

---

## Combinaison des caractéristiques des termes pour l'extension de requêtes en recherche d'information dans les documents semi-structurés

**Lobna Hlaoua, Mohand Boughanem, Karen Pinel-Sauvagnat**

*IRIT-SIG, 118 route de Narbonne, F-31 062 Toulouse Cedex 4, France  
{hlaoua,bougha,sauvagna@irit.fr}*

---

**RÉSUMÉ.** *En Recherche d'Information (RI), l'utilisateur a souvent recours, pour des raisons de simplicité, à de simples mots clés pour exprimer sa requête. Les requêtes composées de mots clés sont aussi très utilisées dans le cadre de la recherche d'information structurée (c'est à dire dans des documents structurés de type XML), puisqu'elles n'exigent pas une connaissance approfondie de la structure des documents. Ces requêtes sont cependant souvent insuffisantes pour décrire les besoins de l'utilisateur. Dans cet article, nous proposons une nouvelle approche de reformulation de requêtes à base de mots clés dans des documents XML. Cette approche permet d'une part d'enrichir la requête initiale en fonction des jugements de pertinence, et d'autre part d'exprimer des degrés de préférence pour chaque nouveau mot clé : c'est ce qu'on appelle la pondération. Notre approche est basée sur une combinaison de mesures caractérisant un terme dans les éléments jugés pertinents pour une requête donnée. Nos propositions ont été évaluées grâce à la campagne d'évaluation INEX et les résultats montrent l'intérêt de nos méthodes.*

**ABSTRACT.** *In traditional Information Retrieval (IR), users often express their needs using simple keywords, because this is the most simple way to query systems. Queries composed of simple keyword terms are also very used in structured IR (i.e. IR on structured documents like XML documents), since they do not require any knowledge of the documents structure. However, such queries are often not sufficient to describe precisely users' needs. In this paper, we propose a new approach of Structured Relevance Feedback on queries composed of simple keywords terms. This approach allows on the one hand to enrich the initial query according to relevance judgements, and on the other hand, to express a degree of preference for each keyword, by weighting them. Our approach is based on a combination of term features in relevant elements. Our proposition is evaluated thanks to the INEX evaluation campaign and results show the interest of our methods.*

**MOTS-CLÉS :** *reformulation, document XML, contexte, contenu, RI.*

**KEYWORDS:** *Relevance Feedback, XML document, context, content, IR.*

---

## 1. Introduction

La recherche d'information est un processus qui se base essentiellement sur la requête exprimée par l'utilisateur pour répondre à ses besoins. Quel que soit le système de recherche utilisé, le résultat d'une recherche ne peut être intéressant si la requête ne décrit pas explicitement et clairement les besoins de l'utilisateur. En général, l'utilisateur se contente de donner quelques mots clés. Ces derniers sont issus d'une connaissance générale sur un domaine donné. Par conséquent, les documents renvoyés par le système de recherche peuvent appartenir à des domaines et disciplines différents par lesquels l'utilisateur n'est pas concerné.

La reformulation de requêtes est une phase importante du processus de recherche d'information. Elle consiste de manière générale à enrichir la requête de l'utilisateur en ajoutant des termes permettant de mieux exprimer son besoin [EFT 00]. Une des techniques les plus répandues en RI est la reformulation par réinjection de la pertinence, communément appelée *Relevance Feedback* (RF). Elle consiste à extraire à partir des documents jugés pertinents par l'utilisateur les mots clés les plus expressifs, et à les ajouter à la requête [ROB 76], [ROC 71].

Notre objectif dans cet article est de proposer une approche de reformulation de requêtes par réinjection de la pertinence dans les documents XML. Or la nature de ce type de documents, comportant du texte et des contraintes structurelles sous forme de balises, réactualise la problématique de la RI classique (plein texte) en général et de la reformulation de requêtes en particulier. Tout d'abord au niveau de l'expression des requêtes, l'utilisateur peut exprimer ses besoins de deux manières, soit en n'utilisant que des mots clés (on parle alors de requêtes CO<sup>1</sup> (Content Only)), ou bien en utilisant des requêtes comportant des mots clés et des contraintes de structure (on parle alors de requêtes CAS (Content And Structure)). Ensuite, contrairement à la RI classique, qui considère le document comme une unité d'information atomique, la RI structurée doit identifier de manière automatique la partie du document (l'élément du document XML), répondant à la fois de manière exhaustive<sup>2</sup> et spécifique<sup>3</sup> [CHI 96] à la requête de l'utilisateur. Remarquons que les requêtes composées de simples mots clés sont les plus utilisées en RI en générale et RI dans les documents XML plus précisément puisqu'elles n'exigent pas une connaissance approfondie de la structure des documents.

Nous nous intéressons donc dans cet article à la reformulation des requêtes de type CO. Nous proposons une nouvelle approche pour enrichir la requête initiale par réinjection de termes pondérés. Les termes sont sélectionnés en utilisant la théorie probabiliste et un facteur contexte et sont pondérés en utilisant soit la théorie probabiliste soit une formule prenant en compte leur importance dans la collection de documents et la collection d'éléments.

L'article est organisé comme suit. Nous commençons par présenter l'état de l'art sur

1. Cette terminologie est utilisée dans le cadre de la campagne INEX (INiative for the Evaluation of XML Retrieval) destinée à l'évaluation de la recherche d'information dans des documents XML.

2. Un élément est exhaustif à une requête s'il répond entièrement au besoin en information de la requête.

3. Un élément est spécifique à une requête si tout son contenu concerne la requête.

la reformulation de requêtes structurées dans la section 2. Nous détaillons notre approche dans la section 3. Nous présentons dans la section 4 le contexte d'évaluation. Enfin dans la section 5 nous détaillons nos différentes expérimentations et nous discutons des impacts des différentes approches.

## 2. Etat de l'art

La reformulation de requêtes par réinjection de pertinence en recherche d'information dans les documents XML a été abordée selon deux principales approches :

- **une réinjection de pertinence orientée contenu,**

basée sur le même principe que la réinjection de pertinence en RI classique. Elle consiste à enrichir la requête initiale en rajoutant des mots clés extraits d'éléments de différentes granularités. La majorité de travaux effectués dans ce cadre se base sur l'algorithme de Rocchio [ROC 71] en considérant chaque élément comme unité atomique. On y trouve notamment les travaux de Y. Mass [MAS 04] et de C. Crouch [CRO 04]. Les résultats obtenus par ces méthodes dans le cadre de la campagne d'évaluation INEX [INE04], [INE05a)] n'excèdent pas les 4% (en moyenne) d'amélioration.

- **une réinjection de pertinence orientée contexte.**

Elle consiste à enrichir la requête initiale en spécifiant le contexte dans lequel un élément pertinent peut être retrouvé (exemple : le nom document, le type d'élément). Cette approche a été appliquée sur deux méthodes. La première est développée par Mihajlovic et al. [MIH 04]. Elle consiste essentiellement à parcourir les éléments jugés pertinents et à extraire le nom du document et la balise auxquels appartiennent les éléments pertinents. Cette méthode a été appliquée sur le système de recherche TIJAH [LIS 03], [MIH 05]. L'amélioration ne dépasse pas les 6%( en moyenne). La deuxième méthode, proposée par Shenkel et Theobald [SCH 05], consiste à extraire des classes décrivant le contexte des éléments pertinents. Ces classes sont : la classe des ancêtres, la classe des descendants et la classe du contenu. Par exemple, pour un élément pertinent de type *section* , les éléments de types *article* et *body* sont ajoutés à la classe des ancêtres, les éléments de types *paragraph* et *subsection* sont ajoutés à la classe des descendants, et les termes de l'élément *section* sont ajoutés à la classe du contenu. L'extension de la requête initiale de type CO consiste à injecter les meilleurs candidats extraits de chaque classe. Ces derniers sont choisis selon leurs fréquences dans les classes correspondantes. On sélectionne ensuite, pour l'élément ancêtre (resp. descendant) à rajouter, le mot clé le plus pertinent selon l'algorithme de Rocchio. Cette technique n'est appliquée qu'aux requêtes de type CO. L'amélioration dépasse en général les 50% (en moyenne).

Dans nos travaux précédents [HLA 06b] et [HLA 06c], nous avons prouvé selon des statistiques que nous avons réalisées sur la collection d'INEX 2005 [HLA 06c], que pour une requête donnée, on peut trouver au moins une structure pertinente et que la majorité des éléments jugés pertinents (90%) appartiennent à trois types de structures différentes. Nous avons alors proposé d'injecter des structures appelées structures gé-

nériques [HLA 06b] dans la nouvelle requête. Les expérimentations ont confirmé notre hypothèse et ont montré l'intérêt de l'approche orientée structure [HLA 06a].

Dans cet article, nous proposons une nouvelle approche de reformulation qui consiste à enrichir la requête initiale de type CO par des mots clés pondérés extraits à partir de l'ensemble des éléments jugés pertinents. Nous étudions également différentes combinaisons de mesures de pertinence caractérisant un terme dans un élément pour une requête donnée.

### 3. Une approche orientée contenu pour la sélection et la pondération des termes à réinjecter dans la requête initiale

#### 3.1. Méthode probabiliste

Une simple adaptation de l'algorithme de Rocchio [ROC 71] pour l'extraction des termes et leur réinjection sans pondération dans la requête initiale n'a pu aboutir à une amélioration intéressante des résultats [HLA 06b]. Notre objectif principal est de pouvoir fixer les critères suivant lesquels nous devons extraire les termes et les pondérer. Nous nous sommes inspirés dans notre approche du modèle probabiliste [ROB 76] déjà appliqué en reformulation de requêtes en recherche d'information classique.

On considère la pertinence d'un terme  $t_j$  comme un événement probabiliste. La probabilité qu'un terme soit pertinent est définie par  $P(t_j/E^p)$ , où  $E^p$  est l'ensemble des éléments pertinents. Cette probabilité est calculée par la probabilité définie par Robertson [ROB 76], et simplifiée comme suit :

$$P(t_j/E^p) = |p_{ej}|/|P_e| \quad [1]$$

où  $|p_{ej}|$  est le nombre d'éléments pertinents appartenant à l'ensemble  $E^p$  dans lesquels apparaît le terme  $t_j$  et  $|P_e|$  est le nombre d'éléments pertinents. Les  $k$  meilleurs termes (ayant les scores les plus élevés) seront réinjectés dans la nouvelle requête.

#### 3.2. Raffinement de la sélection

La sélection des termes en fonction de la probabilité conditionnelle n'est pas assez discriminante. En effet, plusieurs termes peuvent avoir les mêmes scores alors qu'ils n'ont pas la même importance aux yeux de l'utilisateur. Nous essayons alors de raffiner notre sélection.

Notre but est de sélectionner les termes expressifs (pertinents à la requête). La pertinence est définie selon deux dimensions : l'exhaustivité et la spécificité. Notre problème revient alors à trouver les termes qui décrivent des éléments à la fois spécifiques et exhaustifs. Pour ce fait, nous proposons d'examiner de près les indicateurs de pertinence d'un terme souvent utilisés dans la littérature. Nous entendons par indicateurs les différentes fonctions généralement liées aux fréquences relatives des termes comme *tf*, *idf*... Nous pouvons classer ces fonctions selon deux catégories : celles qui expriment la spécificité et celles qui expriment l'exhaustivité.

–  $idf_j$  (inverse document frequency) défini dans [K.S 72] :

$$idf_j = \log \left( \frac{|D|}{|d_j|} + 1 \right) \quad [2]$$

où  $|d_j|$  est le nombre de documents contenant le terme  $t_j$  et  $|D|$  est le nombre total de documents dans la collection. Cette caractéristique met en valeur les termes qui n'apparaissent que dans quelques documents de la collection c'est à dire qui sont spécifiques à ces documents.

–  $tf$  (term frequency) [HAR 92], est la fréquence du terme dans un document.

$$tf_j^d = \log(occurrence_j(d) + 1) / \log(occurrence_{total}(d)) \quad [3]$$

où  $occurrence_j(d)$  est le nombre d'occurrences du terme  $t_j$  dans le document  $d$  et  $occurrence_{total}(d)$  est le nombre total de termes dans le document  $d$ . Cette caractéristique reflète l'exhaustivité. En effet plus le terme est présent dans le document, plus le document répond à une requête donnée contenant le terme en question.

–  $Bruit(t_j)$  défini dans [SAL 83], est basé sur la fréquence du terme dans le document.

$$Bruit_j = \sum_{i=1}^N \frac{tf_j^{e_i}}{tf_j} \log \frac{tf_j}{tf_j^{e_i}} \quad [4]$$

où  $tf_j^{e_i}$  est la fréquence du terme  $t_j$  dans l'élément  $e_i$ ,  $N$  est le nombre d'éléments dans la collection et  $tf_j$  est la fréquence totale du terme  $t_j$  dans la collection. Cette caractéristique peut servir à éliminer les termes qui représentent du bruit et à ne garder que les termes spécifiques.

– le *Contexte* [RUT 02] :

$$context_j^{e_i} = (distribution^{e_i}(q) - \min^{e_i}(t_j)) / distribution^{e_i}(q) \quad [5]$$

$$\min^{e_i}(t_j) = \min_{t_j \neq t_k} |(position^{e_i}(t_j) - position^{e_i}(t_k))| \quad [6]$$

$$distribution^{e_i}(q) = length(e_i) / occurrences^{e_i}(q) \quad [7]$$

où  $distribution^{e_i}(q)$  est la distribution de tous les termes de la requête dans l'élément  $e_i$ , avec  $length(e_i)$  la taille de l'élément  $e_i$  et  $occurrences^{e_i}(q)$  le nombre d'occurrences des termes de la requête  $q$  dans l'élément  $e_i$ ,

et  $\min^{e_i}(t_j)$  est la différence minimale de positions entre n'importe quelle occurrence du terme  $t_j$  et un autre terme  $t_k$  de la requête, avec  $position^{e_i}(t_j)$  la position du terme  $t_j$  dans  $e_i$ .

Cette caractéristique permet de mesurer le degré d'appartenance d'un terme au contexte d'une requête donnée. C'est une caractéristique basée sur la distribution des termes et qui sert à mettre en valeur les termes exprimant l'exhaustivité.

En RF, on part d'un ensemble d'éléments pertinents pour sélectionner les termes expressifs en liaison avec la requête initiale. Une première sélection est effectuée en

fonction de l'appartenance d'un terme à l'ensemble des éléments pertinents (probabilité conditionnelle). Un même poids peut être assigné à des termes différents. Notre objectif est maintenant d'affiner notre sélection. Le facteur  $tf$  ne permet pas ce raffinement contrairement au contexte qui peut nous renseigner sur l'importance d'un terme vis à vis d'une requête donnée. *Idf* quant à lui nous renseigne sur la spécificité du terme par rapport un élément, alors que notre but est de se rapprocher le plus possible de la requête. Nous allons alors combiner le poids déjà calculé par la probabilité conditionnelle avec le contexte du terme calculé dans l'ensemble des éléments pertinents. Le poids d'un terme appelé Poids Contextuel (PC) sera calculé suivant l'équation 8 suivante :

$$PC(t_j) = P(t_j/E^p) \times \prod_{i=1}^{|p_{ej}|} context_j^{e_i} \quad [8]$$

où  $|p_{ej}|$  est l'ensemble des éléments pertinents contenant le terme  $t_j$ .

Si on se contente du degré d'appartenance d'un terme au contexte de la requête, on peut se retrouver le cas où le poids d'un terme  $t_j$  est élevé alors qu'il ne présente que du bruit. Ces termes doivent soit être éliminés ou se voir attribuer un poids très faible. Ceci peut correspondre à l'effet du facteur bruit déjà défini (équation 4). Le bruit d'un terme déjà sélectionné comme pertinent ne pourra être évalué qu'en considérant les éléments jugés non pertinents. Notre sélection des termes tiendra donc aussi compte de leur distribution dans les éléments jugés non pertinents. Si on étudie la fonction du bruit de plus près, on constate qu'elle ne peut être nulle que s'il s'agit d'une distribution exceptionnelle du terme en question, ce qui est le cas quand le terme n'apparaît que dans un seul élément. On propose d'appliquer cette fonction en puissance des poids déjà calculés variant entre 0 et 1. Ceci nous ramène à un poids maximal si le terme n'apparaît que dans un seul élément non pertinent.

La fonction du bruit prend des valeurs  $> 1$  lorsqu'un terme se trouve dans plusieurs éléments avec des distributions assez semblables. Le terme en question aura un poids final inférieur à celui déjà calculé. Pour le cas où les valeurs se situent entre 0 et 1, les poids seront élevés tout en restant inférieurs à 1. Le poids du terme  $t_j$  (Poids Ajusté) sera calculé selon l'équation 9 définie comme suit :

$$PA(t_j) = (P(t_j/E^p) \times \prod_{i=1}^{|p_{ej}|} context_{e_i}(t))^{noise(t_j)} \quad [9]$$

### 3.3. Pondération des termes

Dans nos travaux précédents, nous avons considéré la pertinence de façon binaire (i.e. un terme est soit pertinent soit non pertinent). L'utilisateur ne pouvait pas exprimer une préférence qui traduisait l'importance des différents termes pour une requête donnée. Nous proposons dans cet article de pondérer les termes déjà sélectionnés avant leur réinjection dans la nouvelle requête.

Un premier problème auquel nous avons été confronté est celui du système de recherche XFIRM [SAU 06] qui ne considérait pas la pondération des mots clés dans

une requête. Une première étape a alors été d'adapter ce système afin de supporter la pondération au niveau des calculs de similarité.

Un deuxième problème concerne la pondération utilisée pour les nouveaux termes de la requête. Cette pondération peut être effectuée de deux manières différentes :

– Nous avons utilisé dans un premier temps les poids déjà calculé pour la sélection. Cette utilisation repose sur l'intuition que si un terme a un poids élevé (c'est à dire est pertinent selon les éléments jugés et une requête donnée), c'est qu'il a une importance élevée aux yeux de l'utilisateur.

– Une deuxième hypothèse qui contredit la première est qu'un terme peut avoir un score moins intéressant qu'un autre car il n'apparaît pas autant dans l'ensemble considéré des éléments jugés pertinents, alors qu'il peut avoir une importance supérieure si on considère un ensemble plus étendu d'éléments jugés. Nous proposons alors de recalculer les poids des éléments déjà sélectionnés.

Dans ce calcul nous tenons compte des caractéristiques des mots clés dans toute la collection. Pour ce faire, nous considérons la fréquence du terme dans la collection ainsi que le facteur *idf* défini précédemment et son analogue *ief* défini pour exprimer la spécificité d'un terme dans l'ensemble des éléments de la collection :

$$ief(t_j) = \log \left( \frac{|N|}{|e_j|} + 1 \right)$$

Avec  $|e_j|$  le nombre des éléments dans lequel apparaît le terme  $t_j$  et  $|N|$  le nombre d'éléments dans la collection. Le poids  $Pds_j$  du terme  $t_j$  à réinjecter est la normalisation des poids entre 0 et 1 de la fonction suivante :

$$Pds_j = tf_j \times idf_j \times ief_j \quad [10]$$

Si un terme existe déjà dans la requête initiale il ne sera pas réinjecté.

## 4. Environnement d'évaluation

### 4.1. Système de recherche XFIRM

Nous avons testé notre approche en utilisant le système de recherche XFIRM [SAU 06]. Ce système est basé sur une méthode de propagation de pertinence dans la représentation en arbre des documents XML. Des valeurs de pertinence sont d'abord calculées pour les différents noeuds feuilles (c'est à dire les noeuds contenant du texte). Ces valeurs sont par la suite propagées vers les noeuds ancêtres.

Si on considère un noeud feuille ( $nf$ ) et une requête  $q$  composée de  $n$  mots clés (c'est à dire une requête de type CO), la valeur de pertinence du noeud feuille sera calculée en utilisant la fonction de similarité  $RSV(q, nf)$  suivante :

$$RSV(q, nf) = \sum_{j=1}^n w_j^q \times w_j^{nf} \quad [11]$$

Où :  $w_j^q = tf_j^q$  est le poids du terme  $j$  dans la requête  $q$  et  $w_j^{nf} = tf_j^{nf} \times idf_j \times ief_j$  est le poids du terme  $j$  dans le noeud feuille  $nf$ .

Cette pondération permet d'exprimer à la fois l'importance des termes dans les noeuds feuilles et dans tout le document.  $tf_j^q$  et  $tf_j^{nf}$  sont respectivement la fréquence du terme  $j$  dans la requête  $q$  et dans le noeud feuille  $nf$ ,  $idf_j = \log(|D|/(|d_j| + 1)) + 1$ , avec  $|D|$  le nombre total des documents dans la collection,  $|d_j|$  le nombre des documents contenant le terme  $j$ , et  $ief_j$  est la fréquence inverse du terme  $j$ , c'est à dire  $\log(|N|/|nf_j| + 1) + 1$ , où  $|nf_j|$  est le nombre de noeuds feuille contenant le terme  $j$  et  $|N|$  est le nombre total des noeuds feuilles dans la collection.

La valeur de pertinence d'un noeud interne  $n$  (différent d'un noeud feuille), définie par  $p_n$ , est calculée comme suit :

$$p_n = |L_n^p| \sum_{k=1..N} \alpha^{dist(n, nf_k)-1} * RSF(q, nf_k) \quad [12]$$

Où  $dist(n, nf_k)$  est la distance entre le noeud  $n$  et le noeud feuille  $nf_k$  dans l'arbre du document, (c'est à dire le nombre d'arcs nécessaires pour atteindre le noeud  $n$  en partant du noeud feuille  $nf_k$ ),  $|L_n^p|$  est le nombre de noeuds feuilles descendants du noeud  $n$  ayant un score différent de zéro,  $nf_k$  est un noeud feuille descendant du noeud  $n$ ,  $N$  est le nombre total des noeuds feuilles descendants de  $n$ , et  $\alpha \in [0..1]$ .

#### 4.2. Mesures d'évaluation

Nous évaluons nos propositions grâce à la campagne d'évaluation INEX (*INitiative for the Evaluation of XML Retrieval*) [INE05b]. Cette dernière propose une collection composée de plus de 17000 documents provenant de 21 revues IEEE Computer Society parues de 1995 à 2004. Les jugements de pertinence pour chaque requête sont effectués par les différents participants. Deux dimensions sont utilisées pour définir la pertinence : l'*exhaustivité* (e) et la *spécificité* (s). L'*exhaustivité* est mesurée selon une échelle à 4 niveaux :

e=2 exhaustivité élevée, e=1 exhaustivité moyenne, e=0 pas d'exhaustivité et e=? élément trop petit.

La *spécificité* est mesurée dans un intervalle continu [0,1] où s=1 représente un élément totalement spécifique.

Les mesures d'évaluation utilisées durant la campagne 2005 sont basées sur les mesures XCG et ep/gr [INE05b]. Pour obtenir des résultats de performance avec ces mesures, les 2 dimensions de pertinence (exhaustivité et spécificité) sont agrégées en une seule valeur. Deux types de fonction d'agrégation sont utilisées :

– une agrégation "stricte" pour évaluer si un SRI est capable de retrouver des éléments très spécifiques et très exhaustifs

$$f_{strict}(e, s) = \begin{cases} 1 & \text{si } e = 2 \text{ et } s = 1 \\ 0 & \text{sinon} \end{cases} \quad [13]$$



– une agrégation "généralisée" pour évaluer les éléments selon leur degré de pertinence

$$f_{generalise}(e, s) = e * s \quad [14]$$

Nous utilisons pour les évaluations les mesures proposées par G. Kazai et M. Lalmas [KAZ 05] :

–  $MANxCG[i]$  : la moyenne des Gains Cumulés Normalisés par rapport au gain idéal ( $i$  est le rang des éléments cumulés). Pour un rang donné  $i$ , le gain cumulé  $nxCG[i]$  reflète le gain relatif de l'utilisateur accumulé jusqu'à ce rang, comparé à ce qu'il aurait dû atteindre si le système avait produit une liste triée optimale.

–  $MAep$  : la moyenne non interpolée d'effort-précision. Par analogie au gain cumulé, on définit l'effort-précision ( $ep(r)$ ). C'est le rapport entre le rang pour lequel le gain cumulé est atteint par la courbe idéale et le rang pour lequel le gain cumulé est atteint par le système.

La valeur 1 correspond à une performance idéale, pour laquelle l'utilisateur effectue un minimum d'effort pour atteindre un niveau de gain donné.

Pour évaluer les performances des techniques de la reformulation, on a défini l'amélioration absolue  $AA$  calculée comme suit :

$$AA = (MAP(RF - run) - MAP(base - run)) \quad [15]$$

Pour comparer l'amélioration relative, on a défini l'amélioration relative  $AR$  calculée comme suit :

$$AR = (MAP(RF\_run) - MAP(base - run)) / MAP(base\_run) \quad [16]$$

où  $MAP$  dans le cas de  $nxCG$  est  $MANxCG@1500$  et dans le cas de  $ep/gr$  est  $MAep$ ,  $RF\_run$  est le résultat obtenu avec reformulation, et  $base\_run$  est le résultat de base.

## 5. Expérimentations

Dans cette section, nous allons détailler les différentes expérimentations que nous avons réalisées afin d'étudier l'impact des différents critères définis précédemment.

Pour évaluer notre approche, nous utilisons dans un premier temps la stratégie proposée par INEX. Elle consiste à considérer les jugements des 20 premiers éléments retournés par le SRI, à appliquer notre approche et à relancer la recherche avec les nouvelles requêtes reformulées. On sélectionne ensuite les éléments jugés à partir de l'ensemble des éléments retournés et on les place en tête des résultats. La comparaison des résultats ne peut s'établir qu'en utilisant plus que le top 20 des éléments retournés, en considérant des mesures comme  $MANxCG[25]$ ,  $MANxCG[50]$ ,  $MAep$ ... Nous nous intéressons à la pertinence stricte, et nous considérons les éléments jugés strictement pertinents (exhaustivité  $e=2$  et spécificité  $s > 0.9$ ).

Nous utilisons les résultats de base obtenus par le système de recherche XFIRM [SAU 06] et nous appliquons les mêmes paramètres de recherche. Les résultats de

base correspondent aux meilleurs résultats obtenus pour les requêtes de type CO où  $\alpha = 0.1$  (équation 1), en considérant la fonction d'agrégation stricte. Dans le reste de l'article nous ne considérons par conséquent que la mesure d'agrégation stricte (équation 13).

5.1. Impact du contexte pour la sélection des termes expressifs

Dans ce paragraphe, nous allons comparer les résultats de reformulation obtenus par sélection des termes expressifs selon l'équation 1 et ceux obtenus en considérant le contexte d'un terme dans les éléments pertinents pour une requête donnée (équation 8). Nous allons aussi évaluer la variation des performances selon le nombre de termes réinjectés (nous allons varier de 1 à 10 termes). Le tableau 1 présente les pourcentages des Améliorations Relatives (AR) pour différentes mesures considérées avec une fonction d'agrégation stricte. Nous désignons par Prob-3 (ligne 2) les résultats obtenus par une pondération des termes par l'équation 1 et qui correspond à la réinjection de 3 termes expressifs. Le reste du tableau présente les résultats obtenus pour différents nombres de termes expressifs désignés par Cont-i (i nombre de termes réinjectés) en appliquant l'équation 8 pour l'extraction et la pondération des termes.

Tableau 1. Impact du nombre de termes réinjectés par combinaison de la méthode probabiliste et de l'utilisation du contexte des termes

	MAep strict (%)	MANxCG [25] strict (%)	MANxCG [50] strict (%)
Prob-3	-10.13	2.39	-14.7
Cont-1	-3.16	<b>3.49</b>	-8.55
Cont-2	-3.80	<b>2.94</b>	-9.76
Cont-3	-3.80	<b>2.94</b>	-9.88
Cont-4	-3.80	<b>3.49</b>	-9.76
Cont-5	-4.43	<b>3.49</b>	-10.00
Cont-6	-5.70	<b>2.94</b>	-10.24
Cont-7	-4.43	<b>2.39</b>	-10.48
Cont-8	-4.43	<b>2.94</b>	-10.36
Cont-9	-4.43	<b>3.49</b>	-10.24
Cont-10	-5.06	<b>3.49</b>	-10.24

Nous remarquons des valeurs positives de AR si on considère la mesure MANxCG[25] ce qui reflète un impact positif du contexte sur les 5 éléments classés après le top 20 (puisque'on garde les même éléments que les résultats de base dans le top 20). Les résultats obtenus en considérant le contexte sont en général plus intéressants que ceux obtenus par simple calcul de probabilité conditionnelle (voir ligne 5 du tableau 1). Cependant, d'une manière générale, les résultats ne sont pas assez significatifs. En effet, le tableau 1 ne présente que des moyennes alors que selon la stratégie d'INEX

que l'on a appliquée, certaines requêtes ne sont pas reformulées ou mal reformulées puisque que le top 20 des éléments considérés ne contient pas assez d'éléments pertinents.

Nous proposons alors pour mieux étudier l'impact de l'extraction des termes en considérant le contexte, de faire varier le nombre d'éléments jugés et d'appliquer la reformulation résiduelle [CHA 71]<sup>4</sup>. Nous considérons dans ce qui suit la réinjection de 3 termes expressifs puisque les valeurs obtenues se rapprochent plus de la moyenne des valeurs obtenues pour d'autres nombres de termes réinjectés.

#### 5.1.1. Variation du nombre d'éléments jugés

Dans ce paragraphe, nous appliquons l'équation 8 pour l'extraction et la pondération des termes. Nous varions le nombre des éléments jugés de 10 à 100.

Le tableau 2 représente les moyennes des améliorations relatives.  $N_k$  désigne  $k$  éléments jugés.

Nous remarquons dans le tableau 2 que les résultats sont plus significatifs. En effet, les

**Tableau 2.** Impact de la variation du nombre d'éléments jugés sur la reformulation

	MAep strict (%)	MANxCG [10] strict (%)	MANxCG [25] strict (%)	MANxCG [50] strict (%)
N10	-11.39	<b>82.30</b>	27.57	4.94
N20	-17.72	76.07	4.41	-25.90
N30	-34.18	75.41	-25.00	-59.40
N40	-79.11	12.13	-49.08	-70.84
N50	-79.75	33.44	-43.57	-70.12
N60	-78.48	-9.51	-58.09	-74.82
N70	-79.75	5.90	-50.92	-78.80
N80	-81.01	-6.56	-57.72	-74.22
N90	-85.44	-7.21	-57.54	-77.59
N100	-80.38	59.34	-39.71	-69.16

améliorations obtenues en considérant la mesure MANxCG[10] sont environ de 75% pour 10, 20 et 30 éléments jugés. Les améliorations sont en baisse dès qu'on considère plus que 30 éléments jugés. L'amélioration décroissante en fonction du nombre d'éléments jugés peut être expliquée par le fait que chaque fois qu'on considère un nombre plus élevé d'éléments jugés et en appliquant la reformulation résiduelle, en éliminant les  $k$  éléments jugés il y a une perte plus importante d'éléments pertinents. En effet la meilleure amélioration est obtenue pour 10 éléments jugés. L'exception observée pour le top 100 permet d'émettre l'hypothèse qu'il n'existe pas un nombre d'éléments jugés approprié pour toutes les requêtes mais qu'il dépend de chacune. Cette hypothèse sera détaillée dans le paragraphe 5.4.

4. La reformulation résiduelle consiste à éliminer les éléments déjà jugés du résultat final.

### 5.1.2. Impact du nombre d'éléments jugés pertinents

Nous proposons dans ce paragraphe une autre stratégie d'évaluation. Cette dernière consiste à considérer un nombre fixe d'éléments jugés pertinents et à appliquer la reformulation résiduelle. D'après des expérimentations faites dans [HLA 06b], le nombre d'éléments jugés pertinents le plus satisfaisant est 3 sachant que la moyenne des éléments jugés pertinents dans le top 10 (resp. dans le top 20) est 1 (resp. 1,8). Dans le tableau 3, nous représentons les améliorations relatives de la deuxième stratégie (désignée par P3-AR) d'évaluation comparées avec celles obtenues en considérant 10 éléments jugés (désignée par N10-AR) et ainsi celles obtenues en appliquant le protocole d'INEX (désignée par INEX-AR).

**Tableau 3.** Impact du nombre d'éléments jugés pertinents

	MAep strict (%)	MANxCG [10] strict (%)	MANxCG [25] strict (%)	MANxCG [50] strict (%)
INEX-AR	-3.80	0.00	2.94	-9.88
N10-AR	-17.72	82.30	27.57	4.94
R3-AR	13.29	<b>230.16</b>	38.42	-26.27

Les valeurs observées montrent l'intérêt de la prise en compte du nombre d'éléments jugés pertinents dans la reformulation. En effet l'amélioration de MANxCG[10] obtenue en appliquant la deuxième stratégie est de 230%. Ceci présente un indice très significatif au niveau de la performance du système. Cette amélioration traduit le fait que la reformulation de la requête permet de retrouver deux fois plus d'éléments pertinents dans le top 10. Pour la même stratégie, on observe que AR de MANxCG[50] est négative. En effet, on perd beaucoup plus que si on ne considère que 10 éléments pour le jugement, pour la même raison, on observe une amélioration moins bonne pour la mesure de MAep. Nous constatons aussi que les résultats retrouvés par les différentes stratégies ne sont pas contradictoires. En effet si on considère la mesure de MANxCG[25], cette dernière montre dans le cas de la stratégie d'INEX, que la reformulation permet de retrouver plus d'éléments pertinent parmi les 5 éléments qui suivent le top 20 ; ce que confirment les mesures MANxCG[10] et MANxCG[25] en utilisant les 2 autres stratégies. Dans ce qui suit, nous considérons la deuxième stratégie.

### 5.2. Impact du bruit

Nous nous proposons ici d'évaluer si le facteur bruit permet de raffiner la sélection des termes et donc d'augmenter les performances du système de recherche. Nous avons appliqué le poids calculé selon l'équation 9 désigné par Poids Ajusté. Les résultats sont présentés dans le tableau 4 comparés aux résultats obtenus en appliquant le calcul de poids défini par l'équation 8 désigné par Poids Contextuel et ceux obtenus en appliquant que la probabilité conditionnelle (équation 1) désigné par Prob.

Le facteur bruit ne semble pas montrer d'intérêt pour la reformulation. Toutes les

**Tableau 4.** *Impact du critère bruit sur la reformulation*

	MAep strict (%)	MAxCG [10] strict (%)	MAxCG [25] strict (%)	MAxCG [50] strict (%)
Prob(eq1)	57.59	151.80	43.20	-4.46
Poids contextuel (eq8)	13.29	230.16	38.42	-26.27
poids Ajusté (eq9)	-8.86	103.28	-2.76	-38.19

améliorations sont moins bonnes que celles obtenues en considérant le contexte avec la probabilité conditionnelle. Ceci peut être dû aux paramètres utilisés dans la mesure du bruit, ou à la collection considérée.

### 5.3. Impact de différentes pondérations

Dans ce paragraphe, nous allons tester l'importance du choix de pondération des termes à réinjecter. En effet, dans les expérimentations précédentes nous avons utilisé les poids calculés pour l'extraction pour pondérer les termes de la nouvelle requête. Nous allons comparer les résultats obtenus avec ceux que l'on peut obtenir si on considère la même fonction d'extraction mais d'autres poids pour calculer l'importance des termes (équation 10).

Les résultats sont représentés dans le tableau 5. Poids contextuel désigne les résultats obtenus par pondération des termes en appliquant l'équation 8 et Autre désigne l'application de l'équation 10 pour la pondération des termes.

**Tableau 5.** *Impact du choix de la fonction de pondération*

	MAep strict (%)	MAxCG [10] strict (%)	MAxCG [25] strict (%)	MAxCG [50] strict (%)
Poids contextuel (eq 8)	13.29	230.16	38.42	-26.27
Autre (eq 10)	-46.84	65.90	-16.73	-46.84

On remarque que les améliorations sont moins bonnes en utilisant la nouvelle fonction de pondération. Le choix de la fonction de pondération des termes est donc un paramètre important de la phase de reformulation.

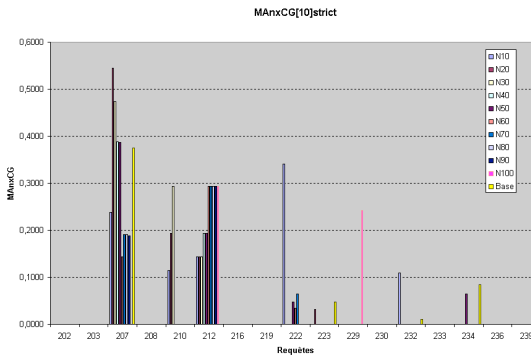
### 5.4. Discussion

Nous avons constaté que les résultats obtenus en variant le nombre de termes expressifs réinjectés ne présentent pas de différences significatives. Ceci peut être expliqué, comme déjà prouvé dans la littérature en RI classique [RUT 99], par le fait que le nombre de termes le plus approprié à rajouter à une requête dépend essentiellement

de la requête. Plus précisément, ceci dépend de la taille de la requête. Un nombre important des termes peut avoir un impact positif sur une longue requête alors qu'il peut y avoir des pertes d'information si on applique le même nombre de termes sur une requête courte. Nous envisageons dans nos futurs travaux de spécifier un nombre de termes à réinjecter qui sera proportionnel à la taille de la requête. Nous pourrions également traiter ce problème en considérant, la difficulté de la requête estimée en fonction de la nature des mots, le taux d'ambiguïté, etc.

La deuxième remarque concerne l'impact du nombre d'éléments jugés utilisés pour la reformulation. Les résultats ne permettent pas de conclure sur le meilleur nombre de jugements à considérer. Ceci dépend également des requêtes. On peut mieux l'observer sur la figure 1. qui présente la valeur de  $MANxCG[10]$  pour les différentes requêtes. On remarque que si on considère le top 60, la moyenne des améliorations est négative alors qu'il permet d'améliorer la  $MANxCG[10]$  de 0 dans les résultats de base à **0.3**. La deuxième stratégie proposée, bien qu'elle permette de mettre en évidence l'amélioration de  $MANxCG[10]$ , est la cause d'une perte importante pour  $MANxCG[50]$ . Une première intuition est de trouver un compromis entre le nombre d'éléments pertinents et le nombre d'éléments jugés utilisés (i.e. fixer un nombre d'éléments pertinents sans dépasser un certain nombre d'éléments jugés pour diminuer la perte). Nous avons évalué cette idée (3 éléments pertinents sans dépasser 30 éléments jugés) en appliquant la pondération des termes avec un poids différent de celui de l'extraction (équation 10) : l'amélioration de  $MANxCG[10]$  est passée de 65% à 115%. Cette observation est aussi vraie pour les autres mesures. On pourra fixer ces paramètres dans nos futurs travaux.

Nous avons également constaté qu'il est intéressant de considérer le facteur contexte



**Figure 1.** Variation du nombre d'éléments jugés.

et que le bruit avait un effet négatif sur la sélection des termes. Nous allons essayer dans nos futurs travaux d'étudier davantage les paramètres de ce facteur et de trouver une meilleure fonction d'agrégation.

Enfin, nous avons montré que la reformulation ne dépend pas seulement des termes extraits mais aussi de leur pondération.

## 6. Conclusion

Dans cet article, notre objectif était de raffiner les requêtes de type CO afin d'obtenir de meilleures performances du système de recherche et surtout de mieux répondre aux besoins de l'utilisateur en partant d'une requête assez naïve sans termes techniques ni préférences. Nous avons proposé une approche qui permet de retrouver les termes qui peuvent exprimer le besoin de l'utilisateur en fonction de la probabilité conditionnelle simplifiée et du facteur contexte d'un terme dans les éléments pertinents pour une requête donnée. Nous avons également proposé une nouvelle stratégie d'évaluation qui nous a permis de mieux analyser les résultats et de mettre en évidence les impacts des différents facteurs de notre modèle. Nous avons aussi constaté l'intérêt du choix des fonctions de pondération des termes. En général, les résultats sont assez satisfaisants. Ces approches sont notamment applicables sur des requêtes comportant des contraintes structurelles ; reste à les tester. Dans nos futurs travaux nous allons affiner cette approche en variant quelques paramètres détaillés dans la section 5.4 et en appliquant cette approche sur des éléments pertinents ayant des structures pertinentes [HLA 06c].

## 7. Bibliographie

- [CHA 71] CHANG Y., CIRILLO C., RAZON J., « Evaluation of feedback retrieval using modifying freezing, residual collection & test and control groups », *The SMART retrieval system : experiments in automatic document processing*, Prentice-Hall, 1971, p. 355-370.
- [CHI 96] CHIARAMELLA Y., MULHEM P., FOUREL F., « A Model for Multimedia Information Retrieval », rapport, 1996, Technical report, FERMI ESPRIT BRA 8134, University of Glasgow.
- [CRO 04] CROUCH C., MAHAJAN A., BELLAMKONDA A., « Flexible XML Retrieval Based on the Vector Space Model », *INEX 2004 Workshop Proceedings*, Germany, December 2004, p. 292,302.
- [EFT 00] EFTHIMIADIS E., « Interactive query expansion : a user based evaluation in relevance feedback environment », *Journal of the American Society for Information Science*, vol. 51, n° 11, 2000, p. 989,1003.
- [HAR 92] HARMAN D., « Ranking algorithms. », *Information retrieval : data structures & algorithms*, 1992, p. 363, 392.
- [HLA 06a] HLAOUA L., SAUVAGNAT K., « Structure-Oriented Relevance Feedback in XML Retrieval. », *InSciT2006*, Merida, Espagne, October 2006.
- [HLA 06b] HLAOUA L., SAUVAGNAT K., BOUGHANEM M., « Réinjection de structures pour la reformulation de requêtes en RI structurée. », *INFORSID 2006*, Hammet, Tunisie, Juin 2006, p. 435,450.

- [HLA 06c] HLAOUA L., SAUVAGNAT K., BOUGHANEM M., « A structure-oriented relevance feedback method for XML retrieval », *CIKM'06*, Arlington, Virginia, USA., November 2006.
- [INE04] *INitiative for the Evaluation of XML Retrieval*, disponible sur <http://inex.is.informatik.uni-duisburg.de:2004/internal/results/rf-results.html>, 2004.
- [INE05a] *INitiative for the Evaluation of XML Retrieval*, disponible sur <http://inex.is.informatik.uni-duisburg.de:2005/tracks/rel/>, 2005.
- [INE05b] *INitiative for the Evaluation of XML Retrieval*, disponible sur <http://inex.is.informatik.uni-duisburg.de:2005/>, 2005.
- [KAZ 05] KAZAI G., LALMAS M., « INEX 2005 Evaluation Metrics », *INEX 2005 Workshop Proceedings*, Germany, November 2005, p. 401,406.
- [K.S 72] K.SPARK-JONES, « A statistical interpretation of term specificity and its application in retrieval », *Journal of documentation*, , n° 28, 1972, p. 11, 20.
- [LIS 03] LIST J. A., MIHAJLOVIC V., DE VRIES A. P., RAMIREZ G., « The TIJAH XML-IR system at INEX 2003 », *Proceedings of INEX 2003 Workshop*, Dagstuhl, Germany, December 2003, p. 102,109.
- [MAS 04] MASS Y., MANDELBROD M., « Relevance Feedback for XML Retrieval », *INEX 2004 Workshop Proceedings*, Germany, December 2004, p. 303,310.
- [MIH 04] MIHAJLOVIC V., RAMIREZ G., DE VRIES A., HIEMSTRA D., BLOK H., « TIJAH at INEX 2004 Modeling Phrases and Relevance Feedback », *INEX 2004 Workshop Proceedings*, Germany, December 2004, p. 276,291.
- [MIH 05] MIHAJLOVIC V., RAMIREZ G., WESTERVELD T., BLOCK H., DE VRIES A., HIEMSTRA D., « TIJAH Scratches INEX 2005 Vague Element Selection, Overlap, Image Search, relevance Feedback, and Users », *INEX 2005 Workshop Proceedings*, Dagstuhl, Germany, November 2005, p. 54,71.
- [ROB 76] ROBERTSON S., SPARK-JONES J., « Relevance weighting of search terms », *Journal of the American Society for Information Science*, vol. 27, n° 3, 1976, p. 129, 146.
- [ROC 71] ROCCHIO J., « Relevance feedback in information retrieval », *The SMART retrieval system-experiments in automatic document processing*, Prentice Hall Inc, 1971, p. 313,323.
- [RUT 99] RUTHVEN I., LALMAS M., « Selective Relevance Feedback Using Term Characteristics », *CoLIS 3, Proceedings of the Third International Conference on Conceptions of Library and Information Science.*, , 1999.
- [RUT 02] RUTHVEN I., LALMAS M., RIJSBERGEN K., « Combining and selecting characteristics of information use », *JASIST*, vol. 53, n° 5, 2002, p. 378,396.
- [SAL 83] SALTON G., MCGILL M., « Introduction to modern information retrieval », *McGraw-Hill Book Company*, , 1983, page 1.
- [SAU 06] SAUVAGNAT K., BOUGHANEM M., « Propositions pour la pondération des termes et l'évaluation de la pertinence des éléments en recherche d'information structurée », *CO-RIA 2006*, Lyon, mars 2006, p. 29-40.
- [SCH 05] SCHENKEL R., THEOBALD M., « Relevance Feedback for Structural Query Expansion », *INEX 2005 Workshop Proceedings*, Germany, November 2005, p. 260,272.