1(t_1) = P_1(t_0) * M_{11} + P_2(t_0) * M_{21} + P_3(t_0) * M_{31} + P_4(t_0) * M_{41} = 0$$
$$P_2(t_1) = P_1(t_0) * M_{12} + P_2(t_0) * M_{22} + P_3(t_0) * M_{32} + P_4(t_0) * M_{42} = \frac{1}{3}$$
$$P_3(t_1) = P_1(t_0) * M_{13} + P_2(t_0) * M_{23} + P_3(t_0) * M_{33} + P_4(t_0) * M_{43} = \frac{1}{3}$$
$$P_4(t_1) = P_1(t_0) * M_{14} + P_2(t_0) * M_{24} + P_3(t_0) * M_{34} + P_4(t_0) * M_{44} = \frac{1}{3}$$

En écriture matricielle, toutes les équations données ci-dessus se résument comme le système :

$$P(t_1) = M^T P_(t_0) = A \times P(t_0)$$

De manière analogue, au temps $t = t_2$ on a :

$$P(t_2) = A \times P(t_1)$$

Et de façon générale :

$$P(t_n) = A \times P(t_{n-1}) = A^n \times P(t_0) \ \forall \ n \geq 1$$

On a alors le théorème ci-dessous montrant la convergence de $P(t_n)$ vers le vecteur μ mesurant l'importance des pages.

Théorème 3.1 (Perron - Frobenius) $\lim_{n\to\infty} P(t_n) = \mu$ *pour toute matrice* A *à colonnes stochastiques.*

Preuve ∎

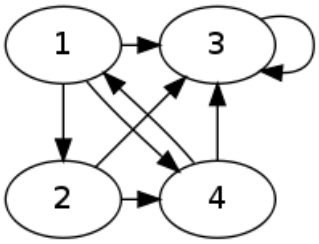
Notons que comme :

$$P(t_n) = A \times P(t_{n-1}) = \ \forall \ n \geq 1$$

On a aussi :

$$\lim_{n\to\infty} P(t_n) = A \lim_{n\to\infty} \times P(t_{n-1}) = \mu = A \times \mu,$$

nous ramenant ainsi au système (4).



4 "Narcissisme" et Algorithme PageRank

Quand une page est citée (comme la page 3 ci-dessus), mais ne cite aucune autre page sauf éventuellement elle-même, on peut prouver, que quelque soit la position de départ donnée par $P(t_0)$, on aura :

$$\lim_{n\to\infty} P(t_n) = \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \\ P_4(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

De façon générale, on atteindra tôt ou tard cette page sans ne plus pouvoir en sortir.

Pour éviter d'être bloqué par ce que l'on nomme par analogie à l'astronomie ces "trous noirs" L'algorithme, PageRank considère que l'internaute à une probabilité $(1-p)$ de suivre les liens existants (p donnée). Dans ce cas, la probabilité d'atteindre une page se calculera comme précédemment. Mais il a une probabilité p de se "téléporter" vers un autre noeud quelconque. Chaque noeud dans ce cas peut s'atteindre de façon équiprobable $(\frac{1}{n})$.

Le calcul de $P_(t_1)$ se réécrit ainsi de la manière suivante :

$$\begin{aligned}
P_1(t_1) &= (1-p) * (P_1(t_0) * M_{11} + P_2(t_0) * M_{21} + P_3(t_0) * M_{31} + P_4(t_0) * M_{41}) + p\tfrac{1}{n} \\
P_2(t_1) &= (1-p) * (P_1(t_0) * M_{12} + P_2(t_0) * M_{22} + P_3(t_0) * M_{32} + P_4(t_0) * M_{42}) + p\tfrac{1}{n} \\
P_3(t_1) &= (1-p) * (P_1(t_0) * M_{13} + P_2(t_0) * M_{23} + P_3(t_0) * M_{33} + P_4(t_0) * M_{43}) + p\tfrac{1}{n} \\
P_4(t_1) &= (1-p) * (P_1(t_0) * M_{14} + P_2(t_0) * M_{24} + P_3(t_0) * M_{34} + P_4(t_0) * M_{44}) + p\tfrac{1}{n}
\end{aligned}$$

On a peut avoir une écriture matricielle de ce système en posant

$$R = (1-p)A + pB$$

où

$$B = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

On en déduit que

$$P(t_n) = R \times P(t_{n-1}) = R^n \times P(t_0)$$

Théorème 4.1 *R est une matrice à colonnes stochastiques*

Preuve ∎

En appliquant le théorème 3.1 de Perron-Frobenius, on en déduit la convergence du procédé itératif vers un nouveau vecteur d'importance des pages

Corollaire 4.2 $\lim_{n\to\infty} P(t_n) = \lim_{n\to\infty} R^n \times P(t_0) = \nu$