Algorithme Page Rank

Source : tangente-google.pdf

1 Histoire de Google

1.1 Google

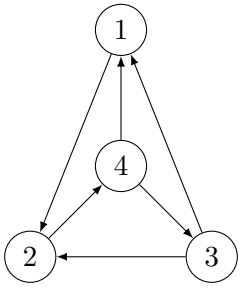
Depuis plus d'une décennie, Google domine le marché des moteurs de recherche sur Internet. Son point fort ? Il trie intelligemment ses résultats par ordre de pertinence. Son fonctionnement repose en fait sur une judicieuse modélisation mathématique.
Depuis sa conception en 1998, Google continue à évoluer et la plupart des améliorations demeurent des secrets bien gardés. L'idée principale, par contre, a été publiée et est disponible en ligne. Le pilier du succès de Google est une judicieuse modélisation mathématique.
Le but de devoir est de comprendre cette modélisation et de l'expérimenter numériquement.

1.2 Moteur de recherche

Une base de données a une structure prédéfinie qui permet d'en extraire des informations, par exemple « nom, rue, code postal, téléphone… ». L'Internet, par contre, est peu structuré : c'est une immense collection de textes de nature variée. Toute tentative de classification semble vouée à l'échec, d'autant plus que le Web évolue rapidement : une multitude d'auteurs ajoutent constamment de nouvelles pages et modifient les pages existantes. Pour trouver une information dans ce tas amorphe, l'utilisateur pourra lancer une recherche de mots clés. Ceci nécessite une certaine préparation pour être efficace : le moteur de recherche copie préalablement les pages Web en mémoire locale et trie les mots par ordre alphabétique. Le résultat est un annuaire de mots clés avec leurs pages Web associées. Pour un mot clé donné, il y a typiquement des milliers de pages correspondantes (plus d'un million pour « tangente » par exemple). Comment aider l'utilisateur à repérer les résultats potentiellement intéressants ? C'est sur ce point que Google a apporté sa grande innovation.

1.3 La Toile est un graphe

Profitons du peu de structure qui soit disponible. L'Internet n'est pas une collection de textes indépendants, mais un immense hypertexte : les pages se citent mutuellement. Afin d'analyser cette structure, nous allons négliger le contenu des pages et ne tenir compte que des liens entre elles. Ce que nous obtenons est la structure d'un graphe, dont la figure suivante montre un exemple en miniature.



Dans cet exemple, la page 1 cite la page 2, la page 2 cite la page 4, la page 3 cite les pages 1 et 2 et la page 4 cite les pages 1 et 3.

1.4 Définition d'un graphe orienté

Un graphe orienté est un ensemble de points nommés nœuds reliés par des flèches nommées arêtes. La relation va dans un seul sens et est donc asymétrique.

1.5 Représentation d'un graphe

On représente un graphe à l'aide d'une matrice carré, $A = (a_{i,j})_{1 \leqslant i,j, \leqslant n} \in \mathcal{M}_n(\mathbb{R})$. La taille de la matrice, n, est le nombre de nœuds et $a_{i,j}$ est égale à 1 si il existe une flèche partant du nœud i et allant au nœud j. Dans l'exemple ci- dessus, on a :

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

Cette matrice est appelée **matrice d'adjacente**.

1.6 Pertinence

Les liens sur Internet ne sont pas aléatoires mais ont été édités avec soin. Quels renseignements pourrait nous donner ce graphe ? L'idée de base, encore à formaliser, est qu'un lien $j \to i$ est une recommandation de la page P_j d'aller lire la page P_i. C'est ainsi un vote de P_j en faveur de l'autorité et de la pertinence de la page P_i. Explorons les différents algorithmes du calcul de la pertinence d'une page P_j, noté m_j.

1.7 Comptage naïf

Il est plausible qu'une page importante reçoit beaucoup de liens. Avec un peu de naïveté, on croira aussi l'affirmation réciproque : si une page reçoit beaucoup de liens, alors elle est importante. Ainsi on pourrait définir l'importance m_i de la page P_i comme le nombre des liens $j \to i$. En formule, ceci s'écrit comme suit :

$$m_i = \sum_{j \to i} 1.$$

Autrement dit, m_i est égal au nombre de « votes » pour la page P_i, où chaque vote contribue par la même valeur 1. C'est facile à définir et à calculer, mais ne correspond souvent pas à l'importance ressentie par l'utilisateur.

1 **Exercice** Écrire un algorithme Python donnant la pertinence des pages à partir de la matrice d'adjacence.

2 **Exercice** Donner une technique afin de manipuler cet algorithme pour qu'une page soit très populaire.

1.8 Comptage pondéré

Certaines pages émettent beaucoup de liens : ceux-ci semblent moins spécifiques et leur poids sera plus faible. Nous partageons donc le vote de la page P_j en l_j parts égales, où l_j dénote le nombre de liens émis. Ainsi on pourrait définir une mesure plus fine :

$$m_i = \sum_{j \to i} \frac{1}{l_j}.$$

Autrement dit, mi compte le nombre de « votes pondérés » pour la page P_i.

3 **Exercice** Écrire un algorithme Python donnant la pertinence des pages à partir de la matrice d'adjacence.

4 **Exercice** Donner une technique afin de manipuler cet algorithme pour qu'une page soit très populaire.

1.9 Comptage récursif

Heuristiquement, une page P_i paraît importante si beaucoup de pages importantes la citent. Ceci nous amène à définir l'importance mi de manière récursive comme suit :

$$m_i = \sum_{j \to i} \frac{m_j}{l_j}.$$

Ici le poids du vote $j \to i$ est proportionnel au poids m_j de la page émettrice. C'est facile à formuler, mais moins évident à calculer

5 **Exercice** Exprimer le système linéaire sous forme matricielle.

6 **Exercice** Expliquer pourquoi le vecteur propre associée à la valeur propre 1 est important

7 **Exercice** Utiliser la bibliothèque Numpy pour déterminer la pertinence des pages à partir de la matrice d'adjacence.

1.10 Surfeur

Sur le web, un ordre de grandeur du nombre de page internet est de 1000 milliards. Il devient impossible d'un point de vue numérique de déterminer la vecteur propre d'une matrice de taille 1000 milliards...
Avant de tenter de résoudre le système linéaire, essayons d'en développer une intuition. Pour ceci, imaginons un « surfeur aléatoire » qui se balade sur Internet en cliquant sur les liens au hasard. Comment évolue sa position ?
À titre d'exemple, supposons que notre surfeur démarre au temps $t = 0$ sur la page P_3. Deux liens partent de la page P_3 , donc au temps $t = 1$ il se trouve sur une des pages P_1, P_2 avec probabilité $1/2$.

8 **Exercice** Écrire un algorithme permettant de déterminer les probabilité de présence après n étapes.

9 **Exercice** Observer la convergence vers le vecteur normalisé du comptage récursif.

1.11 Transition

Comment formaliser la diffusion illustrée ci-dessus ? Supposons qu'au temps t notre surfeur aléatoire se trouve sur la page P_j avec une probabilité p_j. La probabilité de partir de P_j et de suivre le lien $j \to i$ est alors p_j/l_j. La probabilité d'arriver au temps $t + 1$ sur la page P_i est donc :

$$p'_i = \sum_{j \to i} \frac{p_j}{l_j}.$$

Étant donnée la distribution initiale p, cette loi de transition définit la distribution suivante $p' = T(p)$. C'est ainsi que l'on obtient la ligne $t + 1$ à partir de la ligne t dans nos exemples. En théorie des probabilités, ceci s'appelle une *chaîne de Markov*. La mesure stationnaire est caractérisée par l'équation d'équilibre $m = T(m)$, qui est justement notre équation définissant m par un comptage récursif.