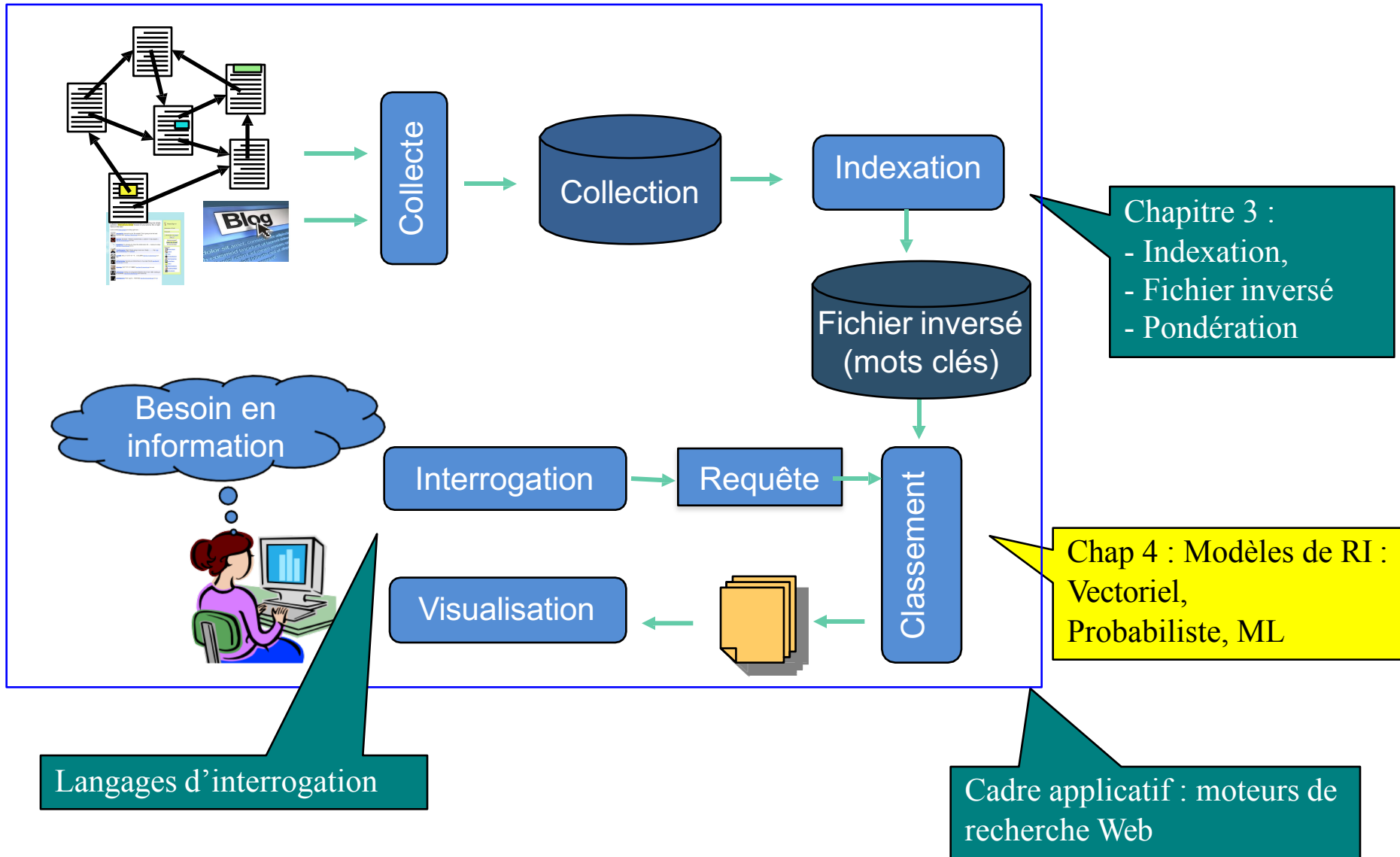


Chapitre 5 : Modèles de RI



Qu'est ce qu'un modèle de RI ?

- But : formalisation de la fonction de pertinence
- Un modèle de recherche spécifie les détails de
 - Représentation du document
 - Représentation de la requête
 - Définition de la pertinence
 - à Implique la notion de tri (classement) des documents
 - Tous les modèles ne trient pas les documents
 - Exact match (appariement exact) versus Best match(meilleur appariement)

Qu'est ce qu'un modèle de RI ?

- Exact match versus Best Match
 - Exact Math
 - Requête spécifie de manière précise les critères recherchés
 - L'ensemble des documents respectant exactement la requête sont sélectionnés, mais pas ordonné
 - Best matching (Ranking based models)
 - Requête décrit les critères recherchés dans un document
 - Les documents sont sélectionnés selon un degré de pertinence (similarité/ probabilité) vis-à-vis de la requête et sont ordonnés

IR models

- Appariement exact :
 - Théorie des ensembles :
 - Boolean model (± 1950)
- Modèle de tri de documents : Ranked models
 - Algèbre
 - Vector space model (± 1970)
 - LSI (Latent semantic Indexing)(± 1994)
 - Probabilité
 - Probabilistic model (± 1976)
 - Inference network model (± 1992)
 - Language model (± 1998)
 - *DFR (Divergence from Randomness model)* (± 2002)
 - Modèles neuronaux(Neural Model)

Appariement exact/Exact matching : Modèle booléen/Boolean Model

Le Modèle Booléen

- Le premier modèle de RI
- Basé sur la théorie des ensembles
- Représentation des documents et des requêtes :
 - Un document est représenté un ensemble de termes
 - Ex : $d1(t1,t2,t5)$; $d2(t1,t3,t5,t6)$; $d3(t1,t2,t3,t4,t5)$
 - Une requête est un ensemble de mots avec des opérateurs booléens : AND (\wedge), OR (\vee), NOT (\leftarrow)
 - Ex: $q = t1 \wedge (t2 \vee \leftarrow t3)$
- Appariement Exact basé sur la présence ou l'absence des termes de la requête dans les documents
 - Appariement $(q,d) = RSV(q,d)=1$ ou 0

- $q = t1 \wedge (t2 \vee \leftarrow t3)$
- $d1(t1,t2,t5); d2(t1,t3,t5,t6); d3(t1,t2,t3,t4,t5)$

Rsv(q,d1)=

Rsv(q,d2)=

Rsv(q,d3)=

Inconvénient du Modèle Booléen

- La sélection d'un document est basée sur une décision binaire
- Pas d'ordre pour les documents sélectionnés
- Formulation de la requête difficile pas toujours évidente pour beaucoup d'utilisateurs
- Problème de collections volumineuses : le nombre de documents retournés peut être considérable

Modèles de tri/ Rank-based models

Modèles de tri

- Les modèles de tri retournent les documents dans un ordre trié censé représenter la pertinence de la requête vis-à-vis du document.
- Requêtes en texte libre: l'utilisateur exprime son besoin en fournissant au moteur de recherche une liste de mots clés
- Dans ces modèles on calcule un score de **pertinence**:
 $\text{RSV}(\text{requête}, \text{document})$


Modèle de tri : modèle formel

- Dans la majorité de ces modèles, même si le cadre théorique diffère,
 - Requête et document sont représentés dans l'espace du vocabulaire (représentation vectorielle, sac de mots; word embeddings)
 - $d(w_1, w_2, \dots, w_n)$
 - $q(q_1, q_2, \dots, q_n)$
- Le score de pertinence
 - $RSV(q, d) = \sum(q_i * w_i)$

Les modèles de RI se différencient clairement dans leur manière d'interpréter la notion de pertinence à ceci influence les poids des termes

Modèle Vectoriel (*Vector Space Model*) (VSM)

- Proposé par Salton dans le système SMART (Salton, G. 1970)
- Documents et requêtes sont représentés sous forme vectorielle
- La pertinence est interprétée comme une similarité vectorielle (c'est ce que nous traitons depuis le début de cet enseignement)

$$score(q, d) = \sum_{t \in q} w(t, q) \cdot w(t, d)$$


Tf*idf

Lapondération se base sur tf.idf

Mais possède (exploite) plusieurs variantes

Term frequency		Document frequency		Normalization	
n (natural)	$tf_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(tf_{t,d})$	t (idf)	$\log \frac{N}{df_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2 + w_2^2 + \dots + w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times tf_{t,d}}{\max_t(tf_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N - df_t}{df_t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } tf_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^\alpha$, $\alpha < 1$
L (log ave)	$\frac{1 + \log(tf_{t,d})}{1 + \log(\text{ave}_{t \in d}(tf_{t,d}))}$				

Une variante est identifiée par un nom d'attribut pour chaque colonne (un tf, un idf, une normalisation)

Une pondération de type lnc à logarithme pour tf, pas d'idf, normalisation cosine

Une pondération de type ltc à logarithme pour tf, idf et cosine

Dans le modèle vectoriel on aura ce type de notation :

ddd.qqq (ddd pour le document, qqq pour la requête)

$$score(q, d) = \sum_{t \in q} w(t, q) \cdot w(t, d)$$

Suite exemple

- ddd.qqq=inc.ltc.

$$RSV(q, d) = \sum_{t \in q} \frac{(1 + \log(tf_{t,d})) \cdot (1 + \log(tf_{t,q})) \cdot idf(t)}{\sqrt{(\sum_{t \in d} (1 + \log(tf_{t,d}))^2 * \sum_{t \in q} (1 + \log(tf_{t,q})) \cdot idf(t))^2}}$$

Norme des vecteurs q et d

$$idf(t) = \log \frac{N}{df(t)}$$

- $df(t)$: nombre de documents contenant le terme t
- N : le nombre de documents dans la collection

Exemple Inc.ltc

Document: car insurance auto insurance

Query: best car insurance

Terme	Req (ltc)						Document(Inc)				Prod
	freq	tf	nd	idf	w(t,q)	Nor.li satio n	freq	tf-	w(t,d)	n'lisa tion	
auto	0	0	5000				1	1			
best	1	1	50000				0	0			
car	1	1	10000				1	1			
insurance	1	1	1000				2	1.3			

$N=10^6$ documents

Score (q,d) = 0.8

Modèle sac de mots

- La représentation vectorielle ne tient pas compte de l'ordre des mots
 - « Un garçon manque une pomme » est représenté par le même vecteur que « une pomme mange un garçon »
 - à c'est ce que l'on appelle « Sac de mots » (Bag of words)

Modèles probabilistes pour la recherche d'information

Modèle probabiliste

- Le modèle probabiliste tente d'estimer la probabilité d'observer des événements liés au document et à la requête
- Plusieurs modèles probabilistes, se différencient selon
 - Les événements qu'ils considèrent
 - $P(\text{pert}/d, q)$: probabilité de pertinence de d vis à vis de q
 - $P(q, d)$
 - $P(q|d)$
 - $P(d|q)$
 - Les distributions (lois) qu'ils utilisent

Plusieurs modèles de RI basés sur les probabilités

*Modèle probabiliste
classique*

Modèle inférentiel

Modèle de langue

BIR

2-Poisson

Inquery

*Modèle de
croyances*

Unigram

Ngram

*Tree
Depend.*

BM25.

DFR

La forme (finale) de BM25

$$RSV(q, d) = \sum_{t \in q} \frac{(k_1 + 1)tf_{t,d}}{k_1((1 - b) + b \frac{dl}{avg(dl)} + tf_{t,d})} \log \frac{N}{df(t)}$$

- k_1 est entre 1.2–2 et b autour de 0.75
- dl : taille du document (en nombre de mots ou en octets)
- $Avg(dl)$: taille moyenne des documents
- $tf_{t,d}$: fréquence du terme t dans le document d
- $df(t)$: nombre de documents contenant le terme t
- N : le nombre de documents dans la collection

Une des formules de calcul de poids des termes à donc $RSV(q,d)$ - les plus performantes et les plus utilisées dans le domaine de la RI