Variations axiomatiques pour la recherche d'information personnalisée

Philippe Mulhem, Nawal Ould Amer, Mathias Géry

Variations axiomatiques pour la recherche d'information personnalisée

Philippe Mulhem * — **Nawal Ould Amer***,** — **Mathias Géry** **

* *LIG - Université de Grenoble, {Nawal.Ould-Amer, Philippe.Mulhem}@imag.fr*
** *LaHC - Université de Saint-Étienne, Mathias.Gery@univ-st-etienne.fr*

RÉSUMÉ. *Cet article s'intéresse à l'exploitation du profil des utilisateurs pour la recherche d'information dans un réseau social d'annotation (tagging). Nous faisons l'hypothèse que le profil doit être filtré de manière adéquate pour permettre une personnalisation efficace de la requête. Afin d'étudier cette personnalisation d'un point de vue axiomatique, la contrainte d'expansion de requête basée sur le profil est alors définie. Elle décrit le comportement attendu des termes du profil utilisateur qui permettront de personnaliser la requête. Nous fixons un exemple d'intégration de cette contrainte dans le cadre d'un modèle probabiliste, avant d'étudier l'impact de la requête dans le filtrage du profil d'un utilisateur. Les expérimentations effectuées sur le corpus Bibsonomy montrent que même une mise en œuvre simple de la contrainte donne de bons résultats pour la personnalisation.*

ABSTRACT. *This paper focuses on difficulty of finding relevant information in a social network. We start from the premise that user profiles must be filtered to have a positive impact on the retrieval. The Profile Query Expansion Constraint is then defined: it defines the expected behavior of the terms that are used to personalize a user query. We define, then, one integration of the constraint in a probabilistic matching framework, before studying when the query may be used to help focusing the social profile of a user asking a query. Experiments are processed on the Bibsonomy corpus. Our findings show that even simple profile adaptations are effective for social personalized information retrieval.*

MOTS-CLÉS : *réseau social, modèles probabilistes, sélection de profil, axiomatique.*

KEYWORDS: *social network, probabilistic retrieval, profile selection, axiomatic.*

1. Introduction

Différents utilisateurs, avec des besoins d'information différents, peuvent attendre des résultats différents d'un système de recherche d'information (SRI). La personnalisation des SRI vise à prendre en compte ces spécificités. Une question encore largement ouverte est de déterminer quelle information personnelle est pertinente à considérer pour personnaliser les résultats. Les informations typiques sont les centres d'intérêts des utilisateurs ou l'historique de leur comportement (*logs*). Les *profils* des utilisateurs sont dédiés à cette représentation. Ces profils peuvent théoriquement être utilisés à toutes les étapes du calcul de correspondance entre requêtes et documents, mais leur emploi est le plus couramment considéré pour l'expansion de requête ou durant un réordonnancement des résultats (*reranking*). L'expansion de requêtes est la solution la plus simple à mettre en place dans la mesure où elle ne modifie pas les modèles sous-jacents de RI, mais elle doit être très contrôlée pour ne pas dégrader les réponses.

Depuis 2004 et les travaux de (Fang *et al.,*, 2004), le domaine de la caractérisation axiomatique de la recherche d'information s'est développé. Le but de ce domaine est de définir les comportements attendus des modèles et des systèmes sous forme d'"axiomes" (en fait des contraintes heuristiques), et d'évaluer théoriquement les modèles en fonction de leur validation de ces axiomes. Un résultat impressionnant de ces travaux (Fang *et al.,*, 2011) a été d'améliorer sur ces critères théoriques l'un des meilleurs modèles de RI, BM25 (Robertson *et al.,*, 1995).

Notre approche est inspirée de ces travaux : nous proposons un axiome pour la personnalisation de la recherche d'information utilisant les annotations (appelées *tags* ou *bookmarks*) des utilisateurs comme profils, à base d'expansion de requêtes.

Dans la suite de cet article, nous décrivons en section 2 une étude empirique réalisée sur une corpus d'annotation sociale qui vont diriger nos propositions. La section 3, présente les travaux de l'état de l'art relatifs à notre proposition. La section 4 présente l'axiome CERP de personnalisation par expansion de requête basée sur le profil, et une étude formelle de la validation de CERP par le modèle de langue avec lissage de Dirichlet utilisé dans (Fang *et al.,*, 2011). La section 5 présente le cadre proposé, et plus précisément l'adaptation d'une approche classique pour garantir la validation de CERP. La section 6 présente les différentes expansions proposées, l'une d'elles intégrant le voisinage social de l'utilisateur qui pose la requête. Les expérimentations sur le corpus Bibsonomy sont décrites en section 7, avant de conclure.

2. Étude empirique

Avant de définir la contrainte heuristique qui est la partie fondamentale de notre proposition, nous étudions un corpus de bookmarking social afin de saisir de façon informelle un comportement approprié d'un SRI dans ce cadre. Pour cela, nous étudions un ensemble de 200 utilisateurs tirés du corpus Bibsonomy. Ce corpus est composé de 240 000 pages web comprenant 1,5 million d'annotations produites par environ 5 000

utilisateurs (cf. section 7.1 pour plus de détails). Nous étudions, pour une requête composée d'un tag employé par un utilisateur pour annoter des documents, la proportion de documents pertinents qui sont annotés par un autre terme du profil de cet utilisateur (ce terme étant affecté par l'utilisateur lui-même ou bien par d'autres utilisateurs). Cette étude nous sert à estimer l'intérêt potentiel d'utiliser le profil de l'utilisateur pour personnaliser une requête. Le tableau 1 montre que pour 100% des documents pertinents, au moins un autre terme du profil de l'utilisateur à été employé pour décrire ce document. De plus, dans plus de 60% des cas, au moins 3 termes du profil de l'utilisateur ont décrit le document. Ce résultat empirique confirme qu'au moins une partie du profil d'un utilisateur peut aider à personnaliser la recherche d'information. Ce point est formalisé dans la section 4.

Tableau 1 – Distribution moyenne des tags des utilisateurs pour 8442 documents pertinents de la section 7.1.

1 ou plus	2 ou plus	3 ou plus	4 ou plus	5 ou plus
100,00%	98,70%	60,46%	23,00%	7,96%

Les résultats ci-dessus sont constatés sur les tags considérés indépendamment, mais nous étudions également les sujets (*topics*) reliés aux tags pour déterminer si différents termes correspondent au même sujet. Cette analyse permettra de caractériser la proportion d'utilisateurs qui ont plusieurs centres d'intérêts d'après leur profil. Pour cela, nous étudions la distribution des sujets sur le même ensemble d'utilisateurs que ci-dessus. Nous avons généré un modèle d'Allocation Latent de Dirichlet (Blei *et al.,*, 2003) sur l'ensemble des utilisateurs du corpus Bibsonomy (cf. partie 7.1), avec un nombre de sujets de 100 (nombre classique, cf. (Zhao *et al.,*, 2011)). En définissant un seuil de 0,1 pour l'assignation d'un sujet à un utilisateur, nous trouvons que 77% de ces utilisateurs ont plus d'un centre d'intérêt (c.-à-d. d'un sujet), et que le nombre moyen de sujets par utilisateur est de 2,3. Si nous estimons, tout comme (Vosecky *et al.,*, 2014), qu'une requête porte sur un sujet, il est alors clair qu'il faut filtrer les termes du profil utilisateur pour étendre la requête en utilisant seulement les termes relatifs au sujet de la requête. Les éléments ci-dessus renforcent notre idée initiale qu'un sous-ensemble *adéquat* du profil utilisateur doit être sélectionné afin de personnaliser la recherche.

3. État de l'art

Recherche d'information personnalisée. Pour personnaliser la recherche d'information, le processus de correspondance peut intégrer un modèle de l'utilisateur basé sur ses centres d'intérêts, son comportement et son historique. Les approches classiques reposent sur l'utilisation de logs (Dou *et al.,*, 2007), les documents générés par les utilisateurs (tweets, blogs, commentaires) (Vosecky *et al.,*, 2014), les tags ou marques des utilisateurs (Biancalana *et al.,*, 2013 ; Wu *et al.,*, 2006). Comme décrit plus haut, les scénarios les plus courants pour utiliser les profils des utilisateurs sont : l'expansion de requêtes et le réordonnancement.

De nombreux travaux personnalisent par réordonnancement les documents en se basant sur les profils utilisateurs et d'autres information sociales, comme (Bouadjenek *et al.,*, 2011 ; Zhou *et al.,*, 2008 ; Vosecky *et al.,*, 2014). Dans les systèmes de bookmarking, ces profils sont dérivés des tags des utilisateurs (Bouadjenek *et al.,*, 2013 ; Xu *et al.,*, 2008 ; Zhou *et al.,*, 2008). La proposition de (Biancalana *et al.,*, 2013) repose sur l'expansion de requêtes, en sélectionnant les termes du profil qui sont liés à la requête. (Bouadjenek *et al.,*, 2011) définit un réordonnancement qui utilise les graphes des relations entre utilisateurs, documents et tags : les relations entre les tags et la requête d'un côté, et entre la requête et le profil de l'utilisateur de l'autre, sont mises en œuvre. (Bouadjenek *et al.,*, 2013 ; Xu *et al.,*, 2008) considèrent de leur côté la correspondance entre les annotations sociales d'un document et la requête, ainsi qu'entre le profil utilisateur et le document. Pour détailler les différents centres d'intérêts d'un utilisateur, (Zhou *et al.,*, 2008) utilise un modèle probabiliste génératif basé sur l'allocation de Dirichlet Latente (LDA). De manière générale, toutes ces approches améliorent les résultats. Les approches à base de sujets (*topics*) par LDA sont utiles pour focaliser sur une partie du profil de l'utilisateur, mais ont des limites en particulier dans le cas de documents courts (Tang *et al.,*, 2014), générant des imprécisions dans l'estimation des probabilités. Rester sur les termes eux-mêmes (ou les tags) est donc une solution plus intéressante, au détriment il est vrai d'une certaine généralisation. Nous explorons dans cet article la sélection des termes du profil pour l'expansion de requête. Pour raffiner le profil des utilisateurs, certains travaux se basent sur d'autres utilisateurs du réseau social. On peut utiliser des relations explicites entre utilisateurs comme des *"followers"* (Vosecky *et al.,*, 2014) ou des *"amis"* (Khodaei *et al.,*, 2015) avec l'auteur de la requête, sélectionner des utilisateurs qui ont annoté un document (Bouadjenek *et al.,*, 2013), ou bien ceux qui ont un centre d'intérêt proche (Xu *et al.,*, 2008) de l'auteur de la requête. (Carmel *et al.,*, 2009) utilise également les relations sociales du réseau, en réordonnant les documents suivant ces utilisateurs. Utiliser d'autres utilisateurs du réseau semble donc une bonne idée, sous réserve de les utiliser à bon escient.

Notre proposition est de considérer des approches à base d'expansion de requêtes comme (Biancalana *et al.,*, 2013 ; Bouadjenek *et al.,*, 2011) et de recherche dans des données sociales (Bouadjenek *et al.,*, 2013) suivant les directions suivantes : (i) l'adaptation du profil de l'utilisateur selon ses voisins sociaux et la requête ; et (ii) la sélection des parties de profil adaptées à la requête en se concentrant sur les éléments reliés à la requête. Ce travail est une extension de (Mulhem *et al.,*, 2016), avec de nouvelles expérimentations et explications théoriques.

Axiomatisation de la RI. Notre proposition repose sur la définition d'axiomes (heuristiques), qui dénotent les comportements attendus d'un SRI personnalisé. De tels axiomes sont supposés : i) expliquer le rôle des éléments utilisés par un SRI, ii) permettre la comparaison des modèles d'un point de vue théorique, iii) proposer de nouveaux modèles basés sur ces axiomes. Par exemple, (Fang *et al.,*, 2004) en 2004 a posé les premières pierres de ce domaine, avec des contraintes liées au tf, à l'idf et à la taille des documents. Des travaux ultérieurs se sont intéressés à l'utilisation de relations sémantiques (Fang *et al.,*, 2006), ou bien au pseudo bouclage de pertinence (Clinchant *et al.,*, 2010). Tous ces éléments renforcent les bases théoriques de la

recherche d'information. A notre connaissance, aucun travail d'axiomatisation (à part (Mulhem *et al.,*, 2016) indiqué plus haut) ne s'est porté sur la personnalisation pour la RI. La raison en est que la RI personnalisée est une question encore exploratoire, et les remarques formulées dans (Pasi,, 2010), soulignant les difficultés de modélisation et d'utilisation des modèles utilisateurs, ainsi que celle de l'évaluation de ces approches, sont encore largement valides de nos jours, ce qui repousse probablement les avancées axiomatiques sur le sujet.

4. Contrainte d'expansion de requête personnalisée

4.1. *Notations*

Nous définissons dans le tableau 2 les notations utilisées dans la suite de l'article.

Tableau 2 – Notations.

Notation	Description
G	Le réseau social de tagging
	G est le graphe : $G =<< D, U, W >, R >$
D	l'ensemble des documents $d \in D$
U	l'ensemble des utilisateurs du réseau, avec $u \in U$
W	l'ensemble des tags assignés par les utilisateurs aux documents
R	les tags assignés par les utilisateurs aux documents
	$(R \subset D \times U \times W)$
Q	l'ensemble des requêtes $q \in Q$
$c(w, d)$	le nombre d'occurrences du mot w dans le document d
$c(w, q)$	le nombre d'occurrences du mot w dans la requête q
$RSV(d, q)$	le score du document d pour la requête q
$Profil(u)$	le profil de l'utilisateur u par rapport à tous les tags qu'il a utilisés
	$Profil(u) = \{w \mid w \in W, d \in D, R_u(d, u, w)\}$
R_u	relation terme-terme pour l'utilisateur u
	$(w, w') \in R_u$ signifie que w et w' sont reliés pour l'utilisateur u
q_u	la requête étendue par des termes provenant du profil de u
$RSV(d, q_u)$	le score du document d pour la requête étendue q_u
$Profil(u, q)$	le profil de u filtré par la requête q
	par rapport à la relation terme-terme R_u
$R_{u-local}$	relation terme-terme pour u basée sur les tags de u
$RSV(d, q, u)$	le score d'un document d pour une requête q et un utilisateur u
$u_{sn} \subset U$	le voisinage social de u : un ensemble d'utilisateurs reliés à u
$R_{u-social}$	relation terme-terme pour un utilisateur u considérant u_{sn}

4.2. *Définition de la Contrainte d'Expansion de Requête Personnalisée*

Cette contrainte est relative au fait que, *ceteris paribus*[1], nous supposons que l'utilisation de termes "adéquats" d'un profil utilisateur doit impacter positivement la qualité des résultats d'un SRI personnalisé. En d'autres termes, les documents qui correspondent le mieux à la requête étendue doivent être présentés avant les documents qui correspondent le mieux à la requête initiale. Le terme "adéquat" fait ici référence aux termes relatifs à la requête qui satisfont une relation terme-terme R_u qui dépend de l'utilisateur u posant la requête. Cette contrainte est exprimée par :

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 $q_u = q \cup \{w'\}$.

Cette définition dépend grandement de la relation entre termes personnalisés R_u qui, de manière évidente, influence le résultat : si R_u ne relie pas correctement des termes selon l'utilisateur u, alors il est quasiment certain que valider la contrainte aura un impact négatif sur la qualité du système. Dans la suite, nous allons étudier l'utilisation d'éléments sociaux pour définir la relation R_u. Il est à noter que notre proposition n'est pas comparable à celle dédiée aux relations sémantiques entre termes de (Fang *et al.,*, 2006). En effet, nous forçons les documents qui correspondent au profil utilisateur à être en tête des réponses, car nous considérons que les données que nous possédons sur les utilisateurs sont de première importance.

4.3. *Étude de la contrainte CERP*

Nous montrons ici que le modèle de langue avec lissage de Dirichlet (l'un des modèles les meilleurs, cf. (Fang *et al.,*, 2011)), ne valide pas inconditionnellement CERP. Cet exemple n'est pas une preuve théorique généralisable, mais il permet de montrer un cas qui entre en conflit avec un axiome classique de recherche d'information. De (Fang *et al.,*, 2004), nous savons qu'une telle correspondance entre un document d et une requête q est calculée par la formule suivante :

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

1. "tout étant égal par ailleurs".

En utilisant la formule [1], et en ne considérant le cas que d'un seul document d' par rapport à d, la conclusion de CERP est réécrite en :

$$\begin{aligned}&\sum_{t\in d\cap q_u}[c(t,q_u).ln(1+\tfrac{c(t,d)}{\mu.p(t|D)})]+|q_u|.ln(\tfrac{\mu}{|d|+\mu})\\&\geq\sum_{t\in d'\cap q_u}[c(t,q_u).ln(1+\tfrac{c(t,d')}{\mu.p(t|D)})]+|q_u|.ln(\tfrac{\mu}{|d'|+\mu})\end{aligned}\quad[2]$$

Cette inégalité stipule donc que la correspondance entre le document d et la requête q est supérieure à celle entre d' et q. Notre objectif étant ici de prouver que cette inégalité n'est pas vérifiée inconditionnellement, il nous suffit de montrer qu'il existe des cas pour lesquels elle n'est pas validée. En posant, *ceteris paribus*, que w et w' n'apparaissent qu'une fois dans q_u (i.e. $c(w,q_u)=c(w',q_u)=1$), que d et d' ont la même longueur (i.e. $|d|=|d'|$), que w et w' ont le même nombre d'occurrences dans d (i.e. $c(w,d)=c(w',d)$) et que w et w' apparaissent autant de fois dans le corpus (i.e. $p(w|D)=p(w'|D)$), l'équation [2] est simplifiée en :

$$2.ln(1+\frac{c(w,d)}{\mu.p(w|D)})\quad\geq\quad ln(1+\frac{c(w,d')}{\mu.p(w|D)})\quad[3]$$

Si nous supposons de plus que w apparaît k fois plus dans d' que dans d (i.e. $c(w,d')=k.c(w,d)$), la contrainte CERP est validée quand :

$$\frac{c(w,d)}{\mu.p(w|D)}+2\quad\geq\quad k\quad[4]$$

L'équation [4] est valide inconditionnellement quand $k\leq 2$. Dans le cas, tout à fait réaliste, où le nombre d'occurrences de w dans d divisé par μ est égal à la probabilité de w dans le corpus, c'est-à-dire si le ratio $\frac{c(w,d)}{\mu}=p(w|C)$, nous avons la contrainte validée pour une valeur de $k\leq 3$. Ici, dès que w apparaît dans plus de 3 fois dans d' (avec toutes les spécificités définies plus haut), alors son score de correspondance est plus élevé que celui de d.

Nous avons donc prouvé que le modèle de langue avec lissage de Dirichlet ne valide pas inconditionnellement CERP, et de plus que les conditions de non validation ne sont pas des cas extrêmes. Nous proposons dans la section suivante une solution générique pour valider inconditionnellement CERP, basée sur du réordonnancement.

5. Recherche d'information personnalisée

5.1. *Fusion classique*

Nous utilisons un modèle probabiliste classique de recherche d'information. Le score de pertinence RSV d'un document d pour une requête q posée par un utilisateur u est défini par :

$$RSV(d, q, u) \quad \propto \quad RSV(d, q_u) \qquad [5]$$

avec q_u la requête étendue à partir du profil de l'utilisateur u : $q_u = q \cup \{w' | w' \in W, \exists w \in q; R_u(w, w')\}$. Dans notre cas, chaque document d est taggé avec un système de tags, et il est décrit par deux facettes : le contenu du document, noté σd, et l'ensemble des tags qui lui sont assignés, noté τd. Nous prenons en compte ces deux facettes de la manière suivante : pour une requête q, nous faisons une combinaison linéaire des scores de σd et τd, comme dans (Bouadjenek *et al.,*, 2013). Ceci donne :

$$RSV(d, q) \quad = \quad \lambda.P(q|\sigma d) + (1 - \lambda).P(q|\tau d) \qquad [6]$$

Chacune de ces probabilités utilise des modèles de langues avec lissage de Dirichlet (cf. formule [1]).

5.2. *Fusion adaptée pour valider CERP*

Nous avons montré plus haut que le modèle utilisé ne valide pas inconditionnellement CERP. Nous proposons une manière simple de forcer la validation de CERP avec les quatre étapes suivantes :

1) Évaluer $RSV(d, q)$ sur les documents du corpus, c.-à-d. sans personnalisation. Le résultat est la liste L_{init} de couples $< d, RSV(d, q) >$. Assigner à la variable $Topscore_{init}$ le score le plus grand de la liste L_{init} ;

2) Évaluer $RSV(d, q_u \setminus q)$ sur les documents du corpus, c.-à-d. les scores des documents d pour la partie étendue de q_u (sans la requête initiale). Le résultat est la liste L_{exp} de couples $< d, RSV(d, q_u \setminus q) >$;

3) Fusionner L_{init} et L_{exp} en respectant la règle suivante : a) pour chaque d à la fois dans L_{init} et L_{exp}, le score final de d est la **somme** d'une fonction de fusion (appelée **fuse_val**) de ses scores dans les deux listes **et** de la valeur de $Topscore_{init}$; b) pour tout d appartenant à L_{init} **mais pas** à L_{exp}, son score final est celui de L_{init} ;

4) Trier les résultats suivant les valeurs obtenues en étape 3.

Avec cette séquence, nous sommes certains que la liste des documents triés selon ces nouvelles valeurs de pertinences valide CERP, car les documents qui répondent à la fois à l'extension (extraite du profil de l'utilisateur) et à la requête sont en tête.

6. Termes de l'expansion personnalisée de requête

Dans la suite, nous étudions différentes variations de u_{sn}, le voisinage social de u, et des variations des profils de ces voisins, par rapport à la requête q. Plus précisément, nous commençons par décrire différentes versions de la relation $R_u(w, w')$ utilisée dans la définition de CERP, ces relations ayant un rôle primordial dans la contrainte. Notre expansion personnalisée (pour un utilisateur u), peut également se servir d'informations d'autres utilisateurs u' du réseau de tagging, et notre idée est d'étudier l'impact de plusieurs catégories de voisinages, basés sur les tags, et le choix des tags des utilisateurs u' pour l'expansion de requête. Nous rappelons que nous notons $Profil(u, q)$ le profil personnalisé de u suivant la requête q. Comme vu précédemment, le lien entre $Profil(u, q)$ et R_u est le suivant : $Profil(u, q) = \{w'|w' \in W, \exists w \in q; R_u(w, w')\}$. Nous définissons dans la suite deux relations, $R_{u-local}$ et $R_{u-social}$, qui décrivent deux manières de définir le profil personnalisé de u.

6.1. *Tagging local :* $R_{u-local}$

Supposons une requête posée par un utilisateur u. La première idée est d'ajouter à la requête des termes provenant du profil de u, en se basant sur le comportement de u : on sélectionne les tags du profil de u qui ont été utilisés conjointement avec l'un des termes de la requête q de u, en supposant que cette co-occurrence ait du sens pour u. Formellement, nous définissons une variante de la relation R_u, notée $R_{u-local}$:

$$R_{u-local}(w, w') \Leftrightarrow \exists d \in D, R(d, u, w) \wedge R(d, u, w') \quad [7]$$

Si cette proposition est simple à mettre en œuvre, il est clair cependant qu'elle ne peut convenir dans les cas suivants : i) un utilisateur ne fait pas forcément usage d'un terme qu'il a déjà utilisé pour tagger des documents, et ii) un utilisateur ne tagge pas forcément un document avec plusieurs tags. C'est pour ces raisons que nous proposons une seconde manière de définir la relation en termes, en se basant sur des "experts" du réseau par rapport à u et à sa requête q.

6.2. *Tagging social utilisant un voisinage :* $R_{u-social}$

Dans un premier temps, nous nous basons sur l'ensemble des utilisateurs capables d'aider l'expansion de requête. Dans la suite, nous notons ce voisinage u_{sn}. Ces "experts" sont choisis selon leur familiarité à la requête, et/ou leur similarité avec l'utilisateur u qui pose la requête. Dans la section 7.2, dédiée aux expérimentations, nous étudions plusieurs variations de ce voisinage. Encore une fois, nous choisissons ici d'utiliser les profils des utilisateurs $u' \in u_{sn}$. Ces profils sont construits comme celui de u dans la sous-section précédente $R_{u'-local}$, les profils sont filtrés par la co-

occurrence des tags avec le terme de la requête. Le profil de u en utilisant ce voisinage est bâti par la relation $R_{u-social}$, telle que :

$$R_{u-social}(w, w') \Leftrightarrow \exists u' \in u_{sn}, R_{u'-local}(w, w') \quad [8]$$

7. Évaluations expérimentales

7.1. *La collection Bibsonomy et le protocole d'évaluation*

Nous considérons le réseau social de tagging, Bibsonomy[2], comme sources du profil des utilisateurs. Bibsonomy est un réseau de tagging dans lequel les utilisateurs partagent des documents avec d'autres utilisateurs via leurs tags. Il est alors possible d'accéder aux documents annotés par un utilisateur, ou d'accéder à des documents annotés par certains tags. Tiré de ce réseau social de tagging, nous avons utilisé la collection Bibsonomy (Benz *et al.,*, 2010) pour nos évaluations : cette collection contient des identifiants d'articles scientifiques (doi) et les urls de pages webs. Pour nos expérimentations, nous n'avons considéré que les documents avec url pour lesquels les pages webs existaient en septembre 2015. Les caractéristiques de la collection utilisée sont les suivantes : 4 911 utilisateurs, avec les annotations de 241 706 pages webs, 59 886 tags uniques pour un total de 1,5 million d'annotations au total. Dans cette collection, en moyenne, chaque utilisateur a utilisé 263 tags et chaque document est annoté par 6 tags.

Avec cette collection, nous utilisons le même protocole d'évaluation que dans (Bouadjenek *et al.,*, 2013) : nous sélectionnons aléatoirement un utilisateur u, et aléatoirement un tag w utilisé par u. Toutes les annotations de u par le tag w sont alors retirées de la collection. Les documents initialement annotés par u avec le tag w sont alors considérés comme pertinents pour la requête $q = \{w\}$ posée par u, et des mesures classiques de rappel et de précision comme la MAP sont alors utilisées pour mesurer la qualité du système. Suivant ce principe, nous avons créé 200 requêtes. Toutes les mesures de signification statistiques sont réalisées par des tests de Student pairés bilatéraux, avec un seuil de significativité fixé à 5%.

Pour étudier de manière plus spécifique nos propositions, nous caractérisons les *runs* par la valeur de couverture de profil, $Prof_{over}$ ($\in [0, 1]$), qui décrit le recouvrement entre la requête étendue et le profil de l'utilisateur u. Selon la section 2, nous nous attendons à ce qu'un bon recouvrement se situe aux alentours de 50% ou moins, car un profil utilisateur contient en moyenne plus de 2 sujets et une requête est supposée ne porter que sur un sujet.

2. http ://www.bibsonomy.org

7.2. *Configurations testées*

Toutes les expérimentations ont été menées avec le système Terrier 4.0 (Ounis *et al.,*, 2006), avec application de l'anti-dictionnaire anglais et la troncature de Porter pour l'anglais. Le modèle testé est le modèle de langue avec lissage de Dirichlet (avec $\mu = 2500$). Similairement à (Bouadjenek *et al.,* 2013), nous fixons $\lambda = 0.5$ dans l'équation [6]. Nous avons testé quatre groupes de configurations : les "basiques", et l'utilisation de voisinages très denses, denses et peu denses. Cette variation de voisinage simule différentes topologies de réseau. En plus de la fusion classique (cf. partie 5.1) qui consiste à étendre la requête avec le profil, filtré ou non, d'un utilisateur, nous testons trois exemples de filtrages adaptés pour la fonction **fuse_val(req, exp)** : la somme $req + exp$ (notée $Scores_{+}$), req (notée $Scores_{req}$) qui intègre les scores de la requête initiale, ou exp (notée $Scores_{exp}$) qui garde les scores de l'extension de la requête. Dans la suite, nous dénotons l'ensemble des runs d'une configuration Rn qui utilisent une adaptation par $Rn_{\bullet}$.

7.2.1. *Runs de base*

Les deux configurations de base que nous proposons sont : (1) une version non-personnalisée utilisant uniquement la requête initiale, et (2) une approche étendant la requête avec le profil général de l'utilisateur, tel que $Profil(u, q) = Profil(u)$. Nous voyons dans le tableau 3 que l'utilisation du profil complet $R1_{+}$, $R1_{req}$, $R1_{exp}$ et $R1$ donnent de meilleurs résultats que l'approche $Base$ sans expansion. La meilleure configuration des variantes de $R1_{\bullet}$ est celle qui utilise les scores de l'expansion seule, $R1_{exp}$. Les différences sont significatives entre le meilleur des runs $R1_{\bullet}$, c'est-à-dire $R1_{exp}$, et les autres runs $R1_{\bullet}$: $R1$ (p=0.28), $Base$ (p=3,37E-10). La différence est également significative entre entre les runs $R1$ classique et $Base$ (p=1.32E-10).

Tableau 3 – Les performances des runs de base (†, différence significative en MAP vs. meilleur run en gras).

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**

7.2.2. *Voisinages très denses*

Ici l'ensemble des utilisateurs de U est considéré comme voisinage de u, c'est-à-dire que dans ce cas : $u_{sn} = U$ (avec |U|=4 911) Pour cette configuration comme les suivantes, nous étudions également les deux paramètres suivants :

– nous filtrons, ou non, les utilisateurs de u_{sn} en déterminant s'ils sont liés à la requête (i.e., s'ils ont annoté avec l'un des termes de la requête). Dans ce cas, le filtrage génère en moyenne 152 voisins pour u ;

– nous filtrons, ou non, les profils des utilisateurs u' de u_{sn}. Sans filtrage, nous avons donc $Profil(u', q) = Profil(u')$, et avec filtrage nous utilisons pour les utilisateurs u' l'approche décrite pour u en section 6.2.

Les résultats de ces configurations sont présentés dans le tableau 4.

Tableau 4 – Les performances aves les 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**

Nous notons du tableau 4 que les variantes des runs $R3_\bullet$ et $R4_\bullet$ (resp. $R3$ et $R4$) ont les mêmes valeurs de MAP. Ceci provient du fait que l'expansion de requête est la même : filtrer u_{sn} génère déjà le profil total de u (vérifié par la valeur $Prof_{over} = 1,0$), donc ajouter de nouveaux utilisateurs ne modifie pas l'expansion. Ici également, les fusions adaptées donnent de meilleurs résultats par rapport aux fusions classiques. On constate également que filtrer le profil donne de meilleurs résultats, ce qui confirme l'idée que choisir les "bons" termes est important. Les meilleurs résultats de ce tableau en MAP et P@5 sont obtenus en utilisant environ 30% des termes du profil de u, ce qui est cohérent avec nos remarques de l'analyse empirique en section 2. La meilleure variante des runs adaptés $R2_+$, c'est-à-dire $R2_+$, a une différence significativement meilleure en MAP que $R2$ (p=0,021), et que tous les autres résultats, excepté le run adapté $R2_{exp}$.

7.2.3. *Voisinages denses*

Dans ce cas, notre idée est de considérer un voisinage relativement dense pour u_{sn}. Un tel voisinage est composé d'utilisateurs u' qui partagent au moins un tag avec u : $\{u'|Profil(u') \cap Profil(u) \neq \emptyset\}$. Tous ces utilisateurs sont reliés à u, mais il restent potentiellement nombreux. Avec ces voisinages, un utilisateur possède en moyenne 872 voisins. Comme précédemment, nous étudions le filtrage de ces utilisateurs selon la requête, et nous étudions le fait de filtrer ou non les profils des u'. Le filtrage des utilisateurs génère en moyenne 40 voisins. Les résultats de ces voisinages sont présentés dans le tableau 5.

Une fois encore, nous constatons dans le tableau 5 que les fusions adaptées sont meilleures en terme de MAP. Nous constatons cependant que les meilleurs résultats

Tableau 5 – Les performances avec les 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

en MAP et précision à 10 sont obtenus quand les profils des voisins ne sont pas filtrés (run $R6_+$), auquel cas l'expansion contient en moyenne 68% du profil utilisateur. Nous notons également qu'avec les runs $R7_\bullet$ et $R7$, augmenter le recouvrement du profil de l'utilisateur n'aide pas, excepté pour la mesure P@5, un peu meilleure pour le run $R7_+$ que pour $R6_+$. Le meilleur run du tableau 5, $R6_+$, n'a pas de valeur de MAP significativement meilleure par rapport à son équivalent non adapté $R6$ (p=0.070). Ces différences sont cependant significatives par rapport au meilleur run adapté non filtré $R7_{exp}$ (p=0.037) et classique $R7$ (p=0.022). Ici, filtrer les utilisateurs mais pas leur profil fournit les meilleurs résultats.

7.2.4. *Voisinages peu denses*

Le dernier ensemble de configurations étudié imite des voisinages creux (inspiré de (Bouadjenek *et al.,*, 2013)) : le voisinage est composé d'utilisateur u' qui ont annotés un document que u a également annoté, mais pas forcément par le même tag : $\{u'|\exists w \in W, \exists d \in D, R(u', d, w) \wedge R(u, d, W)\}$. Par rapport aux autres voisinages, ceux définis ici sont sensés contenir des utilisateurs assez proches de u, car ils se sont intéressés à au moins un document commun même si ce n'est pas pour les mêmes raisons. En moyenne, un utilisateur possède 56 voisins avec cette approche. De plus, filtrer les utilisateurs en fonction de la requête génère 10 voisins en moyenne (cette valeur est similaire à celle obtenue sur des groupes d'utilisateurs de YouTube dans (Mislove *et al.,*, 2007)). Les résultats sont présentés dans le tableau 6.

Le tableau 6 montre que la meilleure configuration est obtenue dans une fusion adaptée, sans filtrage des voisins mais avec filtrage de leurs profils (comme avec les runs denses), mais avec ces voisinages peu denses les meilleures configurations sont toutes très proches, que l'on filtre ou non les voisins et les profils, même si les valeurs de recouvrement avec le profil varient beaucoup.

Tableau 6 – Les performances avec les 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

7.3. *Discussion*

Cette partie compare les résultats des trois densités de voisinages considérés. Le meilleur résultat global à été obtenu pour des voisinages très denses (run $R2_+$ du tableau 4) filtrés par rapport à la requête. Nous notons également que, dans tous les cas sauf un (run $R9_+$ du tableau 6), les fusions adaptées pour valider CERP sont meilleures que les fusions non adaptées. Même s'il est vrai que nous n'avons pas toujours pu conclure sur la significativité statistique des différences en MAP, la constante amélioration des MAP prouve expérimentalement l'intérêt des adaptations proposées, malgré leur simplicité. On remarque également que sur trois adaptations proposées, l'adaptation $Scores_{req}$, qui repondère uniquement par rapport au score de la requête initiale, n'est pas la meilleure idée : prendre en compte dans le réordonnancement les scores de l'extension est donc préférable. D'autres adaptations ou d'autres axiomes devraient aller plus loin dans cette direction. Les voisinages peu denses sont constamment moins bons que les résultats des voisinages denses et très denses. Ceci est expliqué par le fait que, dans les configurations peu denses, plus de 150 requêtes (sur 200) ne génèrent que des profils sociaux vides, ce qui ne provoque aucune expansion de requête. Cependant, on note que pour les 50 requêtes étendues, la MAP passe de 0.2934 (run $Base$ du tableau 3) à 0.3960 (run $R10_+$ du tableau 6), ce qui une fois de plus souligne l'intérêt potentiel de nos propositions.

Nous nous concentrons maintenant sur les tests de significativité des meilleurs runs par voisinage, par rapport au meilleur run de base. Si nous considérons les voisinages pour lesquels le meilleur résultat est meilleur que le run $R1_+$, c'est-à-dire le run $R2_+$ pour les voisinage très dense et le run $R6_+$ pour les voisinages denses, seul le run $R2_+$ possède une différence significative en MAP avec $R1_+$ (avec p=457E-4). Ce résultat montre encore une fois l'intérêt de la validation de CERP. Nous résumons donc nos résultats par :

– les fusions qui valident la contrainte CERP sont constamment meilleures que la fusion classique. La fusion adaptée est simple et devrait certainement être étendue pour prendre en compte de manière plus claire les requêtes multi-termes, mais elle est dores et déjà prometteuse ;

– le filtrage par requête de nombreux voisins (cas de voisinages très denses) semble préférable au fait de sélectionner a priori les utilisateurs potentiellement intéressants (voisinage denses et peu denses). La difficulté avec une sélection tardive des voisins est liée à la complexité des calculs (et des accès disques) aux données nécessaires, et cette approche est quelque peu en contradiction avec les standards de la RI qui poussent à pré-calculer en amont le maximum de choses.

8. Conclusion

Nous avons proposé dans cet article la définition de la contrainte CERP, et un cadre probabiliste qui filtre le profil d'un utilisateur u qui pose une requête q, pour améliorer les réponses fournies en exploitant le profil pour étendre la requête. Nous avons étudié deux paramètres qui aident à sélectionner les parties pertinentes du profil de l'utilisateur qui pose une requête : la première sélectionne par l'intermédiaire de la requête les utilisateurs du voisinage social utilisés, et la seconde filtre les profils de ces utilisateurs. Les conclusions principales que nous tirons d'expérimentations menées sur la collection Bibsonomy sont que 1) considérer un grand nombre d'utilisateurs du voisinage avant de les filtrer, et 2) filtrer leurs profils, améliorent les résultats. De futurs travaux à court terme porteront sur l'étude de relations explicites d'amitié entre utilisateurs (comme les *followers* de Twitter) en tant que voisins, afin de déterminer si notre proposition est généralisable à ces cas. Une difficulté que nous devons traiter est le cas de "démarrage à froid" quand un utilisateur n'a pas (ou peu) de profil initial, car actuellement notre proposition n'est pas utilisable dans ce cas.

Remerciements
Ce travail est soutenu par le projet ReSPIr de la région Auvergne Rhône-Alpes.

9. Bibliographie

Benz D., Hotho A., Jäschke R., Krause B., Mitzlaff F., Schmitz C., Stumme G., « The Social Bookmark and Publication Management System Bibsonomy », *The VLDB Journal*, vol. 19, p. 849-875, 2010.

Biancalana C., Gasparetti F., Micarelli A., Sansonetti G., « Social Semantic Query Expansion », *ACM Trans. Intell. Syst. Technol.*, vol. 4, n° 4, p. 60 :1-60 :43, 2013.

Blei D. M., Ng A. Y., Jordan M. I., « Latent Dirichlet Allocation », *J. Mach. Learn. Res.*, vol. 3, p. 993-1022, 2003.

Bouadjenek M. R., Hacid H., Bouzeghoub M., « Sopra : a new social personalized ranking function for improving web search », *SIGIR'13*, p. 861-864, 2013.

Bouadjenek M. R., Hacid H., Bouzeghoub M., Daigremont J., « Personalized Social Query Expansion Using Social Bookmarking Systems », *ACM SIGIR*, SIGIR '11, p. 1113-1114, 2011.

Carmel D., Zwerdling N., Guy I., Ofek-Koifman S., Har'el N., Ronen I., Uziel E., Yogev S., Chernov S., « Personalized Social Search Based on the User's Social Network », *ACM CIKM*, CIKM '09, p. 1227-1236, 2009.

Clinchant S., Gaussier E., « Information-based Models for Ad Hoc IR », *ACM SIGIR*, SIGIR '10, p. 234-241, 2010.

Dou Z., Song R., Wen J.-R., « A Large-scale Evaluation and Analysis of Personalized Search Strategies », *WWW*, WWW '07, p. 581-590, 2007.

Fang H., Tao T., Zhai C., « A Formal Study of Information Retrieval Heuristics », *ACM SIGIR*, SIGIR '04, p. 49-56, 2004.

Fang H., Tao T., Zhai C., « Diagnostic Evaluation of Information Retrieval Models », *ACM Trans. Inf. Syst.*, vol. 29, p. 7 :1-7 :42, 2011.

Fang H., Zhai C., « Semantic Term Matching in Axiomatic Approaches to Information Retrieval », *ACM SIGIR*, SIGIR '06, p. 115-122, 2006.

Khodaei A., Sohangir S., Shahabi C., *Recommendation and Search in Social Networks*, Springer International Publishing, chapter Personalization of Web Search Using Social Signals, p. 139-163, 2015.

Mislove A., Marcon M., Gummadi K. P., Druschel P., Bhattacharjee B., « Measurement and Analysis of Online Social Networks », *ACM SIGCOMM*, IMC '07, p. 29-42, 2007.

Mulhem P., Amer N. O., Géry M., *Axiomatic Term-Based Personalized Query Expansion Using Bookmarking System*, Springer International Publishing, p. 235-243, 2016.

Ounis I., Amati G., Plachouras V., He B., Macdonald C., Lioma C., « Terrier : A High Performance and Scalable Information Retrieval Platform », 2006.

Pasi G., « Issues in Personalizing Information Retrieval », *IEEE Intelligent Informatics Bulletin*, vol. 11, p. 3-7, 2010.

Robertson S. E., Walker S., Jones S., Hancock-Beaulieu M. M., M.Gatford, « Okapi at TREC–3 », *Overview of the Third Text REtrieval Conference*, p. 109-126, 1995.

Tang J., Meng Z., Nguyen X., Mei Q., Zhang M., « Understanding the Limiting Factors of Topic Modeling via Posterior Contraction Analysis », *in* , T. Jebara, , E. P. Xing (eds), *ICML-14*, JMLR Workshop and Conference Proceedings, p. 190-198, 2014.

Vosecky J., Leung K. W.-T., Ng W., « Collaborative Personalized Twitter Search with Topic-language Models », *ACM SIGIR'14)*, p. 53-62, 2014.

Wu X., Zhang L., Yu Y., « Exploring Social Annotations for the Semantic Web », *WWW*, WWW '06, p. 417-426, 2006.

Xu S., Bao S., Fei B., Su Z., Yu Y., « Exploring Folksonomy for Personalized Search », *ACM SIGIR'08*, p. 155-162, 2008.

Zhao W. X., Jiang J., Weng J., He J., Lim E.-P., Yan H., Li X., « Comparing Twitter and Traditional Media Using Topic Models », *ECIR*, ECIR'11, Springer-Verlag, p. 338-349, 2011.

Zhou D., Bian J., Zheng S., Zha H., Giles C. L., « Exploring Social Annotations for Information Retrieval », *WWW*, p. 715-724, March, 2008.