

Les algorithmes de classement utilisés dans les moteurs de recherche

Michel Habib

habib@liafa.jussieu.fr

<http://www.liafa.jussieu.fr/~habib>

Moteurs de recherche, cours de M2
année 2009 - 2010

Organisation du cours

Cours : Michel Habib habib@liafa.jussieu.fr

TP : Fabien de Montgolfier fm@liafa.jussieu.fr

Plan

Introduction

Comment classer les pages ?

Le Graphe du WEB

Pagerank

L'indexation des pages Web

La sémantique et les bombes

Graph Mining

Quelques exemples de jeux algorithmiques

- ▶ "Information is not Knowledge", Albert Einstein
- ▶ "Information is not Knowledge. Knowledge comes from theory", W. Edward Deming

Sir Timothy

Timothy Bernes-Lee (1955) is generally credited with inventing the world wide web in 1990.

The internet as such already existed, but it was mostly used for email and newsgroups. It was Berners-Lee, together with his Belgian colleague Robert Cailleau, who established the first successful communication between an HTTP client and server via the Internet, thus creating the first web browser.

The first ever website in the world was

<http://info.cern.ch/hypertext/WWW/TheProject.html>.

There are 162 million websites in the world today.

A quel titre je me permets de parler de ce sujet ?

- ▶ Spécialiste d'algorithmique sur les graphes
- ▶ 3 thèses co-encadrées sur les moteurs de recherche
- ▶ Cours de M2 professionnel à Paris Diderot (1 à 2 embauches par an chez Exalead)
- ▶ Contrats sur le sujet avec Orange et Exalead

Quel rapport avec les mathématiques ?

- ▶ Une très belle application de l'algèbre linéaire
- ▶ des théorèmes de point fixe
- ▶ Des marches aléatoires sur des graphes (chaînes de Markov)

Vocabulaire technique minimal

- ▶ url : Uniform Resource Locator
adresse IP + chemin d'accès
- ▶ html : Hypertext Markup Language
Ce langage possède quelques lacunes
- ▶ http : Hypertext Transfert protocol
Protocole très efficace
- ▶ hyperliens ou liens hypertextes

Fonctionnement d'un moteur de recherche

Données : une question (une chaîne de caractères)

Résultat : une liste ordonnée d'URL associées à la question.

Comment cela marche

1. Extraction du contenu de la question (i.e. quelques mots clés)
2. Recherche de "toutes" les pages WEB qui contiennent ces mots clés
3. Tri et affichage d'une liste ordonnée d'url (munies d'une affichette).

Tri des résultats

L'étape 3 est critique, car il peut y avoir plus de 100 000 réponses.

Une question très pertinente

- ▶ Habib Terroriste
- ▶ Google Results (February 2007) approx 504 000 for Habib terrorist. (0,11 seconds)

Un utilisateur normal ne lit que la ou les premières pages des résultats
D'où l'absolue nécessité d'un classement pertinent des réponses

- ▶ Un moteur de recherche c'est :
- ▶ un gigantesque graphe
+
- ▶ une gigantesque base de données
+
- ▶ des algorithmes efficaces

Le contenu

Un moteur de recherche indexe les pages lisibles de l'extérieur mais aussi tout texte (au format pdf, rtf ou doc) mais aussi des images

tous fichiers dans un format lisible et qui ne sont pas protégés en lecture

Nécessité de l'expérimentation car il y a beaucoup d'intox et le culte du secret dans le domaine

Généricité

Les remarques présentées ici tiennent pour la plupart des moteurs de recherche

Ludique

Jeux expérimentaux avec les élèves sur la question "comment cela marche?" On peut faire varier les moteurs de recherche.

Le principe

- ▶ Il s'agit de calculer **algorithmiquement** (pas à la main) un coefficient entre 0 et 1 associé à chaque page
- ▶ Vu la taille des données, l'algorithme doit être très efficace, ce qui interdit l'analyse précise du contenu des pages

Eviter à tout prix la sémantique

En 1999, plusieurs chercheurs ont proposé une formulation récursive de l'importance d'une page.

Cette importance ne dépendant que de la structure des liens entre les pages html.

N'utiliser que la structure des hyperliens entre les pages permet d'éviter les analyses du contenu des pages (on évite ainsi le recours à des programmes d'analyse de la langue naturelle)

Des méthodes "syntagraphiques"

Interprétation des hyperliens

Méthode inspirée des études sur les citations entre scientifiques, par exemple le classement pondéré de G. Pinsky et F. Narin 1976

A cite B s'interprète comme A vote pour B

La première idée

Utiliser le nombre de liens pointant sur une page

S. Brin, L. Page, R. Motwani, T. Winograd

Principe de l'algorithme de PageRank (Google)

Une page a un score d'autant plus élevé qu'elle est référencée par des pages ayant un score élevé

Une définition récursive de l'importance d'une page

Larry Page and Serguei Brin



Principe de la méthode HITS : Hypertext Induced Topic Search

J. Kleinberg

Pour chaque page on calcule de concert deux scores : un coefficient d'autorité et un coefficient d'annuaire (hub)

Une page a un coefficient d'autorité d'autant plus élevé qu'elle est référencée par des pages ayant un coefficient d'annuaire élevé

Une page a un coefficient d'annuaire d'autant plus élevé qu'elle est référencée par des pages ayant un coefficient d'autorité élevé.

Jon Kleinberg



Le graphe du WEB

- ▶ Le graphe du Web un graphe orienté
- ▶ Les sommets sont les pages html, appelées ici pages (10 milliards de pages estimées actuellement)
- ▶ Les arcs correspondent aux hyperliens entre ces pages

Beaucoup de choses ont été écrites sur ce graphe ...

Le fameux modèle du noeud papillon
Broder et al. (2000)

Graphe petit monde

Les degrés vérifient une loi de puissance
 $\log(\text{Prob}(d^-(p) = k)) = \alpha - \lambda \log(k)$ avec $\lambda = 2.1$ pour les degrés entrants et $\lambda = 2.72$ pour les degrés sortants.

Biais introduit par l'outil

T. Bennouas, F. de Montgolfier 2007

La plupart des propriétés trouvées proviennent en fait des méthodes choisies pour l'exploration du graphe

BFS

Un parcours en largeur explique le modèle du noeud papillon.

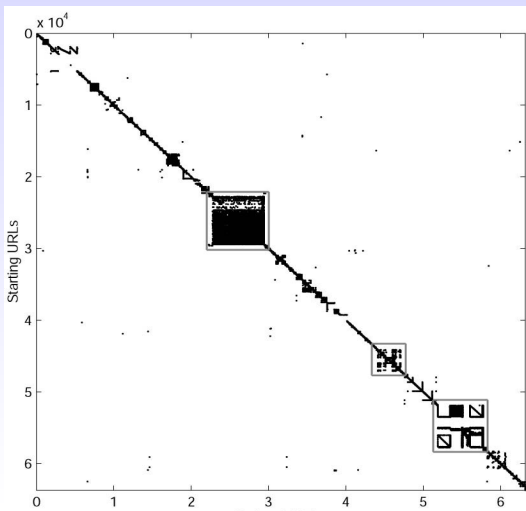
Exploration (Crawl)

L'exploration du graphe du Web est un problème techniquement difficile d'informatique distribuée (des programmes appelés robots suivent les liens)

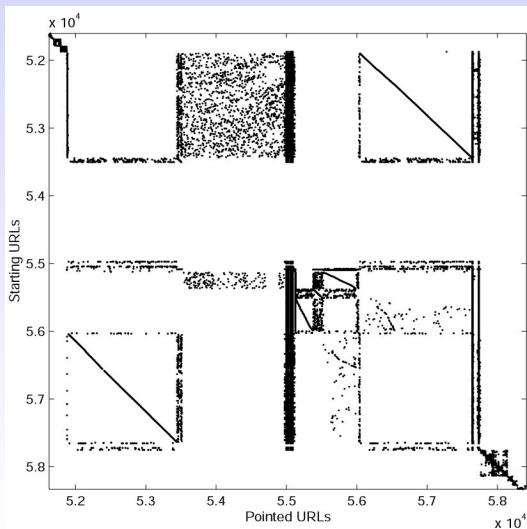
Graphes du Web

P. Boldi et son groupe de recherche propose des données publiques, ainsi qu'un logiciel BV graphs qui permet de compresser les graphes du Web avec 2-3 bits par URL names

Matrice ordonnée par l'ordre alphabétique des noms des URL



Un zoom autour de la diagonale



- ▶ Une page standard contient au plus une centaine de liens vers d'autres pages
Le degré sortant est donc borné
- ▶ C'est donc un graphe peu dense représentable très efficacement
- ▶ Algorithmes parallélisables

Le modèle du radar

- ▶ L'exploration des nouvelles pages se fait en approximativement un mois
- ▶ Périodiquement la base de données opérationnelle est mise à jour et le classement est recalculé sur l'image du graphe du mois précédent
- ▶ Avec des exceptions pour les pages de pub indexées à la demande et les informations quotidiennes
- ▶ **Conséquence : le graphe du WEB n'existe pas !**

Pages cachées

- ▶ Nous ne parlerons ici que du WEB statique
- ▶ Nous ignorerons les pages dynamiques, calculées par un serveur en réponse à une requête d'un usager

Nécessité d'une approche expérimentale

Bien que le WEB soit une construction humaine munie d'une syntaxe (html ...)

personne n'en possède les plans.

Mais il possède une certaine sémantique qu'il s'agit de trouver à la manière des physiciens, sociologues en pratiquant des expérimentations.

Remarques

Le graphe du Web ne sert qu'à calculer le coefficient associé à chaque page,
Il n'est pas utilisé pour la recherche des pages par la suite (le reste dépend de la base de données).

Un modèle linéaire

Une sorte de flot

Soit $R_n(p)$ le coefficient PageRank de la page p à l'étape n du calcul et soit $R_{n+1}(q, p)$ la quantité qui traverse l'arc qp entre les étapes n et $n + 1$.

L'équation

$$R_{n+1}(p) = \sum_{qp} R_{n+1}(q, p)$$

- ▶ Avec l'hypothèse de l'équirépartition du coefficient sur les liens sortant d'une page q

- ▶ On obtient

$$R_{n+1}(q, p) = \frac{1}{\text{degre}(q)} R_n(q)$$

pour tout arc qp sortant de la page q .

- ▶ D'où

$$R_{n+1}(p) = \sum_{qp} \frac{1}{\text{degre}(q)} R_n(q)$$

Vectoriellement

- ▶ $R_{n+1} = A^T R_n$ où A est une sorte de matrice d' incidence du graphe du Web.
- ▶ $A[p, q] = \frac{1}{d+(p)}$ si pq est un arc et 0 sinon
- ▶ Quand la suite R_n converge, sa limite est le vecteur propre associé à la valeur propre 1.

Pourquoi PageRank est-il tant utilisé ?

1. Convergence très rapide
2. Le calcul peut se faire ligne à ligne en utilisant un codage compact du graphe
3. Le calcul se parallélise simplement.
4. Il y a plusieurs interprétations mathématiques intéressantes du calcul

Convergence

A l'aide du théorème de Perron Froebenius

La convergence est assurée si le graphe est fortement connexe et si le pgcd des longueurs des circuits est 1.

Ce qui est impossible à vérifier sur le graphe du Web. Plusieurs astuces sont utilisées pour assurer la convergence du calcul.

Interprétation à l'aide des chaînes de Markov

A est une matrice stochastique et la limite de $R_n(p)$ peut se comprendre comme la probabilité qu'un surfeur aléatoire visite la page p .

Le vecteur R final n'est rien d'autre que la distribution stationnaire d'une marche aléatoire sur le graphe du Web

Un effet de bord ?

M. Bouklit et F. Mathieu ont essayé de modéliser plus avant le comportement d'un surfeur en introduisant par exemple la touche Retour (undo)

Le classement obtenu n'avait pas l'air significativement meilleur.
PageRank est-il un flot de matière ou une probabilité ?

Proposition de projet avec les élèves

Mise au point d'une programmation (ou du fonctionnement à la main) sur des petits exemples de Pagerank (genre séance d'exercices TP ou TD)

Vecteur initial

La question de la forte connexité

Le facteur ZAP (dumping factor)

- ▶ On initialise à $1/N$ le coefficient de PageRank de toutes les pages
où N est le nombre total de pages du graphe du Web
- ▶ $R_{n+1}(p) = \frac{d}{N} + (1 - d) \cdot \sum_{qp} \frac{1}{\text{degre}(q)} R_n(q)$
- ▶ Dans le modèle du surfeur, il peut choisir :
soit de suivre un lien sortant avec un probabilité $1 - d$,
soit "zapper" sur une page aléatoire avec un probabilité d
- ▶ On propose de prendre d le facteur ZAP entre 0.1 et 0.2

Géniale astuce

le graphe devient fortement connexe

Transformation $T : R^n \rightarrow R$, $T(x) = d.\epsilon + (1 - d)Ax$

où ϵ est le vecteur dont toutes les composantes valent $1/N$

Point fixe

Si A est une matrice stochastique alors l'application T est contractante de rapport $1 - d$ et quelle que soit la valeur initiale x_0 , la suite

$x_{n+1} = T(x_n)$ converge vers un unique point fixe μ .

Vitesse de convergence

$$|x_{n+1} - \mu| \leq \frac{1-d}{d} \cdot |x_{n+1} - x_n|$$

$$\text{avec } |y| = \sum_i |y_i|$$

Test d'arrêt

Il suffit de choisir un seuil et de calculer à la fin de chaque itération

$$|x_{n+1} - x_n|$$

En pratique une centaine d'itérations sur les graphes que j'ai testés.

Pour Google $d = 0.15$

Curieusement faire tendre d vers 0, n'améliore pas les résultats

L'extrême robustesse expérimentale de PageRank

Peu dépendant des conditions initiales (cf. Il existe un point fixe unique!)

On peut commencer le calcul avec le Pagerank du mois dernier

En conclusion

Une application intéressante de l'algèbre linéaire
que l'on peut découvrir sur des exemples simples

Question personnelle

Peut-on utiliser ces idées pour des calculs de flots maximum dans un graphe ?

Retour sur le fonctionnement d'un moteur de recherche

1. Préalcul d'un fichier inversé des Pages Web, dans une gigantesque Base de données distribuée
2. Extraction du contenu de la question (i.e. quelques mots clés)
3. Recherche de toutes les pages WEB qui contiennent ces mots clés et calcul d'un score pondéré pour chaque page (le score dépend des mots clés de la question)
4. Filtrer les pages résultats à l'aide d'un profil d'utilisateur (langue, n° IP, académique versus commercial).
5. Trier les pages obtenues à l'aide de PageRank (et de quelques petites astuces secrètes) et afficher cette liste ordonnée d'URL.

Calcul de score

Score pondéré

Construit à partir de :

1. Les mots apparaissent dans le titre de la page (ou le chemin d'accès)
(exemple French Military Victories)
2. Les mots apparaissent α fois dans la description de l'entête de la page.
3. Les mots apparaissent β fois dans la page, avec préférence pour le début de la page.

Un peu de technologie

La base de données est distribuée sur des centaines de milliers de machines (PCs ou serveurs)

Grosse consommation d'énergie électrique

Vu le taux de panne, chaque jour plusieurs centaines de machines sont en panne

Nécessité d'une grande redondance des informations

Comment faire ?

Les bombes de Google

- ▶ Le nom associé à la balise html d'une page q qui pointe sur une page p est utilisé pour indexer la page p
- ▶ C'est presque l'unique moyen de faire passer de la sémantique
- ▶ Mais cela permet de fabriquer des bombes !

Les bombes les plus célèbres

1. Talentless hacker versus Andy Pressman made by Adam Mathes in 2001.
2. Miserable failure versus George W. Bush made by George Johnston 2003
3. Sarkozy versus Iznogood
4. Ministre Blanchisseur versus Renaud Donnadiou de Vabres
5. ...

Dans la plupart des cas le mot de la balise est un bon mot clé pour une page, et une grande partie du succès de Google vient de cela.

First Google answer was : We just show what is between the Web pages.

Une bombe explose uniquement après la mise à jour des coefficient de PageRank (un mois)

Mais après elle résiste au temps, il est difficile de s'en débarrasser !

Existe-t-il une solution algorithmique pour la détection des bombes ?

01/26/2007

Google announced today a modification to their search algorithm that minimizes well-known googlebombing exploits. Searches on "miserable failure" and their ilk no longer bring up political targets. The Google blogger writes : By improving our analysis of the link structure of the web, Google has begun minimizing the impact of many Googlebombs. Now we will typically return commentary, discussions, and articles about the Googlebombs instead.

L'analyse du graphe du Web permet :

- ▶ de calculer un ordre total sur les pages PageRank
- ▶ de calculer d'autres ordres (Annuaire, Autorité, Spam)
- ▶ de rechercher des communautés entre pages fortement reliées
- ▶ d'analyser les logs (traces des visites)

La recherche de mots clés dans les pages permet :

- ▶ d'indexer les pages afin de retrouver l'ensemble des pages contenant un mot clé donné
- ▶ Catégoriser les pages (Science, Religion, Sport ...)

- ▶ Recherche par mots clés avec " " qui permet d'exprimer la proximité des mots
- ▶ Recherche avec des opérateurs logiques : AND, OR ou NOT ... (Possible avec Excite à vérifier)

Nouveau domaine de recherche **Graph Mining**
en français Fouille de Graphes.

Il s'agit d'extraire des connaissances à partir de gigantesques graphes (souvent dynamiques)
Une équipe ATT labs y travaille.

Techniques appliquées sur les Graphes de communications :
téléphone, mail
cela permet de :

- ▶ subscription fraud – new accounts with many fraudulent numbers in their calling circle are suspicious and generate alert
- ▶ targeted (viral) marketing – allows us to find clusters of customers who have high probability of taking a given product offer.
- ▶ repetitive debtors - delinquent customers who try and set up a new account are identified by their calling patterns and the new account can be shut down.

Le modèle économique

1. Accès gratuit pour un individu, payant pour une société
2. Affichage de bandeaux publicitaires
3. Liens publicitaires (paiement au clic)
4. Facturation de logs, et analyse de traces diverses (séquence de requêtes) et statistiques
5. Indexation immédiate de pages à la demande
6. Non indexation immédiate de pages à la demande ! Si on veut faire disparaître une page compromettante
7. Moins correct
Affichage dans les résultats non-publicitaires dans la première page de résultats moyennant finance (paiement au clic, i.e. on paye pour 2000 clics) pour certaines requêtes.

Le modèle économique de Google est un peu différent, vu sa situation de monopole.

Commutativité

La commutativité des mots clés dans une requête ?

Exemples les réponses ne sont pas classées dans le même ordre si l'on pose les questions :

Nicolas Sarkozy

Sarkozy Nicolas

ou encore Nicolas Sarkozi

ou Sarko

Ni Altavista, ni Exalead, ni Google, ni Ask ne sont commutatifs ! ¹
Comment l'expliquer ? Par une catégorisation des noms : prénom
versus nom de famille ?

¹Il existe plusieurs milliers de moteurs de recherche, je n'ai pas tout essayé

Trouver des questions ayant peu de réponses différentes

- ▶ Jeu Google : trouver une question en deux mots ayant ≤ 1 réponse.
(si possible sans guillemets dans la question)
- ▶ dorade droitière
poulpe ambitieuse ...
- ▶ Goolglewhack

- ▶ Les limites des statistiques de mots clés (Rabelais et Dieu)
- ▶ La page recherchée ne contient pas nécessairement le mot de la requête (Page de Harvard, du MIT le mot est remplacé par un logo) ou les pages personnelles.
- ▶ Catégorisation bayésienne (donc heuristique car probabiliste)

Mot clés spéciaux par exemple :

Confidential do not distribute

permet de vérifier la stratégie de publication d'une société.

Une typologie des requêtes

1. Savoir, connaissance : recherche d'information (48%)
2. Localisation : navigation (adresses, cartes, ...) (25%)
3. Achat en ligne (25 %)

Faut-il des moteurs de recherche spécialisés ?

Par exemple : Google Scholar pour le monde académique qui n'indexe que des textes.

A défaut une catégorisation des requêtes en fonction :

- ▶ du pays, de la langue
- ▶ des profils utilisateur
- ▶ de la requête elle-même, ce qui expliquerait l'absence de commutativité

Il faudrait utiliser :

Un moteur de recherche pour expert qui permette le paramétrage de la requête :

- ▶ sur la position des mots clés dans la page
- ▶ un paramétrage des occurrences du mot clé

Problèmes de recherche actuels

- ▶ Maintenir les performances et la pertinence des réponses
- ▶ lutter algorithmiquement contre les spams
- ▶ identifier des communautés, des comportements
- ▶ Indexer les images puis les vidéos algorithmiquement

Quelques références

- ▶ T. Bennouas, PhD Thesis, Montpellier University, 2005.
- ▶ M. Bouklit, PhD Thesis, Montpellier University, 2006.
- ▶ S. Brin, L. Page, R. Motwani, T. Winograd, The PageRank citation ranking : bringing an order to the Web, Technical Report 1999-0120, Computer Science Dept. Standford, 1999.
- ▶ M. Eisermann, Comment fonctionne Google ?, www-fourier-ujf-grenoble.fr/~eiserm
- ▶ J. Kleinberg, Authoritative sources in a hyperlinked environment, J. of the ACM, 1999.
- ▶ A.N. Langville, C.D. Meyer, Google's PageRank and beyond, Princeton University Press, 2006.
- ▶ F. Mathieu, PhD Thesis, Montpellier University, 2004.

Merci de votre attention !!