RECHERCHE D'INFORMATION & TRAITEMENT AUTOMATIQUE DU LANGAGE

Cours 2 : RI - modèles d'appariement

Monday 4th February, 2019

Laure Soulier

Machine Learning & Deep Learning for Information Access

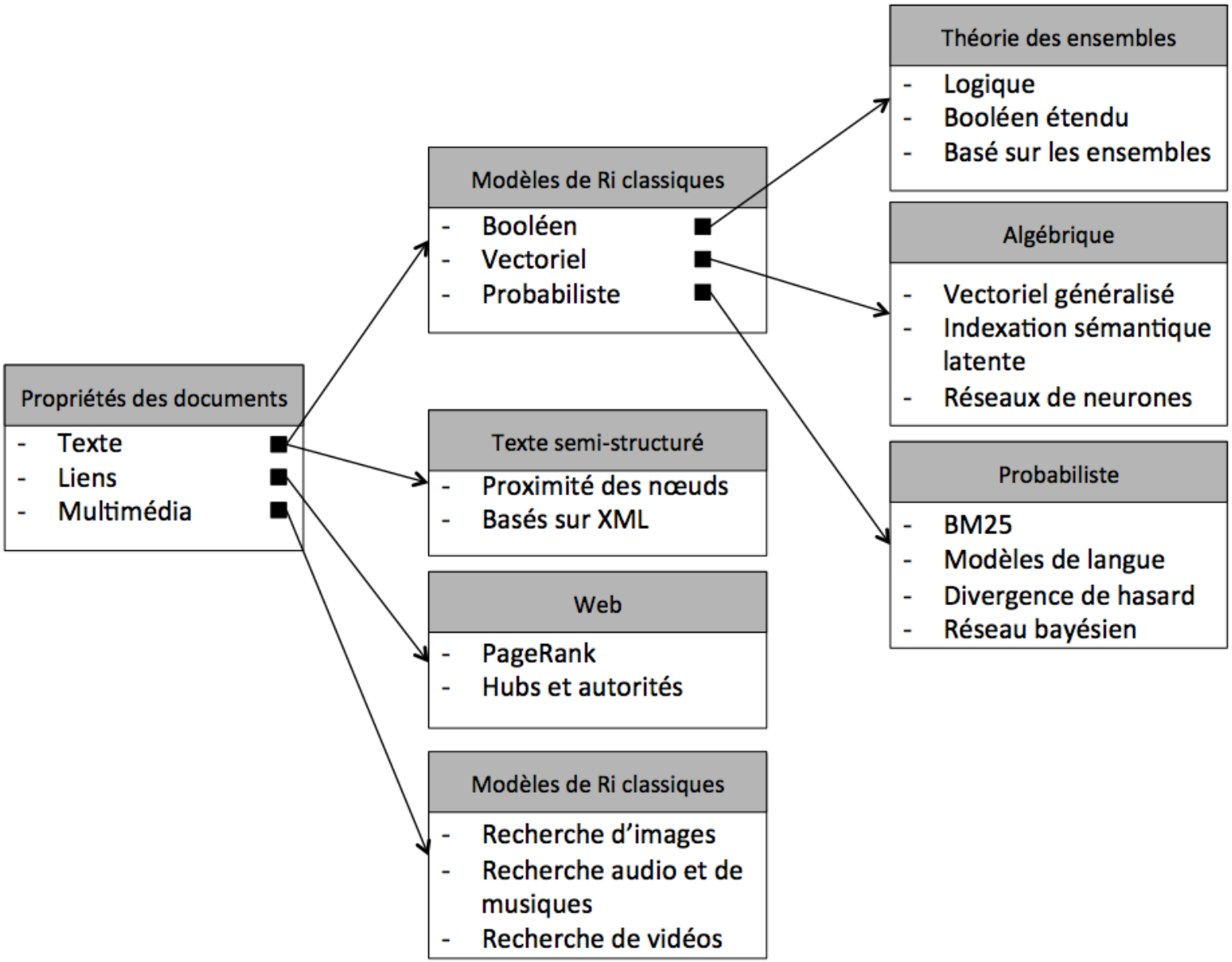
Modèles de recherche

Hypothèses

- Plus la requête et le document ont de mots en commun, plus grande sera la pertinence du document

- Plus la requête et le document ont une distribution de termes similaire, plus grande sera la pertinence du document

Familles de modèles



Modèle booléen

Modèle booléen (Salton, 1971)

- Modèle pionnier
- Basé sur la théorie des ensembles
- Représentation logique des documents L(d) et des requêtes L(q) en utilisant les opérateurs logiques : OU ($\vee$), ET ($\wedge$) et NON ($\neg$).
- Exemple :
  - q = t1 $\wedge$ ($\neg$ t2 $\vee$ t5)
  - d1(t1,t3,t5); d2(t1,t3,t5); d3(t1,t2,t3,t4)
- Score de similarité :

$$RSV(q,d) = \begin{cases} 1 & \text{si L(q)} \subset \text{L(d)} \\ 0 & \text{sinon} \end{cases} \quad (1)$$

Inconvénients du modèle booléen

- Pas de pondération de l'importance des termes
- Score de similarité binaire
- Pas d'ordonnancement possible entre les documents sélectionnés
- Risque de sélectionner beaucoup (trop) de documents, surtout lorsque la collection de documents est volumineuse
- Requête peut être difficile à formuler par l'utilisateur

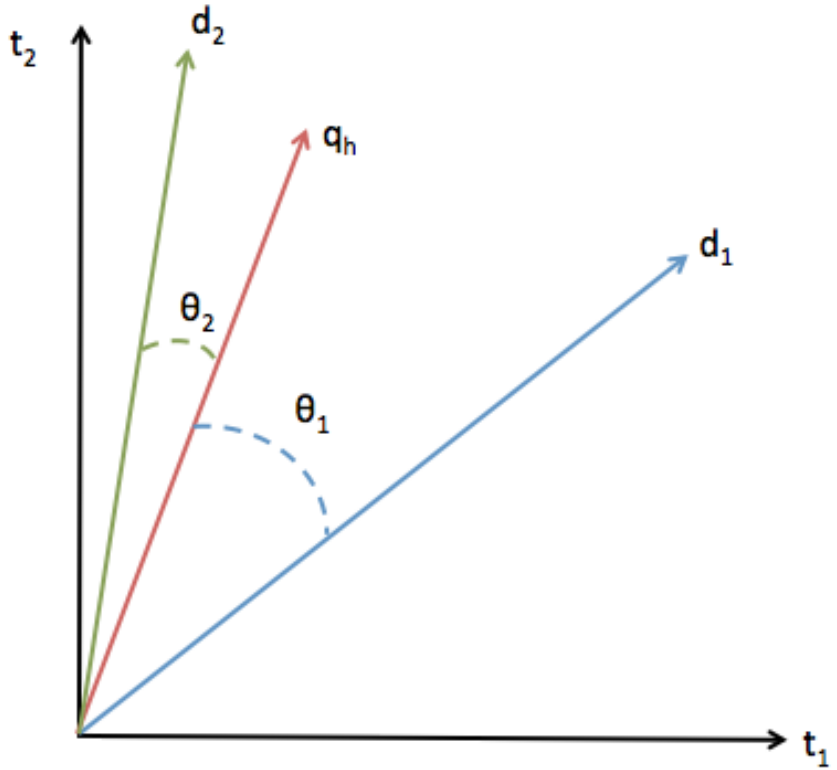
Extensions

Extensions avec une considération pondérée des poids des termes :
- modèle booléen étendu (Salton and McGill, 1983)
- modèle des ensembles flous (Ogawa et al., 1991)

Modèle vectoriel

Modèle vectoriel (Salton et al., 1975)

- Espace de caractéristiques ti, i = 1...n i.e. termes sélectionnés pré-traités
- Représentation des documents - requêtes : vecteur de poids dans l'espace des caractéristiques :
  - document : $d = (x_0, ..., x_{n-1})$
  - requête : $q = (y_0, ..., y_{n-1})$



Pondération des vecteurs de représentation

SCIENCES SORBONNE UNIVERSITÉ

- x_k poids de la caractéristique k dans le document d, e.g.:
  - présence-absence,
  - fréquence du terme dans le document, dans la collection (cf. idf), le plus répandu : tf*idf
  - importance du terme pour la recherche
  - facteurs de normalisation (longueur du document)

- Les mots sont supposés indépendants ⚠

Modèle vectoriel : mesures de similarités

- Différentes fonctions de score peuvent être employées avec un codage fréquentiel des documents :

Inner Product	$\|\|X \cap Y\|\|$
Coef de Dice	$\frac{2*\|\|X \cap Y\|\|}{\|\|X\|\|+\|\|Y\|\|}$
Mesure de cosinus ***	$\frac{\|\|X \cap Y\|\|}{\|\|X\|\|^{1/2}+\|\|Y\|\|^{1/2}}$
Mesure de Jaccard	$\frac{\|\|X \cap Y\|\|}{\|\|X\|\|+\|\|Y\|\|-\|\|X \cap Y\|\|}$

Modèle vectoriel : avantages et inconvénients

SCIENCES SORBONNE UNIVERSITÉ

- Avantage par rapport au modèle booléen
  - Les termes sont pondérés
  - Les documents sont évalués sur une échelle continue → Permet la sélection de documents partiellement pertinents

- Inconvénients
  - hypothèse d'indépendance des termes + ne tient pas compte de l'ordre des mots ("sac-de-mots" / "bag-of-words")
    → Extension : prise en compte des N-grammes (Song and Croft, 1999)
  - similarité != pertinence. Le document le plus similaire peut-être non pertinent
  - initialement conçu pour des documents courts.
    → Documents longs: facteurs de normalisation, approches hiérarchiques par paragraphes (sélection de paragraphes pertinents + combinaison des scores des paragraphes)

Modèle probabiliste

Modèle probabiliste

- Hypothèses
  - Un paire (d,q) est la réalisation d'un tirage aléatoire dans un espace document*requête
  - A chaque paire, on associe une variable aléatoire binaire R qui vaut 1 si d est pertinent et 0 sinon

- Probability Ranking Principle (Robertson 1977)
  - Présenter les documents à l'utilisateur selon l'ordre décroissant de leur probabilité de pertinence P(R=1|d,q)
  - Propriété : principe optimal car il optimise le risque de Bayes pour la règle de décision suivante : d est pertinent ssi $P(R=1|d,q) > P(R=0|d,q)$

Binary independent model

- C'est le modèle de base associé à ce principe
  - (1) d et q sont représentés comme des vecteurs binaires présence/absence de termes
  - (2) Les termes dans les documents et les requêtes sont indépendants (sacs de mots)
  - (3) Les termes non présents dans la requête sont uniformément répartis dans l'ensemble des documents pertinents et non pertinents pour la requête

Binary independent model

- Score de similarité : rapport des probabilités a posteriori (règle de bayes)

$$
\begin{aligned}
p(R|q,d) &\overset{q}{=} \frac{p(R=1|q,d)}{p(R=0|q,d)} \\
&= \frac{p(d|R=1,q)\,p(R=1|q)}{p(d|R=0,q)\,p(R=0|q)} \\
&\overset{q}{=} \frac{p(d|R=1,q)}{p(d|R=0,q)}
\end{aligned}
$$

- Note: on pourrait aussi avoir (**autre modèle**)

$$\frac{p(q|R=1,d)\,p(R=1|d)}{p(q|R=0,d)\,p(R=0|d)}$$

Binary independent model

- En utilisant l'hypothèse d'indépendance des termes

$$p(R|d,q) = \prod_{i:x_i=1} \frac{P(x_i=1|R=1,q)}{P(x_i=1|R=0,q)} \prod_{i:x_i=0} \frac{P(x_i=0|R=1,q)}{P(x_i=0|R=0,q)} \cdot \frac{P(R=1|q)}{P(R=0|q)} \tag{2}$$

- Notons :
  - $p_i = P(x_i=1|R=1,q)$ la probabilité que le terme x_i apparaisse dans un document pertinent pour q
  - $u_i = P(x_i=1|R=0,q)$ la probabilité que le terme x_i apparaisse dans un document pertinent pour q
- On obtient :

$$p(R|d,q) = \prod_{i:x_i=1} \frac{p_i}{u_i} \prod_{i:x_i=0} \frac{1-p_i}{1-u_i} \cdot \frac{P(R=1|q)}{P(R=0|q)} \tag{3}$$

Binary independent model

- Sous l'hyptohèse 3 (On ne considère que les termes de la requête)

$$p(R|d,q) = \prod_{i:x_i=1,y_i=1} \frac{p_i}{u_i} \prod_{i:x_i=0,y_i=1} \frac{1-p_i}{1-u_i} \cdot \frac{P(R=1|q)}{P(R=0|q)}$$

$$p(R|d,q) = \prod_{i:x_i=1,y_i=1} \frac{p_i}{u_i} \frac{1-u_i}{1-p_i} \prod_{i:y_i=1} \frac{1-p_i}{1-u_i} \cdot \frac{P(R=1|q)}{P(R=0|q)}$$

- Pour une requête donnée, $\prod_{i:y_i=1} \frac{1-p_i}{1-u_i}$ et $\frac{P(R=1|q)}{P(R=0|q)}$ sont des constantes, dépendant seulement de la requête

- Score de pertinence :

$$s(q,d) = \sum_{i:x_i=y_i=1} log \frac{p_i(1-u_i)}{u_i(1-p_i)} \tag{4}$$

reprendre au tableau

BIM : Estimation par vraisemblance

- Estimation des probabilités p_i et u_i
  - Maximum de vraisemblance sur une base d'apprentissage (i.e., fréquences relatives)
  - Tableau des fréquences

	Pertinent	Non Pertinent	Total
terme $x_i = 1$	r	$n-r$	n
terme $x_i = 0$	$R-r$	$N-n-R+r$	$N-n$
total	R	$N-R$	N

    * r = nombre de documents pertinents contenant terme xi
    * Avec ces fréquences :

$$p_i = \frac{r}{R}; u_i = \frac{n-r}{N-R} \tag{5}$$

    * En pratique, on lisse ces fréquences pour éviter les 0 (facteur β : r+β, etc. . .)

BIM : Estimation par vraisemblance

Exercice

Que vaut le score de similarité lorsqu'on remplace p_i et u_i par les valeurs de fréquences ?

$$s(q,d) = \sum_{i:x_i=y_i=1} log \frac{p_i(1-u_i)}{u_i(1-p_i)} \tag{6}$$

BIM : Estimation par vraisemblance

Exercice

Que vaut le score de similarité lorsqu'on remplace p_i et u_i par les valeurs de fréquences ?

$$s(q,d) = \sum_{i:x_i=y_i=1} log \frac{p_i(1-u_i)}{u_i(1-p_i)} \tag{6}$$

$$s(q,d) = \sum_{i:x_i=y_i=1} \log \frac{r+0.5}{R-r+0.5} \times \frac{N-n-R+r+0.5}{n-r+0.5}$$

Modèle probabilistes

- Nombreuses variantes / extensions
  - longueur des documents (hypothèse implicite d'égale longueur)
  - expansion des requêtes
  - doc pertinents considérés (e.g. cas recherche on line ≠ off line)
  - cooccurrence de termes, prise en compte de "phrases" ...

Okapi - Robertson et al.

- Modèle de référence en RI :
  - inspiré du modèle BIM
  - les probabilités de pertinence/non-pertinence sont calculées à partir de lois 2-Poissons

Okapi - Robertson et al.

- Modèle de référence en RI

- Principe
  - Un document pertinent contient des termes fréquents dans ce document mais relativement rare dans la collection
  - Notion de groupe élite : termes discriminants sont les termes qui apparaissent rarement dans beaucoup de documents mais avec une fréquence assez élevée dans un groupe distinct de documents
    * Groupes élites : Modélisation par une loi de Poisson de paramètre λ
    * Groupes non élites : modélisation par une loi de Poisson de paramètre μ $(< \lambda)$
  - Probabilité d'apparition d'un terme x_i dans un document d est exprimé par une loi de mélange suivant la distribution de ce document par rapport aux groupes élites/non élites

$$p(x_i|R=1,q) = \alpha_{x_i} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!} + (1-\alpha_{x_i}) \frac{\mu^{x_i} e^{-\mu}}{x_i!}$$

$$p(x_i|R=0,q) = \beta_{x_i} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!} + (1-\beta_{x_i}) \frac{\mu^{x_i} e^{-\mu}}{x_i!}$$

Okapi - Robertson et al.

- Fonction de score basée sur le modèle BIM sauf que les probabilités d'absence/présence sont calculées d'après les lois 2-Poisson :

$$RSV(d,q) = \sum_{x_i \in q \cap d} ln\sigma_{x_i} \tag{7}$$

avec :

$$\begin{aligned}
\sigma_{x_i} &= \frac{P(x_i = tf_i|R=1,q)}{P(x_i = tf_i|R=0,q)} \cdot \frac{P(x_i=0|R=0,q)}{P(x_i=0|R=1,q)} \\
&= \frac{\alpha_{x_i}\lambda^{x_i}e^{-\lambda} + (1-\beta_{x_i})\mu^{x_i}e^{-\mu}}{\beta_{x_i}\lambda^{x_i}e^{-\lambda} + (1-\alpha_{x_i})\mu^{x_i}e^{-\mu}} \cdot \frac{\beta_{x_i}e^{-\lambda} + (1-\beta_{x_i})e^{-\mu}}{\alpha x_i e^{-\lambda} + (1-\alpha x_i)e^{-\mu}} \\
&\propto \frac{(\alpha_{x_i} + (1-\alpha_{x_i})(\mu/\lambda)^{x_i}e^{\lambda-\mu})(\beta_{x_i}e^{\mu-\lambda} + (1-\beta_{x_i}))}{(\beta x_i + (1-\beta x_i)(\mu/\lambda)^{x_i}e^{\lambda-\mu})(\beta_{x_i}e^{\mu-\lambda} + (1-\alpha_{x_i}))}
\end{aligned}$$

- $\mu < \lambda$: $e^{\mu-\lambda} \propto = 0$; Limites pour $x_i \rightarrow \inf$

$$lim_{x_i \rightarrow \inf}\ \sigma_{x_i} = \frac{\alpha_{x_i}}{\beta_{x_i}} \frac{(1-\beta_{x_i})}{(1-\alpha_{x_i})} \tag{8}$$

Okapi - Robertson et al.

- Inverse document frequency (IDF)
  - Pas d'information sur la pertinence des documents

$$"idf" = log(\frac{N}{n}) \tag{9}$$

  - Information sur la pertinence des documents

$$"idf" = \log \frac{r+0.5}{R-r+0.5} \times \frac{N-n-R+r+0.5}{n-r+0.5} \tag{10}$$

Okapi - Robertson et al.

- Formule générale : BM25 (Robertson et Walker, 1994)

$$s(d,q) = \sum_{i;y_i=1} IDF(y_i).\frac{tf(y_i,d)}{tf(y_i,d) + k_1(1-b+b.\frac{|D]}{avgdl}} \quad (11)$$

avec $|D|$: longueur du document, $avgdl$: longueur moyenne des documents. k_1 et b constantes (e.g., resp 1.2 et 0.75)

- Deux modifications :
  - Verbosité
  - coefficient de saturation de la loi 2-poisson qui modélise la distribution des termes dans les documents (élites, non élites)

Modèles de langue

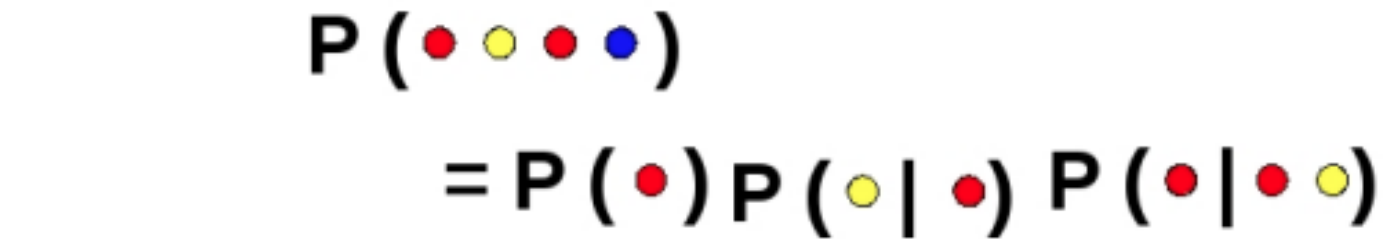
Modèles de langue (Ponte, Croft, Hiemstra, ... 98-99)

- Intuition :
  - Modélise la distribution des mots dans une langue
  - Mesure la probabilité d'observer une séquence de mots dans une langue
  - Identifie la source qui a permis de générer un texte
- Formalisation

$$Score(d, q) = P(s|\theta_M) \tag{12}$$

  - Taille des séquences ?
  - Estimer la probabilité de chaque séquence ?
  - Estimer le modèle de langue ?

Taille des séquences et probabilités



- Unigram Models (Assume word independence)

- Bigram Models

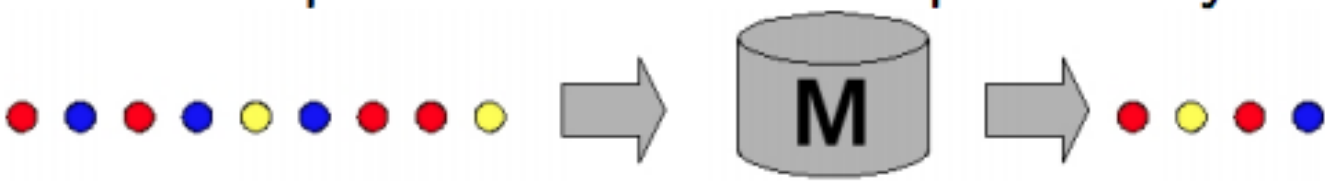
- There are others . . .

Modèle de langue du document

- Usually we do not know the model **M**, but have a sample representative of that model

P (● ○ ● ● | M (● ● ● ● ○ ● ● ● ○))

- First estimate a model from a sample
- Then compute the observation probability



Modèles de langue (Ponte, Croft, Hiemstra, ... 98-99)

- Modèle de langue multinomial
  - Indépendance des termes dans le document
  - Pas d'ordre

$$p(t_1,...,t_n) = \prod_t P(TF(t) = tf(t)|\theta_{Md})^{tf(t)} \stackrel{q}{=} \sum_t tf(t) log(P(t|\theta_{Md})) \tag{13}$$

  - Comment estimer ces probabilités ?

$$argmax_{p_t} \sum_t tf(t) log(p_t) \tag{14}$$

→ Solution avec le Lagrangien :

$$p_t = \frac{tf(t)}{\sum_t tf(t)} \tag{15}$$

Modèles de langue (Ponte, Croft, Hiemstra, ... 98-99)

- Dans le cas où un mot de la requête n'apparaît pas dans le document d
  - Score du document est égal à 0
  - En pratique, on utilise un lissage de cette probabilité : modèle de mélange multinomial entre la distribution des termes dans le document et la distribution des termes dans la collection
    * Jelinek-Mercer ($\lambda = 0.8$ pour les requêtes courtes et 0.2 pour les requêtes longues)

$$P(t|d) = (1 - \lambda_t)P(t|\theta_{Md}) + \lambda_t P(t|\theta_{MC}) \tag{16}$$

    * Dirichlet

$$\frac{tf(t,d) + \mu p(t|\theta_C}{length(d) + \mu} \tag{17}$$

Probabilités et Document Prior

- Rappel : $P(d|q) = P(q|d)P(d)/P(q) \propto P(q|d)P(d)$
- $P(d)$ est généralement considéré comme uniforme $\rightarrow P(d|q) \propto P(q|d)$
- $P(d)$ permet également d'intégrer des connaissances a priori dans le calcul de la probabilité :
  - Longueur du document
  - Longueur moyenne des mots
  - Date de publication : "fraîcheur"
  - Nombre de liens
  - PageRank
  - ...

Reformulation de requêtes

Reformulation de requêtes

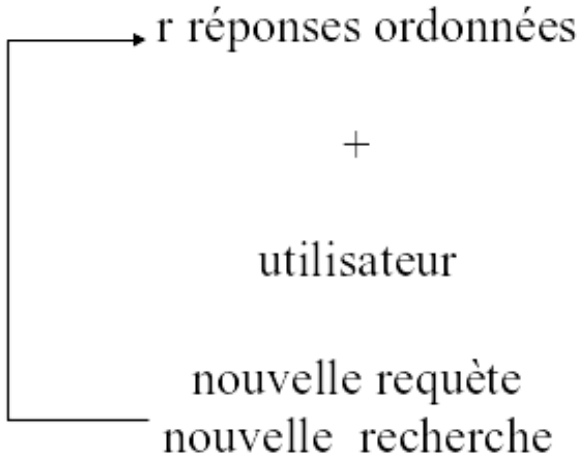
- Intuition
  - Difficile de formuler les requêtes qui correspondent aux documents de la collection
    * On ne sait pas forcément exprimer ce que l'on cherche
    * On ne sait pas forcément à quoi ressemble le document
  - La première requête est souvent naïve, permettant d'avoir une première idée de ce que l'on peut trouver. On peut alors reformuler la requête à partir des résultats :
    * Etendre la requête originale avec des nouveaux termes
    * Re-pondérer la requête (étendue

Reformulation de requêtes

- Relevance feedback : basée sur les feedbacks des utilisateurs
- Pseudo-relevance feedback : basée sur l'analyse locale des documents retournés
- Analyse globale des documents documents de la collection

Relevance feedback

- Méthode classique



  - relevance : valeurs dans $\{0, 1\}$
  - idée : utilisateur examine une partie des meilleurs documents et les étiquette 1/0
  - la requête est reformulée (enrichissement)

Relevance feedback

- Liste ordonnée des r meilleurs documents

$$D_r(q) = d_1, ..., d_r \tag{18}$$

- Partition de ces r documents par l'utilisateur

$$D_r(q) = \{D_r^{rel}(q) \cup D_r^{non-rel}(q)\} \tag{19}$$

- Principe du relevance feedback :

$$q' = f(q, D_r^{rel}(q), D_r^{non-rel}(q)) \tag{20}$$

Relevance feedback - Rocchio 1971

- Modèle de base de l'expansion/reformulation de requêtes

$$\vec{Q} = (a.\vec{Q_0}) + (b.\frac{1}{|D_rel|}\sum_{d+\in D_{rel}} \vec{d+}) - (c.\frac{1}{|D_non-rel|}\sum_{d-\in D_{non-rel}} \vec{d-}) \quad (21)$$

- Améliorations allant de 20% à 80% par rapport à sans RF
- Différentes variantes :
  - considérer seulement les documents pertinents / que les non-pertinents
  - optimiser a, b, c
  - optimiser le nombre de documents du feedback

Relevance feedback : limites

- Le feedback des utilisateurs n'est pas toujours fiable
  - Positif/négatif ?
  - Degré de pertinence ?
  - Pourquoi ce document peut être pertinent ?
  - En pratique, cela marche bien car on bénéficie de l'effet de masse ("wisdom of the crowd")

Analyse locale : pseudo-relevance feedback

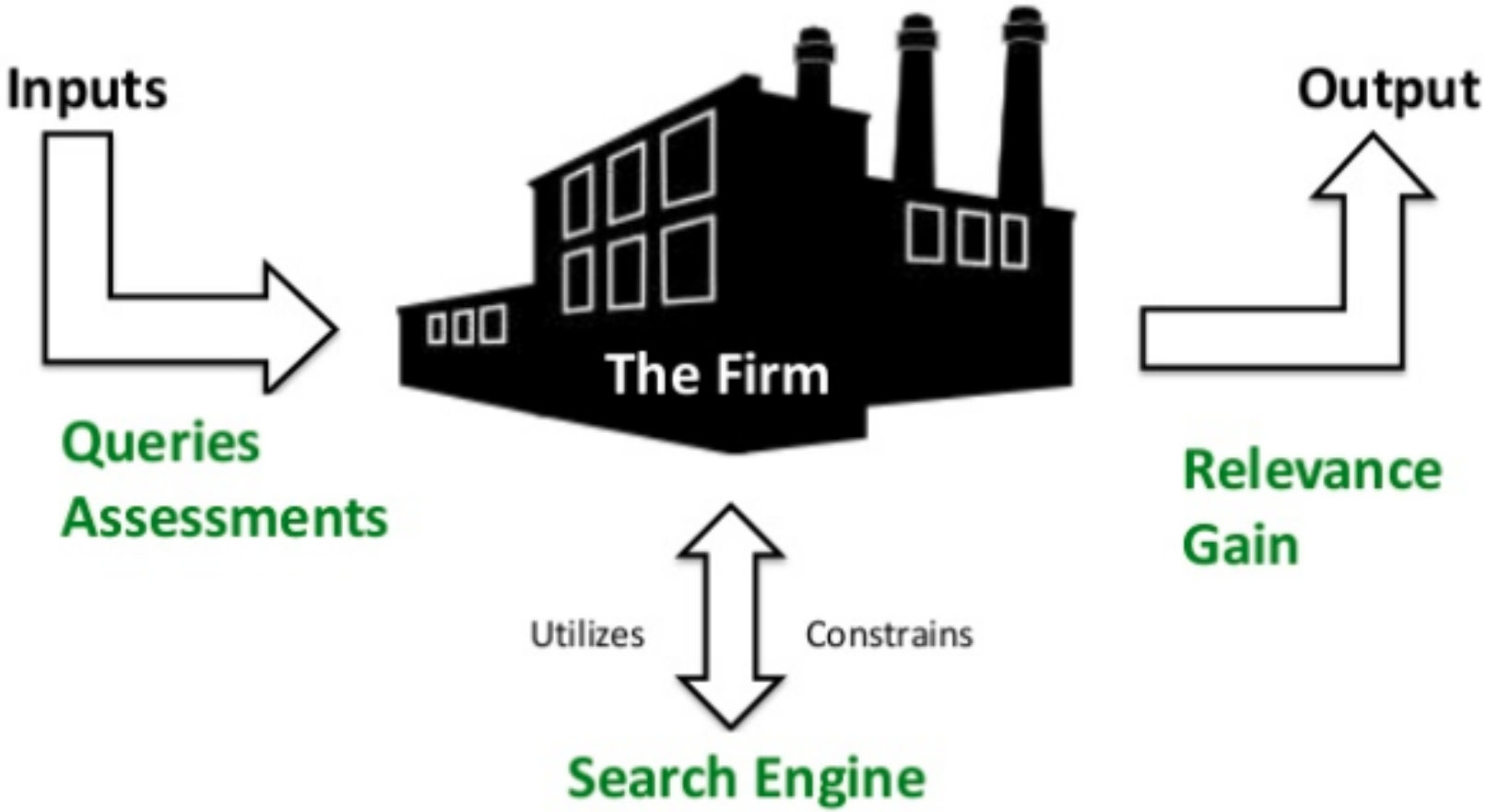
- Suggestion de requête automatique :
  - Pas besoin du feedback utilisateur : les k premiers documents sont considérés comme pertinents
- Approches :
  - Clustering
  - Similarité des terms
  - Analyse des sessions
- Problèmes
  - le système va fournir des documents similaires à ceux déjà trouvés...
  - "Query drift": si les top documents ne sont pas pertinents, la requête reformulée ne reflètera jamais le besoin de l'utilisateur
  - Peut s'avérer coûteux en termes d'exécution

Analyse globale

- Principe :
  - Etendre la requête à partir de la collection
  - Pas d'assistance de l'utilisateur

  Approche : construire le thésaurus des co-occurrences des termes pour identifier les termes les plus proches de ceux de la requête

Un autre modèle - Leif Azzopardi



Un autre modèle - Leif Azzopardi

Let the gain the user receives through their interaction be:

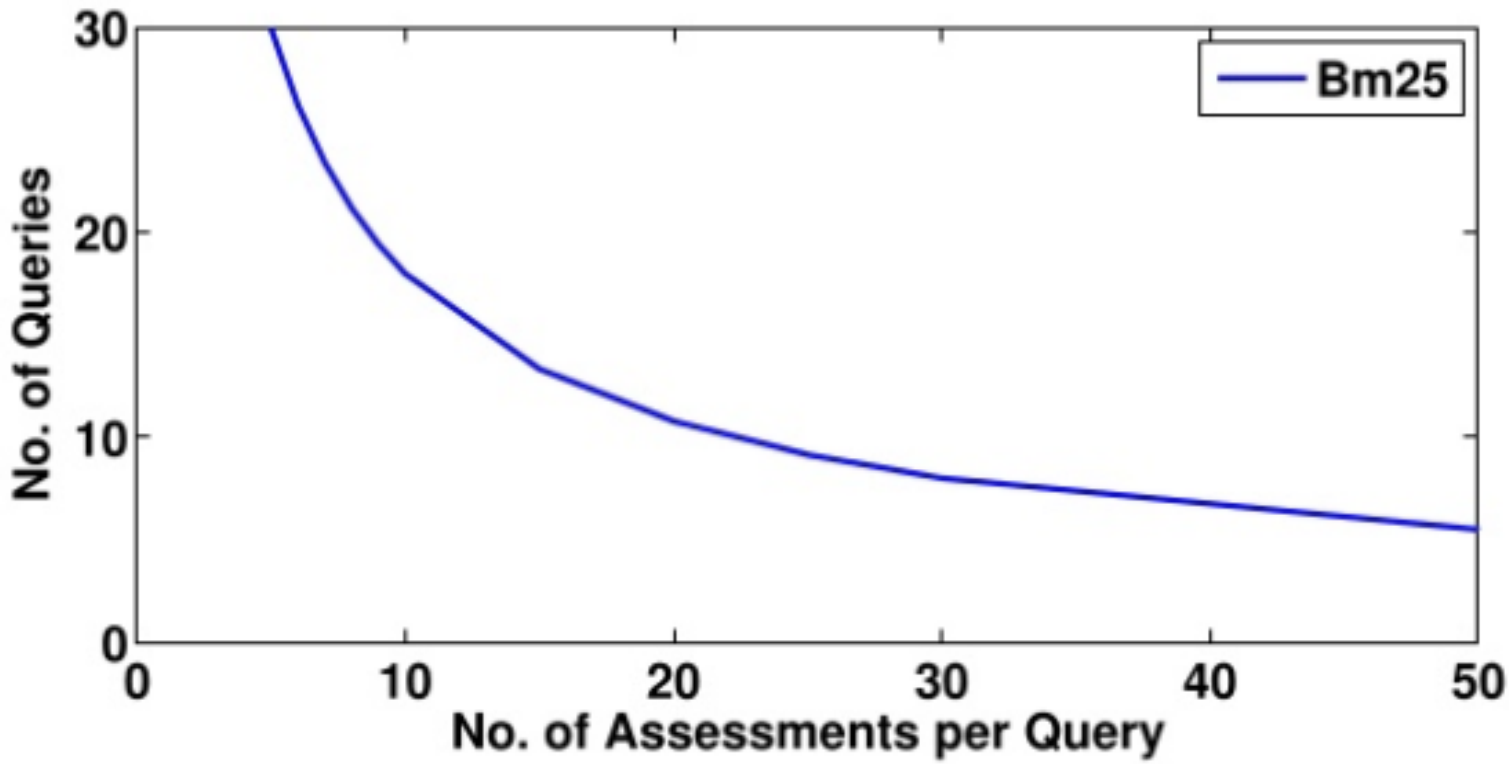
$$g(Q, A) = k.Q^{\alpha}.A^{(1-\alpha)}$$

Where:

Q is the number of queries, and
A is the number of documents examined per query.
α is the relative efficiency of querying to assessing
k is the efficiency of the technology/user to extract/ identify relevant information returned

Azzopardi (2011)

Un autre modèle - Leif Azzopardi



Each point on the curve represents a combination of interactions that will yield the same gain.

Un autre modèle - Leif Azzopardi

The total cost can be calculated by:

$$c(Q, A) = c_q.Q + c_a.A.Q$$

Where:

- c_q is the cost of a query
- c_a is the cost of a assessing a document
- $A.Q$ is the total number of documents assessed

Azzopardi (2011)

Un autre modèle - Leif Azzopardi

- Given our model, we wish to minimize the cost **c(Q,A)**, subject to the constraint that **g(Q,A) = g**
- To do this we used a Lagrangian multiplier

$$\Delta = (\mathbf{c_q} + \mathbf{c_v}.\mathbf{v}).\mathbf{Q} + \left(\frac{\mathbf{c_s}}{\mathbf{p_a}} + \mathbf{c_a}\right).\mathbf{A}.\mathbf{Q} - \lambda\left(k.Q^{\alpha}.A^{\beta} - g\right)$$

Un autre modèle - Leif Azzopardi

The optimal number of assessments per query:

$$\mathbf{A}^{\star} = \frac{\beta.(\mathbf{c_q} + \mathbf{c_v}.\mathbf{v})}{(\alpha - \beta).(\frac{\mathbf{c_s}}{\mathbf{p_a}} + \mathbf{c_a})}$$

The optimal number of queries:

$$\mathbf{Q}^{\star} = \sqrt[\alpha]{\frac{\mathbf{g}}{\mathbf{k}.\mathbf{A}^{\beta}}}$$

A vous de jouer... TD is coming !