

Cours de recherche d'information.

Licence 3^{eme} année I.S.I.L.

Université de BBA

enseignant : M.BELAZZOUG.

Plan

- 1) Le model booléen étendu.
- 2) La reformulation des requêtes.
- 3) La pondération des termes.
- 4) Le model vectoriel.
- 5) Le model probabiliste.

1) LE MODEL BOOLEEN ETENDU 1.

Il s'agit d'une extension du modèle booléen en intégrant le principe des pondérations. Il tient compte de l'importance des termes dans la représentation des documents et dans la requête, et ce, en affectant des poids à chaque terme du document et de la requête.

1) Modèle de connaissances :

$T = \{t_i\}$, $i \in [1, .. n]$, Les t_i indexent les documents

– Un document (D) est représenté par :

- Une formule logique de la même manière que le modèle booléen.**
- Une fonction $WD : t \in [0,1]$, qui pour chaque terme de T donne le poids de ce terme dans D. Le poids vaut 0 pour un terme non présent dans le document.**

1. LE MODEL BOOLEEN ETENDU 2.

2) Fonction de correspondance

Similarité notée Sim :

- Cas 1 : Formules inspirées de la logique floue

- $\text{Sim}(D, a \wedge b) = \text{Min} [\text{WD}(a), \text{WD}(b)]$
- $\text{Sim}(D, a \vee b) = \text{Max} [\text{WD}(a), \text{WD}(b)]$
- $\text{Sim}(D, \neg a) = 1 - \text{WD}(a)$

- cas2

$$\text{Sim}(D, a \vee b) = \sqrt{\frac{W_D(a)^2 + W_D(b)^2}{2}}$$

$$\text{Sim}(D, a \wedge b) = 1 - \sqrt{\frac{(1 - W_D(a))^2 + (1 - W_D(b))^2}{2}}$$

1. LE MODEL BOOLEEN ETENDU 3.

Exemple :

1)

	Booléen Strict				Booléen pondéré	
document	A	B	A ou b	A et b	A ou b	A et b
D1	1	1	1	1	1	1
D2	1	0	1	0	$1/(2)^{1/2}$	$1-1/(2)^{1/2}$
D3	0	1	1	0	$1/(2)^{1/2}$	$1-1/(2)^{1/2}$
D4	0	0	0	0	0	0

2)

			Booléen pondéré 1		Booléen pondéré 2	
document	A	B	A ou b	A et b	A ou b	A et b
D1	1	1	1	1	1	1
D2	0.8	1	1	0.8	0.91	0.86
D3	0	0.5	0.5	0	0.35	0.21
D4	0.5	0	0.5	0	0.57	0.28

2. LA REFORMULATION DES REQUETES

S'il est très peu de résultats, alors on peut assouplir la requête initiale en supprimant un terme.

- Par exemple, soit une requête initiale $q = t1 \wedge t2 \wedge t3 \wedge \dots t10$. Si aucun document n'a été trouvé pour cette requête, on peut l'assouplir en la requête q' suivante:

$$\begin{aligned} Q' &= (t2 \wedge t3 \wedge t4 \wedge \dots t10) \vee \\ &(t1 \wedge t3 \wedge t4 \wedge \dots t10) \vee \\ &\dots \\ &(t1 \wedge t2 \wedge t3 \wedge \dots t9). \end{aligned}$$

- Cette reformulation peut continuer si nécessaire.
- Cette façon n'est pas unique. Une autre façon possible de reformuler consiste à supprimer le terme qui est le plus difficile à satisfaire (celui qui correspond au moins de documents).
- On peut en imaginer encore d'autres. En effet, ces assouplissements sont justifiés seulement par des besoins pratiques. Elles n'ont pas besoin de justification théorique.

3. LA PONDERATION DES TERMES 1

- TF (term frequency) :
 - Idée sous-jacente : plus un terme est fréquent dans un document plus il est important dans la description de ce document
 - Exemple de TF : $tf(i, j) = freq(i, j) / \sum_k [f(k, j)]$
- IDF : (Inverse Document Frequency)
 - Idée sous-jacente : plus un terme est fréquent dans une collection moins il est important dans la description de ce document.

On dit qu'un terme est plus **discriminant** si il est moins fréquent dans une collection.

avec

N : la taille de la collection,
ni le nombre de documents
contenant le terme t_i

$$\log \left(\frac{N}{n_i} \right)$$

3. LA PONDERATION DES TERMES 2

Une autre pondération, la combinaison tf-Idf :

- $tf(t, D) = freq(t, D)$
- $tf(t, D) = \log[freq(t, D)]$
- $tf(t, D) = \log[freq(t, D)] + 1$
- $tf(t, D) = freq(t, d) / \text{Max}[f(t, d)]$
- $idf(t) = \log(N/n_i)$

n_i = #docs contenant t

N = #docs dans le corpus

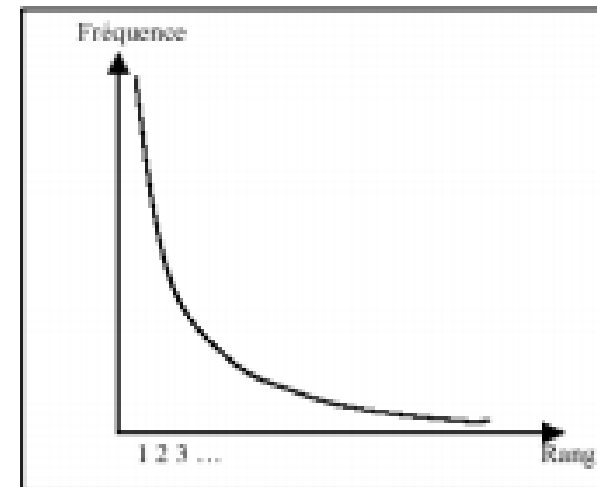
$$\text{weight}(t, D) = tf(t, D) * idf(t)$$

3. LA PONDERATION DES TERMES 3

Le choix des termes :

- **La loi de Zipf**
- Si on classe les mots dans l'ordre décroissant de leur fréquence, et on leur donne un numéro de rang (1, 2, ...), alors: $\text{Rang} * \text{fréquence} \approx \text{constante}$.

Rang	Mot	Fréquence	Rang* Fréquence
1	the	69 971	69 971
2	of	36 411	72 822
3	and	28 852	86 556
4	to	26 149	104 596
5	a	23 237	116 185
6	in	21 341	128 046
7	that	10 595	76 165



- La distribution de mots suit la courbe :
- Une idée est de garder les termes "utiles" : ni trop rares (place en mémoire), ni trop présents (pas discriminants)... choix difficile

4. LE MODEL VECTORIEL 1

Le modèle vectoriel introduit par [Salton 1975] représente chaque document, ainsi que la requête, par un vecteur et calcule un coefficient de similarité entre chaque document et la requête (appelé Retrieval Status Value ou RSV).

- L'espace vectoriel = tous les termes que le système a rencontré durant l'indexation:

$$\langle t_1, t_2, t_3, \dots, t_n \rangle$$

- Document

$$D = \langle a_1, a_2, a_3, \dots, a_n \rangle$$

a_i = La pondération de t_i dans D

- Requête

$$Q = \langle b_1, b_2, b_3, \dots, b_n \rangle$$

b_i = La pondération de t_i dans Q

- $R(D, Q) = \text{Sim}(D, Q)$

4. LE MODEL VECTORIEL 2

La matrice d'incidence :

Document de
l'espace vectoriel



D_1

D_2

D_3

\dots

D_m

Q

t_1

t_2

t_3

\dots

t_n



Termes de
l'espace
vectoriel

a_{11}

a_{12}

a_{13}

\dots

a_{1n}

a_{21}

a_{22}

a_{23}

\dots

a_{2n}

a_{31}

a_{32}

a_{33}

\dots

a_{3n}

a_{m1}

a_{m2}

a_{m3}

\dots

a_{mn}

b_1

b_2

b_3

\dots

b_n

4. LE MODEL VECTORIEL 3

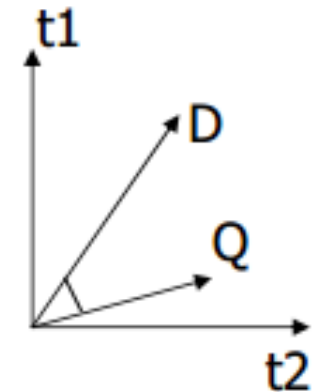
Les Mesures de Similarités :

Produit interne

$$Sim(d, q) = \sum (a_i * b_i)$$

Cosinus

$$Sim(d, q) = \frac{\sum_i (a_i * b_i)}{\sqrt{\sum_i a_i^2 * \sum_i b_i^2}}$$



Dice

$$Sim(d, q) = \frac{2 \sum_i (a_i * b_i)}{\sum_i a_i^2 + \sum_i b_i^2}$$

Jaccard

$$Sim(d, q) = \frac{\sum_i (a_i * b_i)}{\sum_i a_i^2 + \sum_i b_i^2 - \sum_i (a_i * b_i)}$$

4. LE MODEL VECTORIEL 4

Exemple : calcul de tf, idf et tfidf.

- Soit les 2 documents suivants :
 - Japan makes smart robot.
 - China is near to japan and japan is near to South Korea.
- $Tf(japan, d1) = \log(1) + 1 = 1.$
- $Tf(japan, d2) = \log(2) + 1 = 1.3.$
- $Idf(japan) = \log(2/2) = 0.$
- $Idf(china) = \log(2/1) = 0.3.$
- $tfidf(japan, d2) = tf * idf = 0.$
- $Tfidf(china, d1) = 0.3.$

4. LE MODEL VECTORIEL 5

Exemple : calcul de similarité par la formule cosinus.

Soit le document et la requête suivants :

D1 : (0.2, 0, 1, 0).

Q : (0, 0, 1 , 1).

Sim(d1,q)=

cosinus (d1,q)=

$$(0.2*0+0*0+1*1+0*1) / [(0.2^2+1^2)*(1^2+1^2)]^{1/2}$$

$$=1/1.44=\mathbf{0.69}.$$

5. Le model probabiliste

Il existe différentes manières de voir une approche probabiliste de la recherche d'information

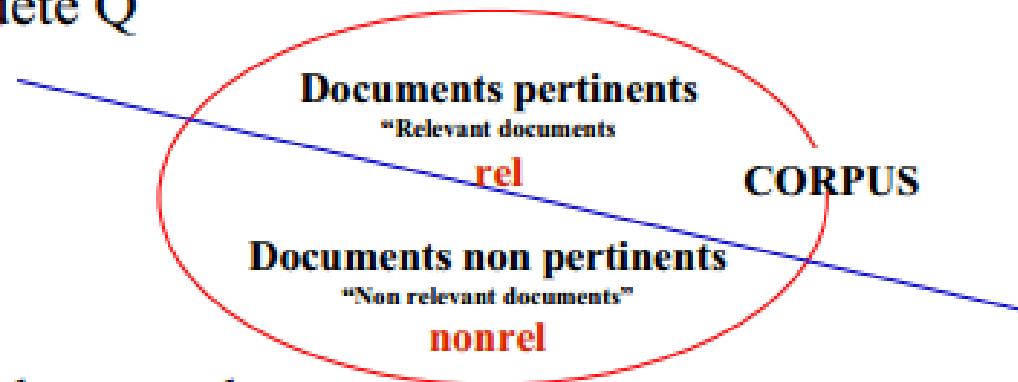
- **Approche classique** : probabilité d'avoir l'événement Pertinent sachant un document et une requête.
- **Approche par réseaux d'inférence** : probabilité que la requête soit vraie d'après une inférence à partir du contenu du document.
- **Approche par modèle de langue** : probabilité qu'une requête posée soit générée à partir d'un document.

5. Le model probabiliste 1

- Le modèle probabiliste **classique** consiste à calculer la pertinence d'un document en fonction de pertinences connues pour d'autres documents.
- Ce calcul se fait en estimant la pertinence de chaque index pour un document et en utilisant le **Théorème de Bayes** et une **règle de décision**.

5. Le model probabiliste 2

– Pour un requête Q



Avec

$$\text{Corpus} = \text{rel} \cup \text{nonrel}$$

$$\text{rel} \cap \text{nonrel} = \emptyset$$



P (pertinence_Q / document D_i)
notée simplement P(rel/ D_i)

Probabilité pour que le document i fasse partie de l'ensemble des documents pertinents à la requête Q

5. Le model probabiliste 3

Fonction de correspondance :

- Utilisation du théorème de Bayes

Probabilité d'obtenir la description D_i des pertinences observées

$P(rel)$ est la probabilité de pertinence, c'est-à-dire si l'on choisit dans le corpus un document au hasard, c'est la chance de tomber sur un document pertinent à la requête Q

$$P(rel / D_i) = \frac{P(D_i / rel).P(rel)}{P(D_i)}$$

Probabilité pour que le document i fasse partie de l'ensemble des documents pertinents à la requête Q

probabilité que le document i soit choisi

5. Le model probabiliste 4

Décision : document retourné si

$$\frac{P(rel / D)}{P(nonrel / D)} = \frac{P(D / rel).P(rel)}{P(D / nonrel).P(nonrel)} > 1$$

- En RI, on cherche un ordre, donc on peut éliminer $Prob(rel)/Prob(nonrel)$ pour une requête donnée.
- En RI, on passe par les LOG :

$$g(D) = \log\left(\frac{P(D / rel)}{P(D / nonrel)}\right)$$

5. Le model probabiliste 5

Avec les notations :

- $p_i = P(x_i=1 / \text{rel})$ alors $P(x_i=0 / \text{rel}) = 1 - p_i$
- $q_i = P(x_i=1 / \text{nonrel})$ alors $P(x_i=0 / \text{nonrel}) = 1 - q_i$

$$g(D) = \log\left(\prod_{x_i=1} \frac{p_i}{q_i}\right) + \log\left(\prod_{x_i=0} \frac{1-q_i}{1-p_i}\right)$$

- On obtient :

$$g(D) \propto \sum_{t_i \in D \cap Q} \log \left(\frac{\frac{p_i}{1-p_i}}{\frac{q_i}{1-q_i}} \right)$$

5. Le model probabiliste 6

- Estimation de p_i et q_i sur un ensemble de requêtes prédéfinies

	Pertinent	Non pertinent	total
terme t_i présent	r_i	$n_i - r_i$	n_i
terme t_i absent	$R_i - r_i$	$N - n_i - (R_i - r_i)$	$N - n_i$
total	R_i	$N - R_i$	N

- Avec
 - R_i : nombre de documents pertinents à une requête contenant un terme t_i
 - N : nombre de documents du corpus
 - r_i : nombre de documents pertinents contenant le terme t_i
 - $n_i - r_i$: nombre de documents non pertinents contenant le terme t_i

5. Le model probabiliste 7

- Estimation de p_i et q_i sur un ensemble de requêtes prédéfinies

	Pertinent	Non pertinent	total
terme t_i présent	r_i	$n_i - r_i$	n_i
terme t_i absent	$R_i - r_i$	$N - n_i - (R_i - r_i)$	$N - n_i$
total	R_i	$N - R_i$	N

- On obtient

$$p_i = \frac{r_i}{R_i} \qquad 1 - p_i = \frac{R_i - r_i}{R_i}$$

$$q_i = \frac{n_i - r_i}{N - R_i} \qquad 1 - q_i = \frac{N - R_i - n_i + r_i}{N - R_i}$$