Data@WEB: Web Data Intelligence

Création de valeur autour des données du WEB

Cours 3 : Modèles Probabilistes

M. Tami (the author of this chapter), C. Hudelot, W. Ouerdane, B.L. Doan

2026

CentraleSupélec

Sommaire

1. Introduction

2. Principe d'Ordonnancement Probabiliste (POP)

3. Modèle d'Indépendance Binaire (MIB)

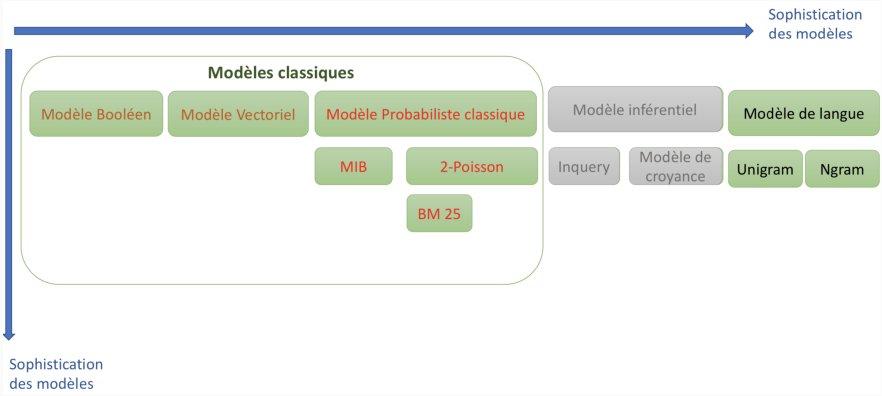
4. Indexation probabiliste et modèle 2-Poisson

5. Modèle Okapi BM25

6. Bilan

Introduction

Les familles de modèles en Recherche d'Information (RI)



Modélisation de la notion de pertinence

La notion de pertinence est centrale pour les modèles de RI

- Elle va permettre de **prendre la décision** de présenter ou non un document suite à la requête d'un utilisateur
- Elle va permettre d'**ordonner** les documents présentés à l'utilisateur
- Elle est centrale pour la **satisfaction** de l'utilisateur et la **performance** du modèle

Les modèles déjà vus et leur modélisation de la pertinence :

- Modèle booléen : pertinence binaire
- Modèle vectoriel : degré de similarité

Dans ce cours : le modèle probabiliste

- Degré de pertinence
- Probabilité pour un document d'être pertinent par rapport à une requête utilisateur

Modèle probabiliste pour la RI

Principe

- Modélisation de la recherche comme un problème de classification
- On dispose d'un ensemble de documents qui, pour chaque requête q, se divise en deux classes :
  - Documents *pertinents* : $\mathcal{R}$
  - Documents *non pertinents* : $\bar{\mathcal{R}}$
- **Objectif** : étant donné un document noté d et une requête q fixée, trouver la probabilité du document d'appartenir à $\mathcal{R}$
- **Règle** : on ne retourne le document d que si

$$P(d \in \mathcal{R}|d, q) > P(d \in \bar{\mathcal{R}}|d, q)$$

$\hookrightarrow$ **Rang de classement** de plusieurs documents basé sur ces probabilités :

$$\frac{P(d \in \mathcal{R}|d, q)}{P(d \in \bar{\mathcal{R}}|d, q)}$$

Plusieurs modèles probabilistes

- Tous ont pour **support une même expérience aléatoire** :
  - une urne contenant :
    - un groupe de requêtes $q_1, \cdots, q_l, \cdots, q_L$ (réalisations d'une variable Q)
    - un groupe de documents $d_1, \cdots, d_i, \cdots, d_N$ (réalisations d'une variable D)
  - une réalisation de l'expérience : tirage simultané d'un couple (q_l, d_i)
- Les modèles probabilistes se distinguent par
  - les événements qu'ils considèrent (en voici trois exemples) :
    - $P(d \in \mathcal{R} | D = d_i, Q = q_l)$ aussi nommé **probabilité de pertinence** de d_i vis-à-vis de q_l
    - $P(Q = q_l, D = d_i)$
    - $P(Q = q_l | D = d_i)$
  - les distributions (ou lois) de probabilités qu'ils utilisent

Modèle probabiliste pour la RI

Probabiliste vs booléen
Les modèles probabilistes permettent de classer les documents mais avec plus de nuances

Probabiliste vs vectoriel

- Modèle vectoriel : classe les documents en fonction de leur similarité avec la requête
- La notion de **similarité ne permet pas de prendre en compte** le fait suivant :
  est-ce que le document est un bon document à renvoyer à l'utilisateur ?
  $\hookrightarrow$ Le document le plus similaire peut être fortement pertinent ou pas du tout
- Modèle probabiliste $\rightsquigarrow$ vers une plus forte formalisation de la pertinence

Modèle probabiliste : illustration de son intérêt en RI

Idée par l'exemple

- Requête : *chat*
- *chat* ? : l'**animal** ou la **discussion ?** $\rightarrow$ aucune distinction pour le modèle vectoriel, les deux sont similaires
  Si on a accès à :
- Probabilité qu'un document sur les chats (mammifères) soit pertinent pour l'utilisateur : 0,8
- Probabilité que l'utilisateur veuille en savoir plus sur les chats (discussion) : 0.2
  $\hookrightarrow$ La **probabilité aide à la décision/ordonnancement** des documents

Plusieurs modèles probabilistes

- Modèles de recherche probabilistes classiques :
  - Principe d'Ordonnancement Probabiliste (POP) $\sim$ en anglais, Probability Ranking Principle (PRP)
    - Modèle d'Indépendance Binaire (MIB) $\sim$ en anglais, Binary Independence Model (BIM)
    - Modèle Okapi BM 25 (BM 25)

    $\hookrightarrow$ Moteurs de recherche (web, entreprises, documentaires, etc.)
- Modèles bayesiens pour la recherche textuelle (non abordé dans ce cours)
- Modèles de langues (non abordé dans ce cours)

$\longrightarrow$ Un sujet important en RI

Principe d'Ordonnancement Probabiliste (POP)

Principe d'Ordonnancement Probabiliste (POP)

- On a :
  - Une collection de documents $\mathcal{C} = \{d_1, \cdots, d_i, \cdots, d_N\}$
  - Une requête q : besoin de l'utilisateur
- Hypothèse : La pertinence d'un document vis-à-vis d'une requête est indépendante des autres documents de la collection

$\hookrightarrow$ Question : dans quel ordre présenter les documents à l'utilisateur ?

- Le POP va répondre à cette question
- L'objectif du POP sera de présenter les documents par ordre de pertinence vis-à-vis de la requête soumise par l'utilisateur

Principe d'Ordonnancement Probabiliste (POP)

Les points importants

- Retour = liste ordonnée de documents
- Le score du document est fonction de sa probabilité de pertinence
- L'ordre des documents est plus important que leur score

Principe important : la pertinence de chaque document est indépendante de la pertinence des autres documents

Principe d'Ordonnancement Probabiliste (POP)

Principe : une expérience support

- Chaque paire (d, q) est la réalisation d'une expérience aléatoire :

"Tirer simultanément une paire (d, q) d'un ensemble prédéfini de documents et de requêtes."

- A chaque tirage est associé
  - deux variables aléatoires D et Q
    ayant pour observations possibles respectives d et q
  - une variable aléatoire binaire $R_{d,q}$
    - $R_{d,q} = 1$ si le document est pertinent par rapport à la requête
    - $R_{d,q} = 0$ sinon
- **Objectif** :
  - Modéliser l'expérience aléatoire par un modèle probabiliste donné
  - Estimer les paramètres du modèle sur une base de requêtes et de documents
  - Pour une nouvelle requête q_{new}, **trier/ordonner** les documents d en fonction de leur probabilité a posteriori $P(R_{d,q} = 1 | D = d, Q = q_{new})$

Principe d'Ordonnancement Probabiliste (POP)

Principe : allégement des notations

- La variable aléatoire **binaire** $R_{d,q}$ représente la pertinence d'un document d par rapport à la requête q (notion binaire de la pertinence)
- Dans la suite, **on utilisera souvent** R **pour** $R_{d,q} = 1$ **et** NR **pour** $R_{d,q} = 0$ **pour alléger les notations**
- On ordonne les documents par l'estimation de leur probabilité de pertinence par rapport à la requête $P(R|d, q)$
- Dans la suite, on considérera que q est toujours donnée et **on utilisera** $P(R|d)$ **au lieu de** $P(R|d, q)$ **pour alléger les notations**

Principe d'Ordonnancement Probabiliste (POP)

Principes de base et importance des termes

- La probabilité qu'un document soit pertinent pour une requête dépend seulement des termes de la requête et des termes utilisés pour indexer le document
- Étant donnée une requête de l'utilisateur, l'ensemble de réponse idéal est $\mathcal{R}$ (l'ensemble des documents pertinents)
- Étant donnés une requête q et un document d dans une collection de documents, le principe du modèle probabiliste est donc d'estimer la probabilité que l'utilisateur trouve d pertinent, i.e. la probabilité que d se trouve dans $\mathcal{R}$

Principe d'Ordonnancement Probabiliste (POP)

Principe : probabilité de pertinence

- Contexte : q une requête fixée ; d un document de la collection $\mathcal{C}$; R une variable "pertinence" d'un document vs une requête ; NR "non-pertinence"
- On doit trouver $P(R|d)$: probabilité que le document retrouvé soit pertinent (i.e. **probabilité de pertinence**)
- On utilise le théorème de Bayes :

$$P(R|d) = \frac{P(d|R)P(R)}{P(d)} \text{ et } P(NR|d) = \frac{P(d|NR)P(NR)}{P(d)}$$

  - $P(R)$: probabilité a priori de trouver un document pertinent ou probabilité de pertinence d'un document quelconque
  - $P(d)$: probabilité que d soit choisi
  - $P(d|R)$ probabilité que si un document pertinent est retrouvé, alors il s'agit de d

Principe : règle de décision

Si $P(R|d) > P(NR|d)$ alors d est pertinent pour q sinon il n'est pas pertinent

Principe d'Ordonnancement Probabiliste (POP)

Attention, quelques précisions :

- $P(R|d,q) = \frac{P(d|R,q)P(R|q)}{P(d|q)}$
- $P(R|q)$ probabilité a priori de trouver un document pertinent pour une requête q ou probabilité de pertinence d'un document quelconque
- $P(d|q)$: probabilité que d soit choisi pour une requête q
- $P(d|R,q)$ probabilité que si un document pertinent (pour q) est retrouvé, alors il s'agit de d

Principe d'Ordonnancement Probabiliste (POP)

Attention, quelques précisions :

- $P(R|d,q) = \frac{P(d|R,q)P(R|q)}{P(d|q)}$
- $P(R|q)$ probabilité a priori de trouver un document pertinent pour une requête q ou probabilité de pertinence d'un document quelconque
- $P(d|q)$: probabilité que d soit choisi pour une requête q
- $P(d|R,q)$ probabilité que si un document pertinent (pour q) est retrouvé, alors il s'agit de d

Modèle probabiliste de base : idée de score de pertinence

- Rappelons nous du **rang de classement** vu au début de ce cours

$$\frac{P(d \in \mathcal{R}|d,q)}{P(d \in \bar{\mathcal{R}}|d,q)}$$

- On a $P(R|d) = \frac{P(d|R)P(R)}{P(d)}$ et $P(NR|d) = \frac{P(d|NR)P(NR)}{P(d)}$

  $\hookrightarrow$ éléments de base pour construire un score de pertinence : une mesure de la probabilité de pertinence

- On remarque que $P(R)$, $P(d)$ et $P(NR)$ sont des constantes
- Comme seul l'ordre importe, la multiplication par une constante peut être ignorée (pas d'influence)

  $\Rightarrow$ Un score mesurant la probabilité de pertinence d'un document vis-à-vis d'une requête ne dépend que de $P(d|R)$ et $P(d|NR)$

Modèle probabiliste de base : idée de score de pertinence

- Rappelons nous du **rang de classement**

$$\frac{P(d \in \mathcal{R}|d,q)}{P(d \in \bar{\mathcal{R}}|d,q)}$$

- On a $P(R|d) = \frac{P(d|R)\cancel{P(R)}}{\cancel{P(d)}}$ et $P(NR|d) = \frac{P(d|NR)\cancel{P(NR)}}{\cancel{P(d)}}$

  $\hookrightarrow$ éléments de base pour construire un score de pertinence : une mesure de la probabilité de pertinence

- $P(R)$, $P(d)$ et $P(NR)$ sont des constantes
- Comme seul l'ordre importe, la multiplication par une constante peut être ignorée (pas d'influence)

  $\Rightarrow$ Un score mesurant la probabilité de pertinence d'un document vis-à-vis d'une requête ne dépend que de $P(d|R)$ et $P(d|NR)$

Modèle probabiliste de base : idée de score de pertinence

- Rappelons nous du **rang de classement**

$$\frac{P(d \in \mathcal{R}|d,q)}{P(d \in \bar{\mathcal{R}}|d,q)}$$

- On a $P(R|d) = \frac{P(d|R)\cancel{P(R)}}{\cancel{P(d)}}$ et $P(NR|d) = \frac{P(d|NR)\cancel{P(NR)}}{\cancel{P(d)}}$

  $\hookrightarrow$ éléments de base pour construire un score de pertinence : une mesure de la probabilité de pertinence

- $P(R)$, $P(d)$ et $P(NR)$ sont des constantes
- Comme seul l'ordre importe, la multiplication par une constante peut être ignorée (pas d'influence)

  $\Rightarrow$ Un score mesurant la probabilité de pertinence d'un document vis-à-vis d'une requête ne dépend que de $P(d|R)$ et $P(d|NR)$

Principe d'Ordonnancement Probabiliste (POP)

Modèle probabiliste de base : $P(R|d_i)$ et $P(d_i|R)$

Étant donnée une collection $\mathcal{C}$ de N documents $(d_1, \cdots, d_i, \cdots, d_N)$, nous avons à ce stade 2 probabilités d'intérêt :

- $P(R|d_i)$ est la probabilité de pertinence sachant que d_i a été choisi
- $P(d_i|R)$ est la probabilité que si un document pertinent est retrouvé, alors il s'agit de d_i
- En général, on a $P(R|d_i) + P(NR|d_i) = 1$
  Un document est soit pertinent soit non-pertinent pour une requête
- En général, on a $P(d_i|R) + P(d_i|NR) \neq 1$ mais $\sum_{i \in \mathcal{C}} P(d_i|R) = 1$

Principe d'Ordonnancement Probabiliste (POP)

Théorème

Trouver les documents qui maximisent $P(R|d_i)$ est équivalent à trouver les documents qui maximisent :

- soit $\frac{P(R|d_i)}{P(NR|d_i)}$,
- soit $\frac{P(d_i|R)}{P(d_i|NR)}$

$\hookrightarrow$ Idée : s'en servir comme score de pertinence basé sur la probabilité de pertinence
$\hookrightarrow$ In fine, ce serait une proposition de critère d'ordonnancement des documents d_i de la collection $\mathcal{C}$ vis-à-vis d'une requête q donnée

Principe d'Ordonnancement Probabiliste : POP

- Théorème $\Rightarrow$ Démonstration

Idée de la preuve : premier point

- $\frac{P(R|d_i)}{P(NR|d_i)} = \frac{P(R|d_i)}{1-P(R|d_i)}$
- $f(x) = \frac{x}{1-x}$ est une fonction strictement croissante sur $x \in [0, 1]$
- Les documents qui maximisent $\frac{P(R|d_i)}{P(NR|d_i)}$ sont les mêmes qui maximisent $P(R|d_i)$

Il nous reste à montrer que les documents qui maximisent $\frac{P(R|d_i)}{P(NR|d_i)}$ maximisent aussi $\frac{P(d_i|R)}{P(d_i|NR)}$

Idée de la preuve : deuxième point

- Théorème de Bayes :

$$P(R|d_i) = \frac{P(d_i|R)P(R)}{P(d_i)} \text{ et } P(NR|d_i) = \frac{P(d_i|NR)P(NR)}{P(d_i)}$$

- Divisons le premier par le second, $\frac{P(R|d_i)}{P(NR|d_i)} = \frac{P(d_i|R)P(R)}{P(d_i|NR)P(NR)}$
- Or, $\frac{P(R)}{P(NR)}$ est une constante indépendante du document pour une requête donnée $\Rightarrow$ le résultat est démontré.

Principe d'Ordonnancement Probabiliste : POP

Il nous reste à montrer que les documents qui maximisent $\frac{P(R|d_i)}{P(NR|d_i)}$ maximisent aussi $\frac{P(d_i|R)}{P(d_i|NR)}$

Idée de la preuve : deuxième point

- Théorème de Bayes :

$$P(R|d_i) = \frac{P(d_i|R)P(R)}{P(d_i)} \text{ et } P(NR|d_i) = \frac{P(d_i|NR)P(NR)}{P(d_i)}$$

- Divisons le premier par le second, $\frac{P(R|d_i)}{P(NR|d_i)} = \frac{P(d_i|R)\cancel{P(R)}}{P(d_i|NR)\cancel{P(NR)}}$
- Or, $\frac{P(R)}{P(NR)}$ est une constante indépendante du document pour une requête donnée $\Rightarrow$ le résultat est démontré.

Mesure de similarité basée sur la notion de probabilité

Pour conclure cette section, une mesure de similarité fondée sur la notion de probabilité de pertinence est proposée

Définition
Les trois items suivants sont équivalents :

- $sim(d_i, q) = \frac{P(R|d_i)}{P(NR|d_i)}$
- $sim(d_i, q) = \frac{P(d_i|R)P(R)}{P(d_i|NR)P(NR)}$ (en utilisant le théorème de Bayes)
- $sim(d_i, q) = \frac{P(d_i|R)}{P(d_i|NR)} k$ avec k qui est une constante pour une requête donnée

On va se servir de cette fonction de similarité comme score d'ordonnancement

Modèle d'Indépendance Binaire (MIB)

Modèle d'Indépendance Binaire (MIB)

Modèle traditionnellement utilisé avec le POP. Il pose un certain nombre d'hypothèses permettant d'estimer les probabilités

1. H1 : les documents et les requêtes sont représentés sous la forme de vecteurs binaires de même taille que le vocabulaire de termes (cf. cours 1)
   $\Rightarrow$ représentation binaire des documents
2. H2 : Les termes présents dans le document sont mutuellement indépendants.
   $\Rightarrow$ approche sac-de-mots (Naïve Bayes)
3. H3 : La règle d'ordonnancement est la suivante : un document d est pertinent par rapport à q si :

$$P(R|d,q) > P(NR|d,q)$$

4. H4 : tous les termes non-présents dans la requête sont uniformément répartis dans les documents pertinents et non-pertinents par rapport à cette requête.

Modèle d'Indépendance Binaire (MIB)

Principe

- Binaire (H1) $\Rightarrow$ Document := vecteur binaire $\mathbf{d} = (t_1, \cdots, t_j, \cdots, t_V)$ avec $t_j = 0$ ou 1 qui indique la présence ou l'absence du terme d'index t_j
- Requête est aussi un vecteur binaire $\mathbf{q} = (r_1, \cdots, r_j, \cdots, r_V)$
- **Il faut estimer comment les termes du document contribuent à la pertinence**
- On estime donc $P(d|R)$ par $P(\mathbf{d}|R)$
- Indépendence (H2) $\Rightarrow$ On considère que les différents termes sont indépendants et on a donc :

$$P(\mathbf{d}|R) = \prod_{t_j \in \mathbf{d}} P(t_j|R) \prod_{t_j \notin \mathbf{d}} (1 - P(t_j|R))$$

  $\hookrightarrow$ $P(\mathbf{d}|R)$ peut être estimée comme étant le produit des probabilités de pertinence associées à chaque terme dans le document, multiplié par le produit des probabilités que les termes absents n'apparaissent pas dans un document pertinent

Valide que si on considère que la présence d'un terme dans un document est indépendante de la présence des autres termes et que la pertinence d'un document ne dépend que des termes qu'il contient

Modèle d'Indépendance Binaire (MIB)

- Comme $t_j = 0$ ou 1 on a :
  - $p_j = P(t_j = 1|R)$ et $1 - p_j = P(t_j = 0|R)$
  - $s_j = P(t_j = 1|NR)$ et $1 - s_j = P(t_j = 0|NR)$
  - p_j probabilité de pertinence de t_j, s_j probabilité de non-pertinence de t_j (probabilité que le terme apparaisse ou non dans un document pertinent, resp. non pertinent)
- On cherche $sim(d_i, q) = \frac{P(R|d_i)}{P(NR|d_i)} = \frac{P(d_i|R)}{P(d_i|NR)} k$. On a en prenant les notations précédentes (contexte MIB) :
  - $P(\mathbf{d}|R) = \prod_{j=1,t_j=1}^{V} p_j \prod_{j=1,t_j=0}^{V} (1 - p_j) = \prod_{j=1}^{V} p_j^{t_j} (1 - p_j)^{(1-t_j)}$
  - $P(\mathbf{d}|NR) = \prod_{j=1}^{V} s_j^{t_j} (1 - s_j)^{(1-t_j)}$

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$
- $sim(d_i, q) = k \prod_{j=1}^{V} \left(\frac{p_j}{s_j}\right)^{t_j} \left(\frac{1-p_j}{1-s_j}\right)^{(1-t_j)}$ après injection des formules de $P(\mathbf{d}_i|R)$ et $P(\mathbf{d}_i|NR)$

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$
- $sim(d_i, q) = k \prod_{j=1}^{V} \left(\frac{p_j}{s_j}\right)^{t_j} \left(\frac{1-p_j}{1-s_j}\right)^{(1-t_j)}$ après injection des formules de $P(\mathbf{d}_i|R)$ et $P(\mathbf{d}_i|NR)$
- $sim(d_i, q) = k \underbrace{\prod_{j:t_j=1} \left(\frac{p_j}{s_j}\right)}_{\text{terme present}} \underbrace{\prod_{j:t_j=0} \left(\frac{1-p_j}{1-s_j}\right)}_{\text{terme absent}}$ nouveau parcours de l'opérateur produit

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$
- $sim(d_i, q) = k \prod_{j=1}^{V} \left(\frac{p_j}{s_j}\right)^{t_j} \left(\frac{1-p_j}{1-s_j}\right)^{(1-t_j)}$ après injection des formules de $P(\mathbf{d}_i|R)$ et $P(\mathbf{d}_i|NR)$
- $sim(d_i, q) = k \underbrace{\prod_{j:t_j=1} \left(\frac{p_j}{s_j}\right)}_{\text{terme present}} \underbrace{\prod_{j:t_j=0} \left(\frac{1-p_j}{1-s_j}\right)}_{\text{terme absent}}$ nouveau parcours de l'opérateur produit
- Hypothèse 4 : les termes qui n'apparaissent pas dans la requête sont équiprobables concernant leurs occurences dans les documents pertinents ou non pertinents, i.e. si $r_j = 0$ alors $p_j = s_j$ et $\frac{p_j}{s_j} = 1$

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$
- $sim(d_i, q) = k \prod_{j=1}^{V} \left(\frac{p_j}{s_j}\right)^{t_j} \left(\frac{1-p_j}{1-s_j}\right)^{(1-t_j)}$ après injection des formules de $P(\mathbf{d}_i|R)$ et $P(\mathbf{d}_i|NR)$
- $sim(d_i, q) = k \underbrace{\prod_{j:t_j=1} \left(\frac{p_j}{s_j}\right)}_{\text{terme present}} \underbrace{\prod_{j:t_j=0} \left(\frac{1-p_j}{1-s_j}\right)}_{\text{terme absent}}$ nouveau parcours de l'opérateur produit
- Hypothèse 4 : les termes qui n'apparaissent pas dans la requête sont équiprobables concernant leurs occurences dans les documents pertinents ou non pertinents, i.e. si $r_j = 0$ alors $p_j = s_j$ et $\frac{p_j}{s_j} = 1$
- D'où, $sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0,r_j=1} \left(\frac{1-p_j}{1-s_j}\right)$

Modèle d'Indépendance Binaire (MIB)

On en déduit les écritures suivantes :

- $sim(d_i, q) = \frac{P(\mathbf{d}_i|R)}{P(\mathbf{d}_i|NR)} k$
- $sim(d_i, q) = k \prod_{j=1}^{V} \left(\frac{p_j}{s_j}\right)^{t_j} \left(\frac{1-p_j}{1-s_j}\right)^{(1-t_j)}$ après injection des formules de $P(\mathbf{d}_i|R)$ et $P(\mathbf{d}_i|NR)$
- $sim(d_i, q) = k \underbrace{\prod_{j:t_j=1} \left(\frac{p_j}{s_j}\right)}_{\text{terme present}} \underbrace{\prod_{j:t_j=0} \left(\frac{1-p_j}{1-s_j}\right)}_{\text{terme absent}}$ nouveau parcours de l'opérateur produit
- Hypothèse 4 : les termes qui n'apparaissent pas dans la requête sont équiprobables concernant leurs occurences dans les documents pertinents ou non pertinents, i.e. si $r_j = 0$ alors $p_j = s_j$ et $\frac{p_j}{s_j} = 1$
- D'où, $sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0, r_j=1} \left(\frac{1-p_j}{1-s_j}\right)$
- On peut donc ne considérer dans les produits que les termes qui apparaissent dans la requête
- Premier produit : termes de la requête trouvés dans le document
- Second produit : termes de la requête non trouvés dans le document

Modèle d'Indépendance Binaire (MIB)

- On inclut les termes de la requête trouvés dans le document ($t_j = 1$ et $r_j = 1$) dans le second produit et on divise simultanément (pour garder l'$\Leftrightarrow$)

Modèle d'Indépendance Binaire (MIB)

- On inclut les termes de la requête trouvés dans le document ($t_j = 1$ et $r_j = 1$) dans le second produit et on divise simultanément (pour garder l'$\Leftrightarrow$)

$$sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0,r_j=1} \left(\frac{1-p_j}{1-s_j}\right) \times \frac{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}$$

Modèle d'Indépendance Binaire (MIB)

- On inclut les termes de la requête trouvés dans le document ($t_j = 1$ et $r_j = 1$) dans le second produit et on divise simultanément (pour garder l'$\Leftrightarrow$)

$$sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0,r_j=1} \left(\frac{1-p_j}{1-s_j}\right) \times \frac{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}$$

- $sim(d_i|q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right) \prod_{j:r_j=1} \left(\frac{1-p_j}{1-s_j}\right)$ après simplifications

Modèle d'Indépendance Binaire (MIB)

- On inclut les termes de la requête trouvés dans le document ($t_j = 1$ et $r_j = 1$) dans le second produit et on divise simultanément (pour garder l'$\Leftrightarrow$)

$$sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0,r_j=1} \left(\frac{1-p_j}{1-s_j}\right) \times \frac{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}$$

- $sim(d_i|q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right) \prod_{j:r_j=1} \left(\frac{1-p_j}{1-s_j}\right)$ après simplifications
- Le second produit concerne tous les termes de la requête et est constant pour une requête donnée
- La seule quantité à estimer pour ordonner les documents est $\prod_{j:t_j=r_j=1} \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right)$

Modèle d'Indépendance Binaire (MIB)

- On inclut les termes de la requête trouvés dans le document ($t_j = 1$ et $r_j = 1$) dans le second produit et on divise simultanément (pour garder l'$\Leftrightarrow$)

$$sim(d_i, q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j}{s_j}\right) \prod_{j:t_j=0,r_j=1} \left(\frac{1-p_j}{1-s_j}\right) \times \frac{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}{\prod_{j:t_j=r_j=1} \left(\frac{1-p_j}{1-s_j}\right)}$$

- $sim(d_i|q) = k \prod_{j:t_j=r_j=1} \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right) \prod_{j:r_j=1} \left(\frac{1-p_j}{1-s_j}\right)$ après simplifications
- Le second produit concerne tous les termes de la requête et est constant pour une requête donnée
- La seule quantité à estimer pour ordonner les documents est $\prod_{j:t_j=r_j=1} \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right)$
- Transformation logarithmique pour sa propriété de monotonie et pour classer de manière égale chaque document
- La quantité utilisée pour le classement est appelée RSV (Retrieval Status Value) et vaut :

$$RSV_d = log \prod_{j:t_j=r_j=1} \frac{p_j(1-s_j)}{s_j(1-p_j)} = \sum_{j:t_j=r_j=1} log \left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right)$$

RSV

- On cherche $RSV_d = \sum_{j:t_j=r_j=1} log\left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right)$
- Notation : $RSV_d = \sum_{j:t_j=r_j=1} c_j$ avec $c_j = log\left(\frac{p_j(1-s_j)}{s_j(1-p_j)}\right)$

Interprétation :

- $c_j = 0$: le terme a autant de chance d'apparaître dans un document pertinent ou non pertinent.
- $c_j > 0$: le terme a plus de chance d'apparaître dans un document pertinent.
- $c_j < 0$: le terme a plus de chance d'apparaître dans un document non-pertinent.

Problème
Comment calculer c_j avec nos données pour une collection et une requête donnée ?

Calcul de la probabilité des termes : approche théorique

Pour chaque terme t_j de la requête, estimer c_j dans la collection à l'aide de tables de contingence des occurrences des documents dans la collection :

	Documents	Pertinent (R)	Non-Pertinent (NR)	Total
Terme présent	$\{t_j = 1\}$	r	$df_{t_j} - r$	df_{t_j}
Terme absent	$\{t_j = 0\}$	$R_p - r$	$N - df_{t_j} - R_p + r$	$N - df_{t_j}$
	Total	R_p	$N - R_p$	N

où,

- df_{t_j} : le nombre de documents contenant le terme t_j
- N : nombre total de documents dans $\mathcal{C}$ dont R_p pertinents
- r : nombre de documents pertinents contenant t_j (dans la requête)

Calcul de la probabilité des termes : approche théorique (suite)

Pour chaque terme t_j de la requête, ce type de **table de contingence** permet d'estimer c_j comme suit :

- $p_j = \frac{r}{R_p}$
- $s_j = \frac{df_{t_j} - r}{N - R_p}$
- $1 - p_j = \frac{R_p - r}{R_p}$
- $1 - s_j = \frac{N - df_{t_j} - R_p + r}{N - R_p}$

$$\Rightarrow c_j = log \frac{r/(R_p - r)}{(df_{t_j} - r)/(N - df_{t_j} - R_p + r))}$$

- Pour éviter les zéros, on ajoute 0.5 à chaque quantité du tableau (lissage) :

$$c_j = log \frac{(r + 0.5)/(R_p - r + 0.5)}{(df_{t_j} - r + 0.5)/(N - df_{t_j} - R_p + r + 0.5))}$$

Modèle d'Indépendance Binaire (MIB)

- On a une table de contingence théorique : et alors ?!
- Comment faire (malgré cette table de contingence) si on ne connait pas par exemple les valeurs de r ?
- Faire de l'apprentissage ? Pour cela on a besoin de données d'apprentissage de type (d_i, q) labélisés avec assez d'exemples pertinents et non pertinents $\hookrightarrow$ On a besoin de données passées de classements renvoyés à l'utilisateur qui sont labélisés par l'utilisateur (pertinent vs non pertinent)
- Si pas d'information sur les documents pertinents $\Rightarrow$ Pas de données d'apprentissage

$\Rightarrow$ **Nous avons besoin de méthode alternative pour la mise en pratique**

Calcul de la probabilité des termes : Approche Pratique

Estimation de s_j

- Hypothèse : les documents pertinents représentent une faible proportion de la collection. Donc, on peut approximer les statistiques pour les documents non pertinents par les statistiques sur la collection entière.
- $P(t_j|NR) = s_j$ est calculée par la distribution des t_j dans la collection : $\frac{df_{t_j}}{N}$, soit IDF . Si le terme est très rare, la probabilité de non-pertinence sera basse et vice versa.

$$\log(\frac{1-s_j}{s_j}) = \log(\frac{N-df_{t_j}}{df_{t_j}}) \approx \log(\frac{N}{df_{t_j}})$$

$\rightsquigarrow$ **Cette approximation ne peut pas s'étendre aux documents pertinents et ne peut pas s'appliquer à** p_j

Calcul de la probabilité des termes : approche pratique

Estimation de p_j

- Initialement, pas de documents retrouvés, pas de jugements de pertinence.
- Le nombre de documents pertinents (R_p dans le tableau précédent) est non connu
- Estimation de $p_j = P(t_j|R)$
  - Sol 1 : [1] $P(t_j|R)$ est constant (ex : 0.5 si aucune info).
  - Sol 2 : Proportionnel à la probabilité d'occurrence dans la collection
  - Sol 3 [2] : Proportionnel au log de la probabilité d'occurrence dans la collection

Modèle d'Indépendance Binaire (MIB)

Estimation des probabilités à l'aide de données d'apprentissage

- On suppose que :
  - L'utilisateur a vu le premier classement
  - L'utilisateur a labelisé plusieurs des documents comme pertinents

  $\Rightarrow$ Données d'apprentissage disponnibles
- On a après ce retour de pertinence
  - N documents dans la collection avec R_p qui sont pertinents
  - df_{t_j} documents contiennent t_j, dont r sont pertinents.

$\rightsquigarrow$ **On peut dans ce cas appliquer l'approche théorique avec la table de contingence.**

Modèle MIB : Bilan

- Avantages :
  - Une formalisation puissante
  - Modélisation explicite de la notion de pertinence
- Inconvénients :
  - La fréquence des termes n'est pas prise en compte
  - Difficulté d'estimer les probabilités sans données d'apprentissage
  - Hypothèse forte d'indépendance entre les termes (H2) est souvent critiquée

Indexation probabiliste et modèle 2-Poisson

Modéliser la fréquence des termes

Quels mots doit-on indexer ?

- Dans un modèle probabiliste, le choix de retenir ou non un terme d'indexation doit être lié à la probabilité qu'un utilisateur désirant ce document écrive ce terme dans la requête.
- Modèle basé sur la distribution théorique des mots dans un document.
- Les occurrences d'un mot t_j dans un document notées $tf_{t_j,d}$ sont distribuées de façon aléatoire : on note TF_j la variable aléatoire associée et la probabilité qu'un mot t_j apparaisse $tf_{t_j,d}$ fois dans un document d suit une loi de Poisson :

$$P(TF_j = tf_{t_j,d}) = \frac{e^{-\lambda}\lambda^{tf_{t_j,d}}}{tf_{t_j,d}!}$$

avec

- λ la moyenne des occurrences du mot dans un document

$\Rightarrow$ **Pour distinguer les mots courants des mots inhabituels**, il suffit donc de vérifier si les occurrences d'un mot $tf_{t_j,d}$ se comportent comme une distribution de Poisson (si oui, le mot est courant).

Indexation probabiliste

- On a constaté que la loi de Poisson décrit bien les mots peu ou pas porteurs de sens : distribution aléatoire
- Les mots porteurs de sens ont tendance à apparaître en groupes. $\Rightarrow$ Les mots porteurs de sens sont ceux dont la distribution s'éloigne de la distribution de Poisson

$\rightarrow$ Les mots inhabituels vont donc particulièrement nous intéresser

Notion d'élitisme

- Principe : on veut sélectionner un terme t pour représenter un document si ce terme apparaît plus fréquemment dans ce document que dans un autre choisi au hasard.
- On veut distinguer les distributions des termes dans les documents où ces termes sont représentatifs de ceux où ils ne le sont pas.
- Élitisme : **on divise l'ensemble des documents en deux groupes** :
  - ceux **qui traitent** du thème représenté par le terme t (dans lesquels le terme t sera plus fréquent) : ensemble *élite*, noté E.
  - ceux **qui ne traitent pas** du thème que t représente et dans lesquels l'apparition de t est marginale $\bar{E}$
- Les distributions du terme t dans les deux groupes sont différentes.

Distribution des termes dans les documents

Distribution mixte 2-Poisson

$$P(TF_j = tf_{t_j,d}) = P(TF_j = tf_{t_j,d} | d \in E)P(d \in E) + P(TF_j = tf_{t_j,d} | d \notin E)P(d \notin E)$$

$$\Rightarrow P(TF_j = tf_{t_j,d}) = \pi . \frac{e^{-\lambda_1} \lambda_1^{tf_{t_j,d}}}{tf_{t_j,d}!} + (1 - \pi) . \frac{e^{-\lambda_2} \lambda_2^{tf_{t_j,d}}}{tf_{t_j,d}!}$$

- Notation : π est la probabilité a priori qu'un document d soit dans le groupe élite (i.e : $P(E)$)
- λ_1 et λ_2 sont les paramètres des lois de poissons de distributions des termes à l'intérieur de chaque groupe ($\lambda_1 \geq \lambda_2$) (moyenne des fréquences des termes dans les groupes élite et non élite)

- On estime les paramètres λ_1 et λ_2 :
  - Si les estimations donnent des valeurs proches, on est en présence d'un terme peu spécifique
  - Si les estimations donnent des valeurs différentes, on est en présence d'un terme spécifique qu'il faut utiliser pour décrire les documents élites de ce terme
- Remarque : Pour les mots vides, la notion d'élitisme n'a pas lieu d'être (la distribution des mots vides sur tous les documents est la même)

Indexation probabiliste

Un autre outils :

- On mesure le degré de recouvrement des deux lois de poissons par :

$$z = \frac{\lambda_1 - \lambda_2}{\sqrt{\lambda_1 + \lambda_2}}$$

- On décide qu'un terme t_j doit indexer un document d si $\eta > 0$ avec

$$\eta = P(d \in E | TF_j = tf_{t_j,d}) + z$$

- η est utilisée comme pondération associée au terme t pour le document d

Indexation probabiliste

- La probabilité qu'un document d soit dans l'ensemble élite d'un terme sachant le nombre d'occurrences $tf_{t_j,d}$ de ce terme dans le document est estimée en utilisant le théorème de Bayes

$$P(d \in E | TF_j = tf_{t_j,d}) = \frac{P(TF_j = tf_{t_j,d} | d \in E) P(d \in E)}{P(TF_j = tf_{t_j,d})}$$

$$P(d \in E | TF_j = tf_{t_j,d}) = \frac{\pi . e^{-\lambda_1} \lambda_1^{tf_{t_j,d}}}{\pi . e^{-\lambda_1} \lambda_1^{tf_{t_j,d}} + (1-\pi) . e^{-\lambda_2} \lambda_2^{tf_{t_j,d}}}$$

Modèle Okapi BM25

Modèle Okapi (Best Match 25)

- Se base sur le modèle probabiliste de base
- MIB a été conçu pour des collections courtes avec des documents de longueur à peu près constante : marche bien dans ce contexte mais sinon pas très performant
  $\hookrightarrow$ MIB ne correspond pas aux collections actuelles.

BM25 : référence dans le développement des systèmes de recherche

- **Incorpore la fréquence des termes** (non binaire)
  $\hookrightarrow$ avec des événements du type $\{TF_j = tf_{t_j,d}\}$ et non présence vs absence de terme
- Incorpore une normalisation de la longueur
  $\hookrightarrow$ les documents de longueur variés sont gérés

Probabilités de pertinence du Modèle Okapi BM 25
BM25 considère **la probabilité de pertinence** :

$$P(TF_j = tf_{t_j,d}|R)$$

où $tf_{t_j,d} \in [0, 1]$ (généralisation de MIB où que 2 valeurs possibles 0 ou 1)

- on estime $p_j = P(TFj = tf_{t_j,d}|R)$, par la probabilité que t_j apparaisse $tf_{t_j,d} > 0$ fois dans les documents pertinents (**probabilité de pertinence**)
- on estime $s_j = P(TF_j = tf_{t_j,d}|NR)$, par la probabilité que t_j apparaisse $tf_{t_j,d} > 0$ fois dans les documents non pertinents (**probabilité de non pertinence**)
- On aura alors $1 - p_j = P(TFj = 0|R)$ et $1 - s_j = P(TFj = 0|NR)$

Modèle Okapi (Best Match 25)

Idée de départ

- Un bon descripteur de document est un terme assez fréquent dans ce document mais qui est relativement rare dans la collection.
- **Un constat** : beaucoup de termes apparaissent avec une fréquence assez basse dans beaucoup de documents d'une collection, alors qu'ils apparaissent avec une fréquence élevée dans un groupe distinct de documents.
  On retrouve la notion de groupe élite avec la modélisation du groupe élite avec une loi de Poisson de paramètre λ_1 et la modélisation du groupe non élite avec une loi de Poisson de paramètre λ_2

Modèle Okapi (Best Match 25)

La probabilité d'apparition d'un terme t_j apparaissant $tf_{t_j,d}$ fois dans un document d peut être exprimée par une loi de mélange de paramètre α_j et β_j selon que le document a été jugé pertinent ou non par rapport à une requête q.

$$P(TF_j = tf_{t_j,d}|R,q) = \alpha_j \frac{\lambda_1^{tf_{t_j,d}} \exp^{-\lambda_1}}{tf_{t_j,d}!} + (1-\alpha_j)\frac{\lambda_2^{tf_{t_j,d}} \exp^{-\lambda_2}}{tf_{t_j,d}!}$$

et

$$P(TF_j = tf_{t_j,d}|NR,q) = \beta_j \frac{\lambda_1^{tf_{t_j,d}} \exp^{-\lambda_1}}{tf_{t_j,d}!} + (1-\beta_j)\frac{\lambda_2^{tf_{t_j,d}} \exp^{-\lambda_2}}{tf_{t_j,d}!}$$

où $\alpha_j := P(E|R)$ et $\beta_j := P(E|NR)$

Modèle Okapi (Best Match 25)

- La fonction de score de BM 25 est inspirée de celle du modèle MIB à la différence que les probabilités de présence et d'absence des termes dans les documents pertinents et non pertinents sont calculées d'après les lois 2-Poisson précédentes
- Une réécriture de la mesure du score de pertinence RSV est :

$$RSV_d = \sum_{j:t_j=r_j=1} \log \delta_j$$

avec

$$\delta_j = \frac{(\alpha_j \lambda_1^{tf_{t_j,d}} \exp^{-\lambda_1} + (1-\alpha_j)\lambda_2^{tf_{t_j,d}} \exp^{-\lambda_2})(\beta_j \exp^{-\lambda_1} + (1-\beta_j)\exp^{-\lambda_2})}{(\beta_j \lambda_1^{tf_{t_j,d}} \exp^{-\lambda_1} + (1-\beta_j)\lambda_2^{tf_{t_j,d}} \exp^{-\lambda_2})(\alpha_j \exp^{-\lambda_1} + (1-\alpha_j)\exp^{-\lambda_2})}$$

NB : δ_j est appelé poids

Modèle Okapi (BM25) : estimation et approximation

- Les paramètres δ_j ne peuvent pas être estimés simplement.
- Etude du comportement asymptotique des δ_j quand $tf_{t_j,d} \to \infty$ [Roberston and Walker,94]
- Avec $\alpha_j > \beta_j$ et $\lambda_1 > \lambda_2$, on peut montrer que :

$$\forall t_j, \lim_{tf_{t_j,d} \to \infty} \log \delta_i \approx \log \frac{\alpha_j(1-\beta_j)}{\beta_j(1-\alpha_j)}$$

- Pour une collection de N documents, le choix des paramètres de mélange :
  - Sans connaissance préalable sur les α_j : on les fixe à 0.5
  - Pour toute requête, la majorité des documents d'une collection sont non pertinents : $\beta_j = \frac{df_{t_j}+0.5}{N}$
- et donc $\forall t_j, \log \delta_j = \log \frac{N-df_{t_j}+0.5}{df_{t_j}+0.5}$ (version lissée des idf des termes)

Modèle Okapi (BM25) : approximation et modification 1

Dans le modèle BM25, deux modifications dans le calcul de la fonction de score :

- Prise en compte du nombre d'occurrences normalisé des termes de la requête dans les documents (prise en compte de la longueur du document) en multipliant dans RSV le logarithme de δ_j par :

$$\frac{(k_1+1)\times tf_{t_j,d}}{k_1((1-b)+b\frac{L_d}{m})+tf_{t_j,d}}$$

  - L_d : longueur du document d
  - $m=\frac{1}{N}\sum_{d\in\mathcal{C}} L_d$: longueur moyenne des documents dans la collection.
  - k_1 : paramètre contrôlant la prise en compte de la fréquence, par défaut $k_1 = 1.2$
  - b : paramètre contrôlant la prise en compte de la longueur, par défaut $b = 0.75$

Modèle Okapi (BM25) : approximation et modification 2

- Prise en compte du nombre d'occurrences normalisé des termes de la requête q dans la requête elle-même en multipliant dans RSV le logarithme de δ_j par :

$$\frac{(k_3 + 1) \times tf_{t_j,q}}{k_3 + tf_{t_j,q}}$$

  - k_3 : paramètre contrôlant la prise en compte de la fréquence, par défaut $k_3 = 1000$
  - $tf_{t_j,q}$: nombre d'occurrences du terme t_j dans la requête q

Modèle Okapi (BM25)

Au final, en tenant compte de l'approximation et des deux modifications nous obtenons une réécriture de la mesure du critère de pertinence

Score d'un document d par rapport à une requête q

$$RSV^{BM25}(d,q) = \sum_{j:t_j=r_j=1} \frac{(k_1+1)\times tf_{t_j,d}}{k_1((1-b)+b\frac{L_d}{m})+tf_{t_j,d}} \times \frac{(k_3+1)\times tf_{t_j,q}}{k_3+tf_{t_j,q}} \times \log \frac{N-df_{t_j}+0.5}{df_{t_j}+0.5}$$

- BM25 est un des modèles les plus important en Recherche d'Information (sur le plan théorique et performance)
- Il est souvent utilisée par les moteurs de recherche même actuels comme par exemple *Qwant*

Bilan

Résumé

- Les modèles probabilistes utilisent la théorie de la probabilité pour modéliser l'incertitude inhérente au processus de recherche
- Les hypothèses sont faites de manière explicite
- Le poids des termes sans information de pertinence est IDF
- Un retour de pertinence (données d'apprentissage) peut améliorer le classement en donnant des estimations de probabilités plus justes
- Dans le modèle de base (MIB) : on n'utilise pas les fréquences des termes entre documents et la longueur des documents n'est pas prise en compte

Parallèle avec le modèle vectoriel

- Concernant les espaces vectoriels, la représentation du document peut être faite par un vecteur dont la dimension j est $log(\frac{p_j(1-s_j)}{(1-p_j)s_j})$
- La requête q est représentée par un vecteur avec valeur 1 dans chaque direction d'un terme de la requête et 0 sinon
- La similarité du MIB est le produit scalaire de ces deux vecteurs
- L'approche probabiliste peut être vue comme une méthode probabiliste pour déterminer les poids dans le modèle vectoriel.

Existants

- Il existe de nombreux modèles probabilistes utilisés par les moteurs de recherche.
- La famille de modèles BM25 est souvent utilisée. Elle permet de prendre en compte la fréquence des termes.
  `http://en.wikipedia.org/wiki/Probabilistic_relevance_model_(BM25)`
- Projet Open Source Xapian : moteur de recherche supportant le modèle probabiliste.
  `http://xapian.org/`
- Mis en place dans Lucene :
  `http://opensourceconnections.com/blog/2015/10/16/bm25-the-next-generation-of-lucene-relevation/`

Lectures conseillées

- Chapitre 11 et 12 - Livre : *Introduction to Information Retrieval*
  `http://nlp.stanford.edu/IR-book/`
- Chapitre 8 - Livre *Information Retrieval : Implementing and Evaluating Search Engines.*
  `https://mitmecsept.files.wordpress.com/2018/05/stefan-bc3bcttcher-charles-l-a-clarke-gordon-v-cormack-information-retrieval-implementing-and-eva.pdf`
- Chapter 7 - Livre *Search Engines. Information Retrieval in Practice.*
  `http://ciir.cs.umass.edu/irbook/`
- Chapitre 3 - Livre de Massih-Reza Amini et Éric Gaussier *Recherche d'information Applications, modèles et algorithmes*
- Les présentations de cours de Pr. Mohand Bourhanem.
  `https://www.irit.fr/~Mohand.Boughanem/Enseignements_RI.php`
- Le cours de Recherche d'information de Chris Manning et Pandu Nayak à l'Université de Stanford
  `https://web.stanford.edu/class/cs276/`

Références

[1] W Bruce Croft and David J Harper. Using probabilistic models of document retrieval without relevance information. Journal of documentation, 35(4) : 285–295, 1979.

[2] Warren R Greiff, W Bruce Croft, and Howard Turtle. Computationally tractable probabilistic modeling of boolean operators. In ACM SIGIR Forum, volume 31, pages 119–128. ACM, 1997.