Algorithmes

2e édition

Massih-Reza Amini - Éric Gaussier

Préface de Stephen Robertson

Recherche d'information

Applications, modèles et algorithmes

Data mining, décisionnel et big data

EYROLLES

2e édition

Recherche d'information

***Massih-Reza Amini**, professeur d'informatique à l'université Grenoble Alpes, est titulaire d'une thèse en Informatique de l'université Paris 6. Ses recherches portent sur l'apprentissage automatique appliqué aux problèmes d'accès à l'information à large échelle. Il est co-auteur de nombreux articles scientifiques notamment de l'ouvrage* Apprentissage machine *paru aux éditions Eyrolles. Il dirige actuellement l'équipe AMA dont les recherches se situent en analyse de données, modélisation et apprentissage automatique.*

***Éric Gaussier**, professeur d'informatique à l'université Grenoble Alpes, est titulaire d'un diplôme en Mathématiques Appliquées de l'école Centrale Paris et d'une thèse en Informatique de l'université Paris 7. Ses travaux de recherche s'inscrivent dans la Science des données, au carrefour de l'apprentissage statistique, de la recherche d'information et du traitement automatique des langues. Il est co-auteur de nombreux articles et brevets dans ces domaines et dirige actuellement le laboratoire d'Informatique de Grenoble.*

Le premier ouvrage francophone sur les algorithmes qui sous-tendent les technologies de big data et les moteurs de recherche !

Depuis quelques années, de nouveaux modèles et algorithmes sont mis au point pour traiter des données de plus en plus volumineuses et diverses. Cet ouvrage présente les fondements scientifiques des tâches les plus répandues en recherche d'information (RI), tâches également liées au data mining, au décisionnel et plus généralement à l'exploitation du big data.

La deuxième édition de cet ouvrage propose un exposé détaillé et cohérent des algorithmes classiques développés dans ce domaine, abordable par des lecteurs qui cherchent à connaître le mécanisme des outils quotidiens d'Internet. De plus, le lecteur approfondira les concepts d'indexation, de compression, de recherche sur le Web, de classification et de catégorisation, et pourra prolonger cette étude avec les exercices corrigés proposés en fin de chapitre.

Ce livre s'adresse tant aux chercheurs et ingénieurs qui travaillent dans le domaine de l'accès à l'information et employés de PME qui utilisent en profondeur les outils du webmarketing, qu'aux étudiants de Licence, Master, écoles d'ingénieurs ou doctorants qui souhaitent un ouvrage de référence sur la recherche d'information.

Sommaire

www.editions-eyrolles.com

Massih-Reza Amini - Éric Gaussier

Préface de Stephen Robertson

Recherche d'information

Applications, modèles et algorithmes

2e édition

EYROLLES

ÉDITIONS EYROLLES
61, bd Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com

Préface

La recherche d'information, autrefois vue comme un domaine de spécialité à l'intersection des techniques documentaires et de la science informatique, est devenue l'une des technologies majeures du XXIe siècle. Chacun s'attend en effet aujourd'hui à pouvoir trouver en quelques secondes des informations diverses sur tout type de sujet : horaires des transports en commun, principes de la production d'électricité, nature des maladies infectieuses, pharmacie la plus proche fournissant des antalgiques, films à l'affiche du cinéma voisin, analyse critique des œuvres d'Erik Satie, fondements de l'existentialisme de Jean-Paul Sartre ou tout détail trivial de la vie courante. Chacun considère cela comme allant de soi et cet « allant de soi » est né du développement des moteurs de recherche sur le Web.

Les fondements technologiques des moteurs de recherche peuvent être décrits très simplement, même si de nombreuses connaissances, combinaisons de développements théoriques et de savoir-faire expérimentaux, ont été accumulées dans ce domaine. Créer un moteur de recherche médiocre est facile ; en créer un qui soit à la fois pertinent et rapide est une tout autre histoire, et cela quelle que soit la taille de la collection considérée (collection personnelle de courriers électroniques ou intégralité du corpus de la Bibliothèque nationale). De façon étonnante, ce sont les moteurs de recherche sur Internet qui ont tenu le haut du pavé ces quelque vingt dernières années. Pour toutes sortes de raisons, ils ont atteint un niveau de maturité qui semble bien en avance sur ce qui se pratique à des échelles plus réduites.

Cet ouvrage est une introduction fondamentale à la technologie de la recherche d'information et ses applications, pour la plupart liées au Web. Il combine traitement automatique des langues et modèles théoriques. Outre l'ordonnancement de documents en réponse à une requête, il couvre la classification supervisée (en catégories prédéfinies) et non supervisée (clustering). L'importance des concepts statistiques dans ce domaine est centrale, depuis les caractéristiques statistiques des langues (loi de Zipf) jusqu'aux modèles probabilistes de recherche d'information et aux modèles à thèmes latents. Cet ouvrage était nécessaire pour mettre à la portée d'un plus large public les fondamentaux de cette technologie moderne incontournable qu'est la recherche d'information.

Stephen Robertson
septembre 2012

Table des matières

Notations

$\mathcal{C} = \{d_1, ..., d_N\}$	Collection contenant N documents
$\mathcal{V} = \{t_1, ..., t_V\}$	Vocabulaire constitué de V termes
M	Nombre total de mots dans $\mathcal{C}$
M_y	Nombre total de types de mots dans $\mathcal{C}$
M_{Nor}	Nombre de types de mots après racinisation
$\mathbf{X}_{V \times N}$	Matrice termes-documents
$\text{tf}_{t,d}$	Nombre d'occurrences du terme t dans $d \in \mathcal{C}$
$\text{ntf}_{t,d}$	Nombre d'occurrences normalisé du terme t dans le document d
$\text{tf}_{t,\mathcal{C}}$	Nombre d'occurrences du terme t dans la collection $\mathcal{C}$
$\text{tf}_{t,q}$	Nombre d'occurrences du terme t dans la requête q
$\|d\|$	Nombre de termes différents dans d
l_d	Nombre total d'occurrences des termes du vocabulaire dans d ; $l_d = \sum_{t \in \mathcal{V}} \text{tf}_{t,d}$
$l_\mathcal{C}$	Nombre total d'occurrences des termes du vocabulaire dans $\mathcal{C}$; $l_\mathcal{C} = \sum_{d \in \mathcal{C}} \sum_{t \in \mathcal{V}} \text{tf}_{t,d}$
df_t	Nombre de documents de la collection contenant le terme t ($\text{idf}_t = \log \frac{N}{\text{df}_t}$)
$\mathbf{d} = (w_{id})_{i \in \{1,...,V\}}$	Représentation vectorielle du document d
w_{id}	Poids du terme d'indice i du vocabulaire dans d

$s(q,d)$	Score de similarité entre une requête q et un document d (ce score est parfois noté RSV(q,d), où RSV représente la *Retrieval Status Value*)
$\mathbf{I}$	Matrice identité
$R_{d,q}$	Jugement de pertinence binaire assigné au document d pour la requête q
$\mathcal{A}_q$	Ensemble des documents (ou *alternatifs*) de $\mathcal{C}$ concernés par la requête q
$P(.)$	Une distribution de probabilité
$P(t \mid d)$	Probabilité de présence du terme t dans $d \in \mathcal{C}$
$P(t \mid \mathcal{C})$	Probabilité de présence du terme t dans la collection
$\mathcal{Y} = \{1, \cdots, K\}$	Ensemble de K étiquettes de classes, apprises ou données
$\mathcal{Z} = \{z_1, \cdots, z_K\}$	Ensemble de K thèmes latents
$\mathrm{df}_t(k)$	Nombre de documents de la classe d'étiquette k contenant le terme t
$S = ((\mathbf{d}_j, c_j))_{i=1}^m$	Ensemble d'apprentissage contenant m documents avec leur étiquette de classe
S_k	Ensemble des documents de l'ensemble S appartenant à la classe d'étiquette k
$N_k(S)$	Nombre de documents de l'ensemble S appartenant à la classe d'étiquette k
$\mathcal{D}$	Distribution de probabilité suivant laquelle les exemples d'apprentissage sont identiquement et indépendamment générés
$\mathcal{F}$	Une classe de fonctions $\mathcal{F} = \{f : \mathbb{R}^V \to \mathcal{Y}\}$
$\langle ., . \rangle$	Produit scalaire
$\hat{R}_m(f, S)$	L'erreur empirique de la fonction de prédiction f sur l'ensemble d'apprentissage S de taille m
$R(f)$	L'erreur de généralisation de la fonction de prédiction f
$\{G_1, \ldots, G_k\}$	Ensemble de K classes obtenues sur une collection de documents $\mathcal{C}$
$\{\mathrm{r}_1, \ldots, \mathrm{r}_K\}$	K vecteurs représentant les centres de gravité des classes ; $\forall k, \mathrm{r}_k = \frac{1}{\lvert G_k \rvert} \sum_{\mathbf{d} \in G_k} \mathbf{d}$
IMP	Information mutuelle ponctuelle
$\mathrm{IM}(t, c)$	Information mutuelle du terme t dans la classe c
$[\![\pi]\!]$	Fonction indicatrice, égale à 1 si le prédicat π est vrai et 0 sinon
$\mathbb{E}(X)$	Espérance mathématique de la variable aléatoire X

Liste des tableaux

Liste des figures

Liste des algorithmes

Chapitre 1
Introduction

L'idée d'utiliser les ordinateurs pour traiter et trouver l'information enfouie dans une collection de documents et correspondant à une demande précise remonte à la fin de la Seconde Guerre mondiale. À cette époque, les militaires américains cherchaient un moyen pour indexer l'ensemble des documents scientifiques allemands afin d'y déceler toutes les informations et données pertinentes pour leurs recherches. Après la fin de la guerre, le domaine de la recherche d'information (`RI`*) s'est développé grâce notamment à l'activité des libraires, des cabinets de juristes et d'autres professionnels travaillant sur des collections de documents spécialisées. Dès cette époque, la définition académique donnée à la tâche de* `RI` *était : la recherche de données non structurées, généralement des documents, satisfaisant une demande d'information spécifique. Par données non structurées, on entendait les données qui ne possèdent pas de structure interne précise ou qui sont sémantiquement ambigües.*

Cette définition différencie bien cette forme de recherche de la recherche de données *relationnelles* ou *structurées*, stockées dans des bases de données telles que celles utilisées par les sociétés pour maintenir l'inventaire de leurs produits ou les dossiers de leur personnel. Dans ce cas, les données relationnelles ont une structure et une sémantique bien claires. La recherche dans ces bases s'effectue aussi en retournant les données qui satisfont exactement une expression d'algèbre relationnelle. De ce fait, dans les systèmes de gestion de bases de données, aucune erreur n'est tolérée. En revanche, pour les systèmes de recherche d'information, le but est plutôt d'ordonner les données suivant une mesure de similarité par rapport à la requête qui généralement est un ensemble de mots-clés, ne respectant aucune expression régulière ni algébrique. Le souci est alors de disposer des mesures de similarité qui feront que les données les plus pertinentes par rapport à la requête vont se trouver en tête de la liste retournée, au-dessus des données moins pertinentes ou non pertinentes. Au final, le point central en RI est de construire un

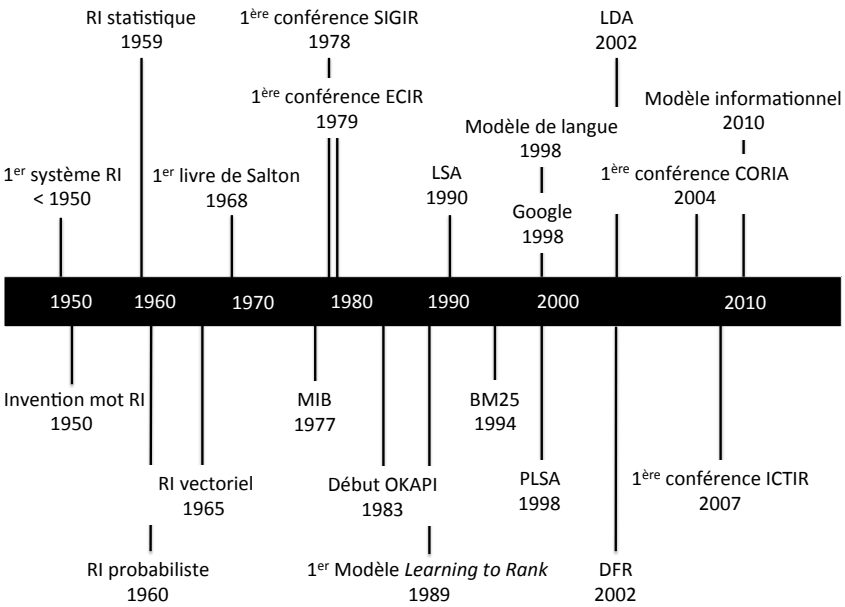

Figure 1.1 - Les faits marquants de l'évolution de la RI, relatés dans cet ouvrage, à partir du premier système créé pendant la Seconde Guerre mondiale jusqu'en 2010. *SIGIR* est la conférence par excellence du domaine. *ECIR* et *CORIA* sont les conférences européenne et francophone en RI. *ICTIR* est une conférence internationale sur la théorie de la RI.

système qui respectera au mieux l'ordre qu'il faut attribuer aux données, du plus pertinent au moins pertinent par rapport à la requête de base.

Jusqu'au début des années 1990, le domaine de la RI est resté relativement cloisonné au monde professionnel. Plusieurs études avaient d'ailleurs montré à cette époque que le grand public préférait s'informer auprès d'autres personnes plutôt que d'utiliser des systèmes automatiques de RI. Cette tendance s'est néanmoins inversée (et de façon irréversible) avec l'arrivée d'Internet et l'utilisation massive des moteurs de recherche. La Toile est maintenant un entrepôt universel qui a permis un partage sans précédent d'informations de toutes sortes à travers le globe. Ce développement a mis l'emphase sur l'utilisation des techniques émergentes issues de la RI pour accélérer et rendre plus efficace la fouille à large échelle sur la Toile. De nos jours, la recherche d'information est très diversifiée et couvre maintenant d'autres problématiques liées à la fouille de données dans les grandes collections. Outre ses propres outils, la RI s'appuie aussi sur un ensemble de techniques issues d'autres domaines comme la statistique, l'apprentissage automatique ou le traitement automatique du langage naturel. La figure 1.1 montre les faits marquants de l'évolution de la RI, à partir du premier système créé pendant la Seconde Guerre jusqu'en 2010. Nous relatons une partie de cette évolution dans ce livre.

1.1 Concepts étudiés dans ce livre

Cet ouvrage présente les fondements scientifiques des tâches les plus répandues en RI à un niveau accessible aux étudiants de master et aux élèves ingénieurs. Notre souci principal a été de proposer un exposé cohérent des algorithmes classiques développés dans ce domaine, à destination des lecteurs qui cherchent à connaître le mécanisme des outils d'Internet qu'ils emploient tous les jours. Cette étude ne se limite pas à l'application initiale de RI et aborde aussi les problèmes connexes dans lesquels de nombreuses avancées techniques ont été réalisées ces dernières années. Nous allons nous intéresser plus particulièrement aux concepts décrits dans les paragraphes suivants.

Indexation, représentation et compression

Les constructions du *dictionnaire* et de *l'index inversé*, ainsi que la représentation vectorielle des documents, constituent le point de départ dans toute manipulation ou recherche en RI. Dans une collection donnée, construire le dictionnaire ou le vocabulaire correspond à extraire une liste de termes utiles, caractéristiques des documents présents dans la collection. Cette liste servira à représenter les

documents dans un espace vectoriel. L'autre concept fondamental en RI est la constitution de l'index inversé. Il s'agit ici de construire, pour chaque terme du dictionnaire, la liste des index de documents contenant ce terme. Cette liste, aussi appelée *liste inversée*, rend l'appariement entre les requêtes et les documents de la collection plus efficace. Pour les très grandes collections de données, un problème majeur est le stockage de l'index et du dictionnaire dans la mémoire ou sur le disque. Le défi dans ces cas est de trouver un moyen de compression simple et rapide des données.

Recherche d'information

Cette tâche constitue le cœur de cet ouvrage. Un système de recherche transcrit un besoin d'information donné sous la forme d'une requête constituée de mots-clés. Lorsque l'utilisateur examine le résultat de la recherche, il voit les documents triés par ordre décroissant de pertinence. Si la requête est une expression booléenne, l'utilisation de l'index inversé permet de trouver facilement et en un temps minimal tous les documents qui satisfont cette requête. En revanche, les systèmes booléens purs ne permettent pas de retrouver les documents similaires au besoin d'information de l'utilisateur et ne contenant pas exactement les termes de la requête. Plusieurs modèles ont été développés pour pallier ce problème, depuis les modèles vectoriels jusqu'aux modèles probabilistes. De même, plusieurs stratégies, qui consistent à étendre la requête afin d'y inclure des termes similaires mais non mentionnés originellement par l'utilisateur, ont vu le jour afin d'enrichir ces différents modèles.

Recherche sur le Web

La Toile (ou le Web) est un entrepôt dynamique et distribué de documents qui, par sa taille, par le manque de supervision dans la génération et la suppression de documents, ainsi que par la diversité du type de ces derniers, rend la recherche bien plus difficile que la recherche traditionnelle effectuée sur des collections classiques. Les premiers moteurs de recherche sur la Toile reproduisaient néanmoins directement les méthodes de RI classiques, le défi principal étant de gérer des index inversés de très grandes tailles. La prise en compte, vers la fin des années 1990, d'une des caractéristiques essentielles du Web, à savoir les liens hypertextes reliant les documents entre eux, a permis, d'une part, de réaliser une meilleure indexation des pages Web et, d'autre part, de donner un score de notoriété à chaque page sur la base de la topologie de la Toile. Cela a conduit à la première

génération des moteurs de recherche vraiment adaptés au Web, dont Google fut le prototype. De nos jours, d'autres éléments sont pris en compte et les modèles utilisés reposent sur des techniques récentes d'apprentissage automatique.

Classification de documents

Un système de classification de documents a pour but de catégoriser automatiquement une collection de documents suivant un ensemble de classes prédéfinies. Un exemple de tels systèmes est le catégoriseur de messages électroniques incorporé dans la plupart des boîtes e-mail et qui place les courriers suspects automatiquement dans le dossier des indésirables. Les systèmes de classification sont généralement conçus avec des techniques issues de *l'apprentissage statistique* et opèrent en deux phases. La première est la phase d'entraînement, lors de laquelle les paramètres du système sont réglés sur une base d'apprentissage contenant des documents avec leurs classes respectives. Le système apprend l'association entre les documents et leurs classes. C'est lors de la seconde phase, dite de *test*, que le système assigne une classe à chaque nouveau document entrant. Habituellement, les paramètres des systèmes d'apprentissage sont mis à jour périodiquement pendant le laps de temps où il n'y a pas de traitement à effectuer sur des documents arrivants.

Partitionnement de documents

La tâche de partitionnement de documents (ou *document clustering* en anglais) est un autre problème important en RI. Le but est ici de réunir dans le même groupe les documents qui sont similaires par rapport à un critère donné. En pratique, les résultats de partitionnement indiquent non seulement la structure d'une collection, mais ils sont aussi souvent utilisés dans d'autres tâches de RI comme la navigation ou la recherche distribuée. La tâche de partitionnement est depuis longtemps un sujet de recherches intensives dans différents domaines, comme la *statistique*, l'*apprentissage machine*, l'*analyse de données* et la RI. Grâce à la capacité croissante des ordinateurs à traiter de grandes quantités de données dans des laps de temps réduits, ce paradigme a connu un développement rapide ces dernières années, aussi bien d'un point de vue théorique que pratique. On distingue trois types d'approches en partitionnement : les méthodes à base de similarité qui partitionnent les données en k groupes, où chaque groupe optimise un critère de partitionnement fondé sur une mesure de similarité ; les approches à base de projection qui se fondent sur la recherche des vecteurs propres d'une matrice de similarité

entre documents pour projeter puis regrouper les documents dans un espace de dimension réduite ; et, finalement, les méthodes à base de densité qui modélisent la distribution de probabilité des différents groupes avant d'assigner les documents aux différentes partitions. Nous présenterons en détail les méthodes à base de similarité, qui comptent parmi les plus utilisées.

Réseaux de neurones profonds

Un réseau de neurones est un ensemble d'unités élémentaires, appelées neurones artificiels, reliées entre elles et permettant d'effectuer différentes transformations non linéaires. Depuis 2010, nous sommes témoins d'une percée importante de l'application de ces réseaux de neurones profonds dans plusieurs domaines phares en informatique comme la vision par ordinateur ou le traitement automatique du langage naturel, et il est attendu que l'apprentissage profond ait aussi un grand impact dans le domaine de la recherche d'information.

Recherche de thèmes latents

Depuis le début des années 1990, on assiste à une recherche intensive sur l'extraction de *thèmes cachés* ou *latents* dans des collections de documents ou d'images. Ces études sont motivées par une meilleure prise en compte du discours dans la représentation des documents ou des images. L'approche sac de mots, très populaire et répandue en RI, consiste à caractériser les documents avec les termes qui les constituent, sans prendre en compte l'ordre ou le sens de ces termes. Ainsi, les documents qui expriment les mêmes concepts mais avec des termes différents (phénomène de synonymie) auront des représentations très différentes et ne seront pas tous considérés comme pertinents pour une requête fondée sur ces concepts. De façon similaire, les termes ayant des sens différents dans différents contextes (phénomène de polysémie) sont en général traités comme équivalents et rapprochent des documents qui ne devraient pas l'être. Les modèles latents ont été conçus pour pallier ces problèmes en exhibant les thématiques sous-jacentes et le vocabulaire qui leur est associé. Ces techniques visent ainsi à mieux représenter les documents textuels ou les images.

1.2 Organisation du livre

Les chapitres de ce livre décrivent les concepts présentés précédemment. À la fin de chaque chapitre, nous donnons aussi un ensemble d'exercices (corrigés) relatifs aux sujets développés. L'enchaînement des idées est le suivant :

- Dans le **chapitre 2**, nous décrivons la chaîne complète d'indexation qui nous permet de construire l'ensemble du vocabulaire en partant d'une collection de documents donnée. Nous décrivons aussi les algorithmes les plus répandus pour la construction de l'index inversé dans les collections *statiques*, *distribuées* et *dynamiques*. Dans la partie exercice, nous étudions, entre autres, un ensemble de techniques permettant de comprimer le vocabulaire et l'index inversé afin que, pour des collections de très grandes tailles, ces éléments puissent être stockés en mémoire et accessibles en temps raisonnable.

- C'est dans le **chapitre 3** que nous nous intéressons aux modèles de recherche d'information les plus courants. Nous présentons les avantages et les inconvénients de chacun, en mettant l'accent sur les éléments qui ont motivé leur évolution, depuis les modèles booléens et vectoriels jusqu'aux modèles probabilistes récents (de langue et fondés sur l'information). Beaucoup d'entre eux se retrouvent dans d'autres tâches comme la traduction automatique ou la reconnaissance de la parole.

- Le **chapitre 4** décrit les modèles et stratégies développés pour la recherche sur la Toile. Nous présentons d'abord les techniques usuelles pour collecter et indexer les documents entreposés sur différents serveurs connectés à la Toile, avant d'exposer les éléments spécifiques aux moteurs de recherche sur le Web : calcul d'un score de notoriété pour chaque page et déploiement de méthodes d'apprentissage d'ordonnancement à partir des données de clics des utilisateurs. Ce chapitre se termine avec la présentation d'une technique de calcul approché de similarités entre documents au sein de grandes collections à l'aide des tables de hachage.

- Le **chapitre 5** présente d'abord les techniques de sélection de variables utilisées en RI pour réduire la taille du vocabulaire et donc la dimension de l'espace de représentation. Nous nous intéressons ensuite aux premières techniques, qualifiées de *génératives*, introduites en RI pour la catégorisation de documents. Ces techniques visent tout d'abord à modéliser les distributions de probabilité générant les documents avant d'utiliser ces distributions pour prendre une décision sur les classes des documents. Dans la dernière partie de ce chapitre,

nous présentons les modèles *discriminants* issus principalement du domaine de l'apprentissage machine pour cette tâche.

- Nous exposons les deux approches les plus utilisées pour le partitionnement de documents dans le **chapitre 6**, à savoir les méthodes de réallocation et les méthodes de partitionnement agglomératif. Comme nous l'avons mentionné plus haut, le partitionnement a des applications dans différents domaines. Certaines d'entre elles sont également passées en revue dans ce chapitre.
- Le **chapitre 7** présente les modèles formels des réseaux de neurones avec une emphase sur la retro-propagation, qui est l'algorithme le plus connu pour apprendre les paramètres d'un réseau profond. Ce chapitre se termine par la présentation de deux applications des réseaux de neurones très utilisées en RI, à savoir la réduction de dimension et la représentation vectorielle des mots.
- C'est dans le **chapitre 8** que nous décrivons les modèles latents les plus utilisés pour l'extraction de thèmes dans les images et les documents. Nous nous intéressons tout d'abord aux premières tentatives réelles de modélisation des phénomènes de synonymie et polysémie à travers la décomposition en valeurs singulières, avant d'aborder les versions plus récentes qui étendent le cadre de base. Ces différentes versions seront illustrées aux travers des résultats qu'elles fournissent sur des collections de textes standards.
- Nous consacrons finalement quelques pages en fin d'ouvrage aux logiciels libres de recherche d'information, de catégorisation et de partitionnement.

Enfin, nous souhaitons qu'au-delà de sa vocation pédagogique, ce livre, qui s'adresse non seulement aux étudiants de master et aux élèves d'école d'ingénieurs, mais aussi aux doctorants en RI et aux ingénieurs de ce domaine, puisse donner à ses lecteurs le désir d'en savoir plus.

Chapitre 2
Représentation et indexation

Dans ce chapitre, nous allons nous intéresser aux mécanismes amenant à la représentation documentaire et à l'indexation et qui constituent les briques de base de presque tous les algorithmes classiques en accès à l'information. Ces mécanismes comprennent un ensemble de prétraitements permettant la constitution du vocabulaire à partir d'une collection de documents, la représentation des documents dans l'espace des termes, la construction de l'index inversé des termes ainsi que la compression du vocabulaire et de l'index qui favorise l'accélération des traitements et l'accès aux données. La figure 2.1 illustre ces différents mécanismes.

Dans la première partie de ce chapitre, nous allons passer en revue l'ensemble des prétraitements qui aboutissent à la création du vocabulaire (section 2.1). Nous présenterons ensuite les deux lois de base en accès à l'information décrivant l'évolution de la taille du vocabulaire par rapport à la taille de la collection de départ (loi de Heaps) et la distribution des termes par rapport à leur fréquence d'apparition dans une collection donnée (loi de Zipf). Dans les sections 2.3 et 2.4, nous présenterons successivement le modèle vectoriel, qui est communément utilisé pour la représentation documentaire, et les algorithmes classiques pour la constitution de l'index inversé dans des environnements statiques et dynamiques. La dernière section sera consacrée aux exercices concernant les différentes notions présentées dans ce chapitre. Nous allons aussi y traiter en particulier deux algorithmes de compression du vocabulaire et de l'index. La compression est dans ces cas sans perte de données et a pour but de stocker ces dernières de façon optimale ou d'accélérer leur transfert du disque d'une machine à sa mémoire vive. L'étude de ces algorithmes a son importance puisque les systèmes de recherche classiques parcourent plus rapidement les index compressés que leur version non compressées, et ceci parce que le transfert des parties compressées de l'index du disque et leur décompression dans la mémoire vive s'effectuent plus rapidement que le transfert direct des parties non compressées. En exemple, nous détaillerons l'impact des différents algorithmes de traitement et de compression présentés dans ce chapitre sur la collection du Wikipédia français constituée de 1 349 539 documents textuels français de plus de 20 mots visibles sur le site à la date du 27 mai 2011 [1].

2.1 Prétraitements linguistiques

Dans cette partie, nous allons présenter l'ensemble des prétraitements linguistiques conduisant à la constitution de l'index inversé et de la représentation documentaire à partir d'une collection de documents donnée.

Ces prétraitements comprennent la *segmentation* des séquences de caractères présents dans les documents en mots distincts, la *normalisation* des mots, opération qui consiste à fournir une *forme canonique* pour chaque mot, et le *filtrage* qui vise

1. Voir `http://dumps.wikimedia.org/frwiki/latest/frwiki-latest-pages-articles.xml.bz2-rss.xml` pour les derniers transferts disponibles.

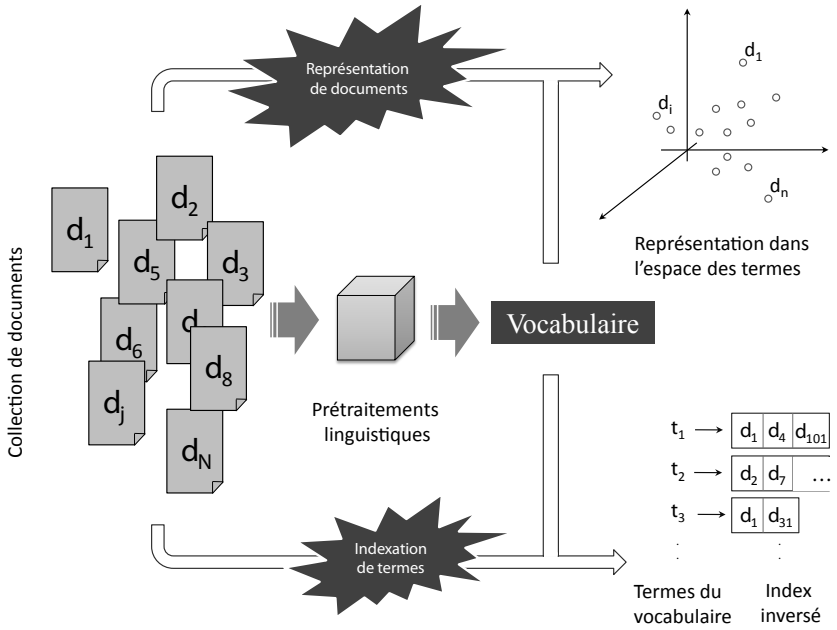

Figure 2.1 - Constitution du vocabulaire, index inversé des termes et représentation des documents dans l'espace des termes pour une collection de documents donnée.

à supprimer les mots les plus fréquents [2]. Ces processus sont schématisés par le cube grisé dans la figure 2.1.

2.1.1 Segmentation

La segmentation (en anglais *tokenisation*) consiste à séparer une suite de caractères en éléments sémantiques, ou mots. Un *type* de mot est la classe de tous les mots ayant la même séquence de caractères et un *terme* est un type de mots que l'on garde pour former le vocabulaire.

Dans l'exemple suivant :

Au sens étymologique, l'information est ce qui donne une forme à l'esprit.

nous avons 14 mots :

Au, sens, étymologique, l', information, est, ce, qui, donne, une, forme, à, l', esprit

2. Nous réservons le nom *terme* pour désigner les mots normalisés restant après l'étape du filtrage.

mais seulement 13 types, puisqu'il y a deux instances de {*l'*}. Après filtrage par un antidictionnaire (section 2.1.3), il ne restera plus que 6 termes {*sens, étymologique, information, donne, forme, esprit*} dans le vocabulaire.

Cette étape est difficile et cruciale puisqu'une mauvaise segmentation a un impact négatif direct sur le résultat de la recherche. En effet, si dans une collection certains termes ne sont pas indexés, les requêtes composées de ces termes ne pourront jamais être appariées avec les documents de la collection qui les contiennent.

Une bonne segmentation dépend de la prise en compte des spécificités de la langue des textes traités. Pour certaines langues asiatiques, comme le chinois, les mots dans un texte ne sont pas séparés par des espaces et la segmentation est dans ce cas une tâche ardue qui constitue un défi scientifique majeur. Pour les langues indo-européennes, la tâche de segmentation est plus aisée puisque l'espace et les signes de ponctuation donnent une première indication de séparation entre les différents éléments lexicaux. Néanmoins, chaque langue de ce groupe linguistique a sa spécificité propre et une simple segmentation par des espaces et des signes de ponctuation conduit généralement à des résultats d'indexation médiocres. Par exemple, pour le français, nous avons :

- les composés lexicaux à trait d'union comme *chassé-croisé*, *peut-être*, *rendez-vous*, etc.
- les composés lexicaux à apostrophe comme *jusqu'où*, *aujourd'hui*, *prud'homme*, etc.
- les expressions idiomatiques comme *au fait*, *poser un lapin*, *tomber dans les pommes*, etc.
- les formes contractées comme *Gad'zarts (les gars des Arts et Métiers)*, *M'sieur*, *j'*, etc.
- les sigles et les acronymes comme *K7, A.R., CV, càd, P.-V.*, etc.

Dans ce cas, lorsque nous indexons un texte en français, nous ne souhaitons pas, par exemple, que le mot *aujourd'hui* soit séparé en *aujourd* et *hui* alors que nous voulons que les mots *un homme* et *l'homme* soient indexés sous le même terme, *homme*. Ce problème devient même extrême avec l'allemand, où les noms composés s'écrivent sans espace ; par exemple le mot *Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz* doit, *a* priori, être séparé en les dix mots qui le constituent - Rind (*boeuf*) + Fleisch (*viande*) + Etikettierung (*étiquetage*) + Überwachung (*surveillance*) + Aufgabe (*tâche*) + Übertragung (*transmission*) + Gesetz (*droit*) - avant d'être indexé. Pour les langues européennes, différents logiciels de segmentation commerciaux existent. Le but de certains de ces logiciels est de réaliser une analyse plus poussée du texte en associant aux mots d'une phrase leur fonction grammaticale. La segmentation dans ces cas est une étape préliminaire à cette analyse.