Variations axiomatiques pour la recherche d'information personnalisée

Philippe MULHEM, Nawal OULD AMER, Mathias GÉRY

Contexte

- Recherche d'Information Personnalisée :
  - Deux utilisateurs ;
  - Une même requête ;
  - ➔ un même besoin d'information ?
- Axiomatisation de la RI [Fang et al. 2004] :
  - Formaliser le comportement attendu des modèles de RI.
  - Contraintes heuristiques.
- Axiomatisation de la RI Personnalisée ?

Plan

- Introduction
- Objectifs
- État de l'art
  - Recherche d'Information Personnalisée
  - Axiomatisation de la Recherche d'Information
- Proposition :
  - Axiome pour la RI Personnalisée : CERP
  - Instanciation
- Expérimentations
- Conclusion

Objectifs

- Proposition :
  - Un axiome simple (contrainte) pour la RI Personnalisée ;
  - Expansion de requête personnalisée ;
  - Une instanciation.
- Étudier l'impact des propositions :
  - Dans le cadre de tagging de documents social ;
  - Expérimenter plusieurs configurations.

État de l'art – RI Personnalisée

- RI Personnalisée : nombreux paramètres.
  - Modèle de l'utilisateur (profil) : centres d'intérêts, comportement, historique, etc.
  - Données : logs, documents générés (articles, tweets, commentaires, etc.), annotations (tags, documents), relations utilisateurs, etc.
  - Personnalisation : expansion de requête, réordonnancement des résultats, etc.

➔ Difficile de savoir ce qu'il se passe !

État de l'art – Axiomatisation de la RI

- Axiomes : comportement attendu d'un modèle de RI.
- Travail initial [Fang et al. 2004] :
  - 7 heuristiques : TF, IDF, longueur de documents.
  - Exemple : Term Frequency Constraint (**TFC1**) :

> Intuition : « Toutes choses étant égales par ailleurs, un document ayant un plus grand nombre d'occurrence d'un terme de la requête doit obtenir un score plus élevé ».

- Développements :
  - Extensions : relations terme-terme, pseudo-relevance feedback, etc.
  - Amélioration des modèles classiques (**BM25**, [Fang et al. 2011]).

➔ Et pour la RI Personnalisée ?

Proposition : axiome **CERP**

- Expansion de requête personnalisée :
  - Utilisateur u : $Profil(u)$.
  - Relations terme-terme : $R_u(w,w')$.
  - Requête « personnalisée » : $q_u = q \cup q_{exp}$

Intuition : « Un document qui contient un terme relié au profil de l'utilisateur doit être retourné avant les documents qui ne contiennent pas ce terme ».

Contrainte d'Expansion de Requête Personnalisée (CERP) : Posons une requête $q = \{w\}$, un document d du corpus C tel que $c(w,d) > 0$, et un utilisateur u de profil $Profil(u)$.

Si $\exists w' \in Profil(u)$ tel que $R_u(w,w')$, et $c(w',d) > 0$,
alors pour tout $d' \in D$ tel que $c(w,d') \neq 0$ and $c(w',d') = 0$ on a :
$RSV(d,q_u) \geq RSV(d',q_u)$, avec $q_u = q \cup \{w'\}$.

Proposition : axiome **CERP** (2)

- **CERP** est non-validée par un modèle de langue classique avec lissage de Dirichlet.

$$RSV(d,q) = \sum_{t \in d \cap q} [c(t,q).ln(1 + \frac{c(t,d)}{\mu.p(t|D)})] + |q|.ln(\frac{\mu}{|d| + \mu})$$

- Avec :
  - $q = \{w, w'\}$
  - documents d et d'
  - $c(w,d) = c(w',d)$
  - $|d| = |d'|$
  - $p(w|D) = p(w'|D)$
  - $c(w,d') = k * c(w,d)$

- **CERP** validée si et seulement si :

$$\frac{c(w,d)}{\mu.p(w|D)} + 2 \geq k$$

- Contre-exemple (réaliste) :
  - $c(w,d) = \mu * p(w|D)$
  - $k > 3$

Proposition : instanciation

- Correspondance (requête q quelconque) :
  - Documents : contenu + tags [Bouadjenek et al. 2013].

$$RSV(d,q) = \lambda.P(q \mid \sigma d) + (1-\lambda).P(q \mid \tau d)$$

  - Avec :
    - σd = contenu de d
    - τd = tags de d
    - $P(.\mid.)$: modèle de langue, lissage de Dirichlet.

Proposition : instanciation (2)

- Expansion de requête personnalisée :
  - Utilisateur u : $Profil(u)$.
  - Relations terme-terme : $R_u(w,w')$.
  - Requête « personnalisée » : $q_u = q \cup q_{exp}$

- Deux définitions de R_u avec le tagging $R(d,u,w)$:

$$R_{u-local}(w, w') \Leftrightarrow \exists d \in D, R(d, u, w) \wedge R(d, u, w')$$
$$R_{u-social}(w, w') \Leftrightarrow \exists u' \in u_{sn}, R_{u'-local}(w, w')$$

(u_{sn}: voisinage social de u)

Proposition : instanciation (3)

- Expansion de requête personnalisée :
  - Utilisateur u : $Profil(u)$.
  - Relations terme-terme : $R_u(w,w')$.
  - Requête « personnalisée » : $q_u = q \cup q_{exp}$

- Correspondance (requête $q_u = q \cup q_{exp}$) :
  - Fusion classique : $RSV(d,q,u) \propto RSV(d,q_u)$
  - Fusion « **adaptée** » qui valide **CERP**.

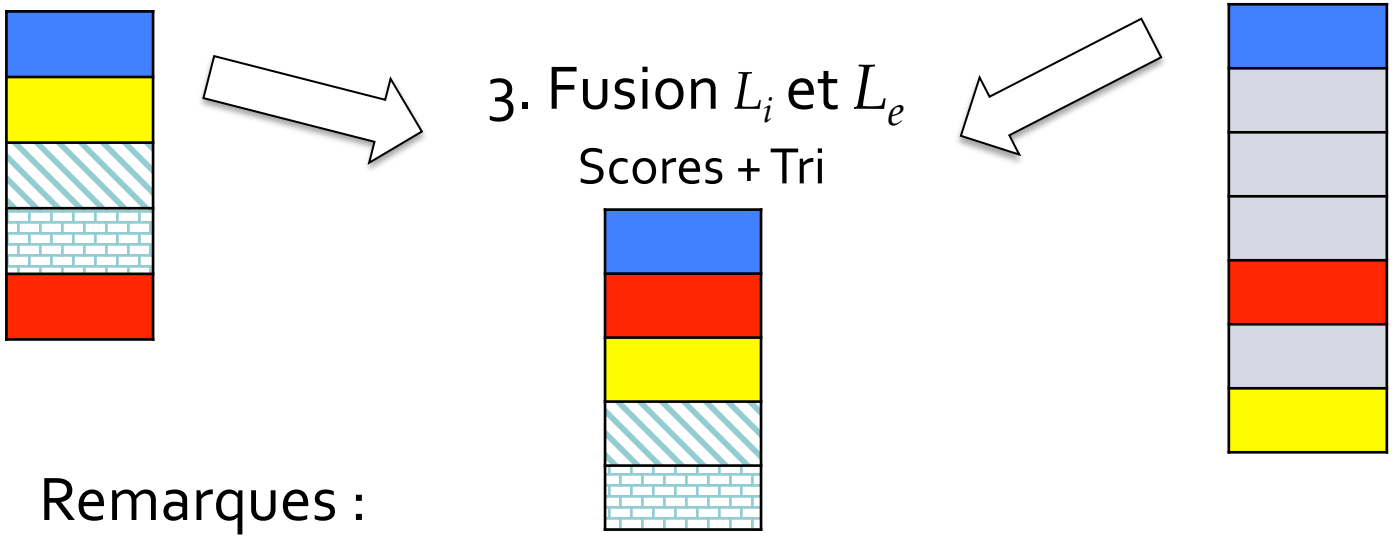
Proposition : instanciation (4)

- Approche « **adaptée** » pour valider **CERP** :

1. $RSV(d,q)$
➔ $L_i = [<d,s>]$

$$q_u = q \cup q_{exp}$$

2. $RSV(d,q_{exp})$
➔ $L_e = [<d,s>]$



3. Fusion L_i et L_e
Scores + Tri

- Remarques :
  - Priorité aux documents appartenant à L_i **et** à L_e.
  - Éliminer les documents absents de L_i.
  - Variantes *Scores* : $RSV(d,q)$, $RSV(d,q_{exp})$ ou $RSV(d,q) + RSV(d,q_{exp})$

Expérimentations : collection de test

- Corpus Bibsonomy [Benz et al. 2010] :
  - 309 Kdocs, 241 Kdocs taggés ;
  - 4,9 Kutilisateurs ;
  - 1,5 Mtags.
- Topics :
  - 200 requêtes mono-terme [Bouadjenek et al. 2010].
- Évaluation : MAP, $P@5$, $P@10$.
- Mesure additionnelle :
  - $Prof_{over}$: chevauchement entre q_u et $Profil(u)$.

Expérimentations : système

- Terrier 4.0 [Ounis et al. 2006].
- Configuration :
  - LMDir (paramètre par défaut $\mu = 2500$).
  - Anti-dictionnaire, troncature de Porter.
- Équilibre contenu / tags des documents :
  - $\lambda = 0,5$ [Bouadjenek et al. 2010].

Expérimentations : configurations

- **(1)** Baseline : sans expansion.
- **(4)** Fusions $q_u = q \cup q_{exp}$:
  - (**1**) Classique : $RSV(d,q_u)$;
  - (**3**) Adaptées (**CERP**) : variantes de $Scores$:
    - $Scores+ : RSV(q,d) + RSV(q_{exp},d)$
    - $Score_{req} : RSV(q,d)$
    - $Score_{exp} : RSV(q_{exp},d)$
- (**1** + **3*****3**) Relations $R_u(w,w')$:
  - (**1**) Profil de l'utilisateur $Profil(u)$
  - (**3**) Voisinages sociaux u_{sn} : très dense, dense et peu dense.
    - (**3**) Profil des voisins $Profil(u',q)$:
      - Filtrer les voisins par rapport à q : oui / non
      - Filtrer les profils voisins par rapport à q : oui / non

Total : 41 runs
1 + **4** * (**1** + **3*****3**)

Expérimentations : résultats

- Baseline et $R_u(w,w')$ basé sur $Profil(u)$:
  - Profil : 
  - Fusion adaptée (**CERP**) : 
  - $Scores_{exp}$: 

Run	Fusion	Profil(u,q)	$Prof_{over}$	MAP	P@5	P@10
$Base$	-	–	0,0	0,2934†	0,1010	0,0585
$R1$	classique	Profil(u)	1,0	0.4616†	0,1562	0,0965
$R1_+$	$Scores_+$	Profil(u)	"	0,4945	0,1900	0,1274
$R1_{req}$	$Scores_{req}$	Profil(u)	"	0,4176†	0,1483	0,1260
$R1_{exp}$	$Scores_{exp}$	Profil(u)	"	**0,5007**	**0,1970**	**0,1303**

† Différence significative (MAP) vs **meilleur run**

Expérimentations : résultats (2)

- $R_u(w,w')$ basé sur u_{sn} et $Profil(u',q)$.
  - Fusion adaptée (**CERP**) : ☺
  - $Scores_+$ et $Scores_{exp}$: ☺
  - Voisinages très denses : ☺
  - Filtrer les voisins : ☺
  - Filtrer les profils des voisins :
    - Voisinages très denses : ☺
    - Voisinages denses, peu denses : ☹

Voisins	Fusion	u_{sn}	$Profil(u',q)$	$Prof_{over}$	MAP	P@5	P@10
Très dense	$Scores_+$	filtré	filtré	0,3086	**0,5537**	0,2060	0,1269
Dense	$Scores_{exp}$	filtré	non filtré	0,6770	0,4842	0,1811	0,1239
Peu dense	$Scores_+$	filtré	non filtré	0,6300	0,4063	0,1582	0,1090

Conclusion

- Axiome simple pour la RI Personnalisée :
  - **CERP** : Contrainte d'Expansion de Requête Personnalisée.
  - Très restrictive (trop ?).
- A**daptation** de l'expansion de requête → valider **CERP**.
- Expérimentations :
  - 41 configurations.
  - Intérêt :
    - de la fusion adaptée (**CERP**).
    - de l'utilisation du voisinage.
    - du filtrage les voisins.
    - du filtrage des profils dans le cas de voisinages très denses.

Futur

- Étudier le choix des termes d'expansion :
  - Construction du voisinage ;
  - Filtrage des voisins et des profils ;
  - Utiliser des relations explicites entre utilisateurs.

- Étudier les conflits avec d'autres contraintes (Fang, Gaussier, …) ➔ modélisations cohérentes pour la RI.

- « Assouplir » CERP ? ➔ s'inspirer des logiques de descriptions (variantes +/- expressives) : des classes de contraintes +/- restrictives ?

Merci !

Expérimentations : résultats (2)

- $R_u(w,w')$ basé sur u_{sn} et $Profil(u',q)$, voisinages **très denses** :

Run	Fusion	u_{sn}	Profil(u',q)	$Prof_{over}$	MAP	P@5	P@10
$R2$	classique	filtré	filtré	0,3086	0.5179†	0.1781	0.1090
$R2_+$	$Scores_+$	filtré	filtré	"	**0.5537**	0.2060	0.1269
$R2_{req}$	$Scores_{req}$	filtré	filtré	"	0.4386†	0.1542	0.1050
$R2_{exp}$	$Scores_{exp}$	filtré	filtré	"	0.5532	**0.2080**	0.1289
$R3$	classique	filtré	non-filtré	1.0	0.4616†	0.1562	0.0965
$R3_+$	$Scores_+$	filtré	non-filtré	"	0.4945†	0.1900	0.1274
$R3_{req}$	$Scores_{req}$	filtré	non-filtré	"	0.4176†	0.1483	0.1055
$R3_{exp}$	$Scores_{exp}$	filtré	non-filtré	"	0.5007†	0.1970	**0.1303**
$R4$	classique	non-filtré	non-filtré	1.0	0.4616†	0.1562	0.0965
$R4_+$	$Scores_+$	non-filtré	non-filtré	"	0.4945†	0.1900	0.1274
$R4_{req}$	$Scores_{req}$	non-filtré	non-filtré	"	0.4176†	0.1483	0.1055
$R4_{exp}$	$Scores_{exp}$	non-filtré	non-filtré	"	0.5007†	0.1970	**0.1303**

† Différence significative (MAP) vs **meilleur run**

- Utiliser la fusion adaptée (**CERP**) est mieux.
- Utiliser $Scores_+$ ou $Scores_{exp}$ est mieux.
- Filtrer voisins et profils est mieux en MAP et $P@5$.

Expérimentations : résultats (3)

- $R_u(w,w')$ basé sur u_{sn} et $Profil(u',q)$, voisinages **denses** :

Run	Fusion	u_{sn}	Profil(u',q)	$Prof_{over}$	MAP	P@5	P@10
$R5$	classique	filtré	filtré	0.2508	0.3926†	0.1373	0.0806
$R5_+$	$Scores_+$	filtré	filtré	"	0.4167†	0.1632	0.0900
$R5_{req}$	$Scores_{req}$	filtré	filtré	"	0.3429†	0.1413	0.0821
$R5_{exp}$	$Scores_{exp}$	filtré	filtré	"	0.4016†	0.1592	0.0930
$R6$	classique	filtré	non-filtré	0.6770	0.4475	0.1582	0.0920
$R6_+$	$Scores_+$	filtré	non-filtré	"	0.4828	0.1811	0.1229
$R6_{req}$	$Scores_{req}$	filtré	non-filtré	"	0.3966†	0.1403	0.1025
$R6_{exp}$	$Scores_{exp}$	filtré	non-filtré	"	**0.4842**	0.1811	**0.1239**
$R7$	classique	non-filtré	non-filtré	"	0.4247†	0.1552	0.0876
$R7_+$	$Scores_+$	non-filtré	non-filtré	0.8695	0.4394	0.1791	0.1055
$R7_{req}$	$Scores_{req}$	non-filtré	non-filtré	"	0.3801†	0.1552	0.0935
$R7_{exp}$	$Scores_{exp}$	non-filtré	non-filtré	"	0.4405†	**0.1821**	0.1075

† Différence significative (MAP) vs **meilleur run**

- Utiliser la fusion adaptée (**CERP**) est mieux.
- Utiliser $Scores_+$ ou $Scores_{exp}$ est mieux.
- Filtrer les voisins et ne pas filtrer leur profil est mieux.

Expérimentations : résultats (4)

- $R_u(w,w')$ basé sur u_{sn} et $Profil(u',q)$, voisinages **peu denses** :

Run	Fusion	u_{sn}	Profil(u',q)	$Prof_{over}$	MAP	P@5	P@10
$R8$	classique	filtré	filtré	0.2286	0.3780	0.1303	0.0756
$R8_+$	$Scores_+$	filtré	filtré	"	0.3913	0.1512	0.0925
$R8_{req}$	$Scores_{req}$	filtré	filtré	"	0.3537†	0.1423	0.0900
$R8_{exp}$	$Scores_{exp}$	filtré	filtré	"	0.3874	0.1493	0.0896
$R9$	classique	filtré	non-filtré	"	0.3690†	0.1323	0.0791
$R9_+$	$Scores_+$	filtré	non-filtré	0.6300	**0.4063**	0.1582	0.1090
$R9_{req}$	$Scores_{req}$	filtré	non-filtré	"	0.3405†	0.1343	0.0980
$R9_{exp}$	$Scores_{exp}$	filtré	non-filtré	"	0.3742†	0.1602	**0.1109**
$R10$	classique	non-filtré	non-filtré	"	0.3736	0.1343	0.0786
$R10_+$	$Scores_+$	non-filtré	non-filtré	0.8150	0.4059	0.1612	0.0980
$R10_{req}$	$Scores_{req}$	non-filtré	non-filtré	"	0.3574†	0.1512	0.0940
$R10_{exp}$	$Scores_{exp}$	non-filtré	non-filtré	"	0.3909	**0.1662**	0.1010

† Différence significative (MAP) vs **meilleur run**

- Utiliser la fusion adaptée (**CERP**) est mieux.
- Utiliser $Scores_+$ est mieux.
- Filtrer les voisins et ne pas filtrer leur profil est mieux.

Expérimentations : résultats

- Baseline et $R_u(w,w')$ basé sur $Profil(u)$:

Run	Fusion	Profil(u,q)	$Prof_{over}$	MAP	P@5	P@10
$Base$	-	–	0,0	0,2934†	0,1010	0,0585
$R1$	classique	Profil(u)	1,0	0.4616†	0,1562	0,0965
$R1_+$	$Scores_+$	Profil(u)	"	0,4945	0,1900	0,1274
$R1_{req}$	$Scores_{req}$	Profil(u)	"	0,4176†	0,1483	0,1260
$R1_{exp}$	$Scores_{exp}$	Profil(u)	"	**0,5007**	**0,1970**	**0,1303**

† Différence significative (MAP) vs **meilleur run**

- Utiliser le profil est mieux.
- Utiliser la fusion adaptée (**CERP**) est mieux.
- Utiliser $Scores_{exp}$ est mieux.

État de l'art – RI Personnalisée

- Exemples :
  - Profil et activité de l'utilisateur [Biancalana et al. 2013]
    - Co-occurrence de tags, expansion de requête.
  - Réseaux sociaux [Vosecky et al. 2014]
    - Relations utilisateurs, LDA, expansion de requête.

Proposition : CERP

- Remarque : **CERP** est :
  - différente de la contrainte **TFC3** (favorise les documents ayant plus de termes de la requête distincts [Fang et al. 2004]) ;
  - différente de la contrainte sur les relations sémantiques entre termes de [Fang et al. 2006] ;
  - incompatible avec **TF-LNC** (TF et longueur du document [Fang et al. 2004]) ;