

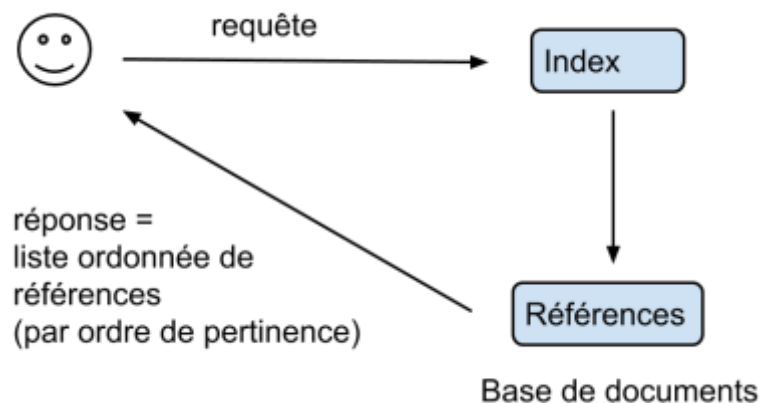
# Indexation des documents texte

## 1. Données texte

- Fouille de textes : "Text mining".
- Bases de documents : données de type texte ("String").
- Algorithmes de recherche sur des documents textes.

### 1.1 Bases documentaires

- $B$  : base documentaire:
  - Librairie, bibliothèque, centre de documentation, ...
  - Dossier contenant des documents (fichiers textes, documents formatés, pages web, forums...)
  - Serveur de fichiers (serveur web, fournisseur de contenus, ...)
  - Moteur de recherche
- $n$  : nombre de documents :  $n = |B|$ .
- $d \in B$  : un document appartenant à la base  $B$ .
- $t \in d$  : un terme présent dans le document  $d$ .



Recherche documentaire :

### 1.2 Ordres de grandeur

- Centre de documentation :  $n \in [10^2 - 10^4]$
- Fournisseur de contenus (Amazon) :  $n \in [10^4 - 10^8]$
- Moteur de recherche (Google) :  $n \in [10^8 - 10^{16}]$

-> "Big data"

### 1.3 Enjeux

- Temps de réponse (< 1s ?)
- Stockage des données
- Protection des données

- Vie privée

## 1.4 Méthodes

- Théorie de l'information
- Apprentissage automatique ("Machine Learning")
- Reconnaissance de formes
- Statistiques

## 1.5 Exemples de moteurs de recherche

- [Elastic Search](#)
- [Solr](#)

## 2. Algorithmes sur les textes

[Données texte](#)

[Recherche/remplacement](#)

[Expressions régulières](#)

[Appariement, similarité, distance](#)

[Arbres d'expressions](#)

## 3. Principes généraux : double indexation

### Fichier inverse - BitMap

On appelle fichier inverse la structure de données qui, à tout terme  $t$ , associe l'ensemble  $D(t)$  des documents contenant le terme  $t$ .

La structure de données implémentant les fichiers inverse est un tableau bidimensionnel binaire  $T$  (appelé "bitmap")

- A chaque document  $d$  est attribué un entier  $i \in \{1..n\}$
- A chaque terme  $t$  est attribué un entier  $j \in \{1..m\}$
- $T[i,j] = 1 \iff j \in D(i)$
- $T[i,j] = 0 \iff j \notin D(i)$
- Chaque colonne  $T[:,j]$  désigne l'ensemble des documents contenant le terme  $t_j$
- Chaque ligne  $T[i,:]$  désigne l'ensemble des termes contenus dans le document  $d_i$

Soit  $q$  une requête constituée de plusieurs termes  $t_1, t_2, \dots, t_k$ .

Cette requête s'interprète comme la recherche des documents contenant à la fois les termes  $t_1$  et

$\$j_2, \dots\$$  et  $\$j_k\$$ .

L'ensemble des documents sélectionnés par la requête est donnée par l'opérateur booléen "and":  $\$T[:,j_1] \text{ and } T[:,j_2] \text{ and } \dots \text{ and } T[:,j_k] \$$

On note  $\$\{i_1, i_2, \dots\}\$$  la liste des documents sélectionnés par la requête  $\$q\$$ .

## Similarité

Pour chaque document  $\$i\$$  sélectionné par fichier inverse, on calcule un score de **similarité** :  $\$\text{sim}(i,q)\$$

Les documents sont ensuite classés par ordre *décroissant* de score. La réponse est une liste *ordonnée* de documents :  $\$(i_1, i_2, \dots)\$$  avec  $\text{sim}(i_1,q) \geq \text{sim}(i_2,q) \geq \dots$

## 4. similarité : approche binaire

- Un document  $\$d\$$  est une liste ordonnée de termes :  $\$d = (d[1], d[2], \dots, )\$$
- On cherche une représentation "quantitative" permettant de réaliser des comparaisons entre documents.
- L'approche la plus courante : "Bag of words" : on regarde les distribution des fréquence d'occurrence des mots dans un texte sans se préoccuper de leur position dans le texte.

## Représentation ensembliste des documents

un document  $\$d\$$  est représenté par l'**ensemble des termes qu'ils contient** (sans se préoccuper de leur ordre)

### Similarité de Jaccard

Soient deux ensembles A et B.

L'indice de Jaccard est le rapport entre la cardinalité (la taille) de l'intersection des ensembles considérés et la cardinalité de l'union des ensembles. Il permet d'évaluer la similarité entre deux ensembles:  $\$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \$$

## Représentation vectorielle des documents

- Un document  $\$d\$$  est représenté par sous la forme d'un vecteur **binaire**  $\boldsymbol{x}$  de dimension  $\$m\$$
- $\$m\$$  est la taille du vocabulaire
- Chaque terme  $\$t \in V\$$  est indexé par un entier  $\$i \in 1..m\$$
- $\$x_i = 1 \Leftrightarrow i \in d\$$
- $\$x_i = 0 \Leftrightarrow i \notin d\$$

## Similarité vectorielle

- Similarité basée sur la distance :
  - Distance vectorielle :
    - distance de Hamming :

$$\text{dist}(\mathbf{x}, \mathbf{y}) = \sum_{i \in 1..m} |x_i - y_i|$$

- Similarité :

$$\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{1}{1 + \text{dist}(\mathbf{x}, \mathbf{y})}$$

- Similarité du cosinus :

$$\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \times \|\mathbf{y}\|}$$

## 5. Statistiques sur les textes

On dispose d'une base  $B$  de  $n$  documents textes.

On note  $B = \{d_1, d_2, \dots\}$ , où chaque  $d_i$  représente un document différent de la base.

Les documents sont écrits dans un langage  $L$  obéissant à un vocabulaire  $V$ .

On note  $V = \{t_1, t_2, \dots\}$  où chaque  $t_i$  est un terme du vocabulaire  $V$ .

On note  $A$  l'alphabet : ensemble des symboles (caractères) utilisés par le langage  $L$ .

On note  $\alpha \in A$  un caractère de l'alphabet  $A$ .

Soit  $d$  un document de  $B$ . Il peut être décrit comme :

- une suite de symboles :  $d = (\alpha_1, \alpha_2, \dots)$  avec  $\alpha_i \in A$ .
- ou une suite de mots :  $d = (t_1, t_2, \dots)$  avec  $t_i \in V$ .

[Statistiques sur les lettres](#)

[Statistiques sur les termes](#)

## 6. Représentation vectorielle

- Un document  $d$  est représenté par sous la forme d'un vecteur **réel**  $\mathbf{x}$  de dimension  $m$
- $m$  est la taille du vocabulaire
- Chaque terme  $t \in V$  est indexé par un entier  $j \in 1..m$
- $x_j = h(j, d) \rightarrow j \in d$ 
  - $h(j, d)$  correspond :

- à la fréquence de  $f_d(j)$  dans  $d$  :

$f_d(j)$

- au score TF-IDF de  $f_d(j)$  dans  $d$  :

$f_d(j) \log g_B(j)$

- $x_j = 0 \rightarrow j \notin d$

### Comparaison vectorielle

- Similarité euclidienne
  - distance euclidienne :

$\text{dist}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i \in 1..m} (x_i - y_i)^2}$

- similarité euclidienne :

$\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{1}{1 + \text{dist}(\mathbf{x}, \mathbf{y})}$

- Similarité du cosinus :

$\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \times \|\mathbf{y}\|}$

## 7. Recherche Web

### PageRank

Le World Wide Web (www) est un réseau



- formé de documents (les pages html) hébergées sur des serveurs,
- les serveurs sont localisés à l'aide de leur adresse (url : universal resource locator)
- les documents sont liés entre eux par des liens hypertextes.

Les algorithmes d'indexation utilisés par les principaux moteurs de recherche prennent en compte la popularité des différentes pages~; les pages les plus "populaires" reçoivent un score plus élevé qui a tendance à les faire apparaître en premier dans la liste des réponses, selon:



$\text{score}(i, q) = \text{PR}(i) \times \text{sim}(i, q)$  où  $\text{PR}(i)$  est la popularité de la page  $i$ .





Last update: 2019/01/11 10:39 public:omi-5a-o-rech:1.\_recherche\_et\_indexation [https://wiki.centrale-med.fr/informatique/public:omi-5a-o-rech:1.\\_recherche\\_et\\_indexation](https://wiki.centrale-med.fr/informatique/public:omi-5a-o-rech:1._recherche_et_indexation)

---

From: <https://wiki.centrale-med.fr/informatique/> - **WiKi informatique**

Permanent link: [https://wiki.centrale-med.fr/informatique/public:omi-5a-o-rech:1.\\_recherche\\_et\\_indexation](https://wiki.centrale-med.fr/informatique/public:omi-5a-o-rech:1._recherche_et_indexation)

Last update: **2019/01/11 10:39**

