# Identifier des critères distinguant les requêtes pour améliorer la RI-XML

Gilles Hubert*, Josiane Mothe*†

*Institut de Recherche en Informatique de Toulouse, Equipe SIG/EVI, Université Paul Sabatier, 118 route de Narbonne, F-31062 Toulouse cedex 9, {hubert, mothe}@irit.fr
† ERT34, Institut Universitaire de Formation des Maîtres, 56 avenue de l'URSS, F-31400 Toulouse

***Résumé***

*Cet article s'appuie sur un moteur de recherche d'information dans des collections de documents XML qui est configurable pour s'adapter à différents contextes de recherche. Au regard des résultats obtenus lors de différentes campagnes d'évaluations, notre moteur présente une efficacité globale satisfaisante. Cependant, l'efficacité de la méthode est inégale d'une requête à l'autre. De plus, différentes configurations de la méthode ne conduisent pas à la même efficacité pour la même requête. Cette constatation se retrouve pour bon nombre d'autres approches. Nous introduisons dans cet article différents critères pour distinguer différents types de requête. Nous tentons de recenser des critères ayant une influence sur notre méthode et de définir la configuration la plus adaptée à un type de requête. Nous étudions également d'autres méthodes suivant ces mêmes critères pour voir si les approches se comportent différemment face au même type de requêtes. Ceci permet d'estimer si une combinaison d'approches pourrait améliorer les recherches d'information XML. Cet article s'appuie sur des expérimentations menées sur les jeux de tests proposés lors des campagnes INEX d'évaluations des systèmes de recherche d'information XML.*

*Mots-clés : Recherche d'information XML, distinction de requêtes.*

***Abstract***

*This paper is based on a search engine for XML information retrieval. This engine can be configured in order to suit to different retrieval contexts. Regarding results obtained in different evaluation initiatives the effectiveness of our search engine is globally satisfactory. However, the method effectiveness is variable from one query to another. Moreover, different configurations of the method do not lead to the same effectiveness for a given query. This observation is found for numerous approaches. We introduce in this paper various criteria to identify query types. We try to determine criteria influencing the method and we try to define the method configuration adapted to a query type. We also study other methods according to the same criteria to see if these approaches have different behaviours for the same type of queries. This allows estimating if a combination of different approaches could improve XML information retrieval. This article is based on experiments that use testbeds provided in the INEX initiative for the evaluation of XML retrieval.*

*Keywords: XML Information Retrieval, query type distinction.*

# 1. INTRODUCTION

De nombreuses propositions de systèmes de Recherche d'Information (RI) combinant recherche textuelle et recherche sur la structure ont vu le jour répondant à l'utilisation croissante du langage XML (eXtensible Markup Language) [3] pour représenter les documents et à ses particularités. Dans le cadre de documents structurés, les unités d'indexation et de recherche ne sont plus seulement les documents entiers comme généralement en RI. La structure hiérarchique des documents XML implique également que plusieurs niveaux de granularité au sein d'un document peuvent correspondre à un besoin d'information. Dans ce contexte, l'utilisateur peut également choisir la granularité des composants XML qu'il souhaite pour la réponse et indiquer des contraintes de structure sur le contenu des éléments à restituer (contraintes de localisation). Ces changements ont un impact sur l'efficacité des systèmes de RI (SRI) dans le cadre XML et sur la manière d'évaluer cette efficacité. INEX[1] est un programme qui s'intéresse à l'évaluation de SRI permettant la manipulation de documents XML. Bien que fournissant des

[1] http://inex.is.informatik.uni-duisburg.de/

évaluations par requête, les études s'appuient généralement sur des résultats globaux c'est-à-dire en calculant des moyennes de performance sur un ensemble de requêtes. Ce principe cache les disparités des résultats obtenus.

Dans cet article, nous tentons de dégager des caractéristiques des requêtes de recherche XML afin de distinguer des types de requêtes. Nous étudions certaines caractéristiques des requêtes pour un ensemble de configurations de notre méthode de recherche. Nous tentons d'identifier des caractéristiques pour lesquelles notre approche est plus ou moins efficace notamment pour cibler dans quelle direction faire évoluer notre méthode. Nous comparons également plusieurs configurations pour tenter de mettre en évidence des différences d'efficacité pour un même type de requête et éventuellement pouvoir appliquer la configuration la plus efficace pour chaque type de requête. Dans cette même optique, nous étudions également d'autres approches pour estimer si les critères de distinction de requête sont également utilisables pour dégager les forces et faiblesses de chaque approche et éventuellement pouvoir les combiner pour améliorer les recherches d'information XML (RI-XML).

Cet article est organisé comme suit. La section 2 présente un aperçu des différents travaux liés à la RI-XML et à des études sur les requêtes. Dans la section 3, nous introduisons les principaux aspects de notre méthode de recherche. Les critères de distinction des requêtes que nous pouvons identifier sont présentés en section 4. La section 5 décrit les expérimentations menées suivant un sous-ensemble de critères pour définir les types de requêtes conduisant à de bons résultats ou de moins bons et la configuration de notre méthode la plus appropriée à un certain type de requête. Cette section présente également le comportement d'autres approches au regard des critères. La section 6 conclut cet article.

## 2. TRAVAUX DU DOMAINE

Les propositions relatives à la RI-XML s'appuient principalement sur des modèles probabilistes [18][20][24], le modèle vectoriel [6][9][15] ou sur un système de gestion de bases de données [8][16]. Utilisant le modèle vectoriel, [15] s'appuie sur la mesure de similarité cosinus et plusieurs index correspondant à différents niveaux de granularité et fusionne ensuite les listes de résultats des différents niveaux. Dans [6] un vecteur étendu représentant un document regroupe un ensemble de sous-vecteurs correspondant à différentes classes d'information (par exemple, nom d'auteur, résumé). La similarité entre vecteurs étendus est définie comme

une combinaison linéaire des similarités entre sous-vecteurs correspondants. [9] propose de générer automatiquement à la volée l'espace vectoriel correspondant à une requête à partir de l'index des documents XML au niveau des types d'éléments de base. La mesure de similarité tf.ief dérivée de tf.idf est utilisée gérant des éléments XML plutôt que des documents entiers. Dans le cadre des approches probabilistes, un modèle de langage hiérarchique est proposé par [18]. Les documents XML sont représentés par des arbres où chaque nœud correspond à une balise dans le document. Chaque composant XML est estimé en utilisant une interpolation linéaire de son contenu, des modèles de ses nœuds enfants et du modèle de son nœud parent. [24] propose un modèle de langage multinomial avec lissage utilisant des index à différents niveaux et normalisation de taille. L'approche présentée dans [20] utilise des réseaux bayésiens. Le réseau décrit la hiérarchie de documents. Les scores des éléments sont calculés récursivement au travers du réseau du plus grand élément racine du réseau jusqu'aux plus petits éléments constituant les feuilles. Les approches basées sur les langages de requêtes de base de données introduisent quant à elles, comme par exemple XIRQL [8], des notions de prédicats vagues et d'approximation structurelle (par exemple, similarités entre noms d'éléments). XIRQL utilise également un principe de classement de résultat basé sur un calcul probabiliste utilisant une pondération des termes d'indexation des documents et des requêtes. [16] suit une architecture de base de données à trois niveaux : conceptuel, logique et physique. Le système est principalement étendu au niveau logique par l'introduction d'une algèbre prenant en compte la structure spécifique des données XML, la manipulation de scores et le classement des résultats. La comparaison des différentes approches reste une tâche délicate. Dans cette optique, INEX est une initiative pour offrir un cadre pour l'évaluation des systèmes de recherche d'information dans des collections XML notamment en termes de précision des résultats. Le cadre INEX n'a pas permis de mettre en avant un type d'approche (probabiliste, vectoriel ou base de données) pour la recherche XML. Parmi les meilleures évaluations obtenues lors des campagnes INEX, se trouvent des systèmes basés sur un modèle vectoriel comme [15], sur un modèle probabiliste comme [24] ou sur une architecture de base de données comme [16].

Notre approche s'inscrit dans les méthodes à base de modèle vectoriel. Cependant plutôt qu'une mesure de similarité habituelle, elle s'appuie sur un principe de prise en compte directe des contributions individuelles de chaque élément d'une requête. Notre approche n'est pas définie spécifiquement pour la recherche XML. Elle est définie de manière modulaire et paramétrable afin de pouvoir être adaptée à différents

contextes de recherche d'information. Cette approche a donné des résultats intéressants lors de différentes campagnes d'évaluation INEX.

Peu de travaux en recherche d'information XML se sont intéressés à l'analyse de résultats d'expérimentations pour dégager des types de requêtes traités de manière satisfaisantes et d'autres traités moins efficacement. [19] divisent les requêtes INEX en deux catégories 'Broad' et 'Narrow' suivant la spécificité (niveau article peu spécifique vs niveau paragraphe très spécifique) des éléments jugés majoritairement pertinents. L'étude montre que deux approches différentes sont plus efficaces pour chacune des catégories de requêtes. [24] tente de distinguer les requêtes INEX selon le nombre d'éléments constituant les jugements de pertinence, la taille et le degré d'imbrication des éléments et ce afin d'expliquer les différences d'efficacité obtenues d'une requête à l'autre. Ces classifications possèdent néanmoins le désavantage de se baser sur une étude des jugements de pertinence.

Dans le cadre d'une recherche d'information non XML différents travaux se sont intéressés à l'identification de types de requêtes. [7] tente de classifier les requêtes et les systèmes en fonction des résultats obtenus par les différents participants à la tâche « Nouveauté » de TREC [10] notamment pour caractériser les requêtes difficiles. Dans le cadre de la tâche « Robustesse » de TREC [27], [29] définit deux algorithmes de prédiction de difficulté d'une requête pour améliorer les performances du moteur de recherche. Sur la même collection, [13] combine les idf (fréquence documentaire inverse) et les fréquences moyennes des termes des requêtes pour prédire les meilleures et plus mauvaises requêtes principalement pour des requêtes courtes.

Nous tentons dans cet article de distinguer différents types de requêtes en se basant uniquement sur les caractéristiques de ces requêtes. Nous utilisons ce processus pour tenter d'améliorer l'efficacité de notre méthode de recherche d'information XML.

## 3. NOTRE METHODE DE RI-XML

La méthode de recherche est principalement basée sur la représentation des éléments XML et des requêtes sous forme de vecteurs. La correspondance entre requête et élément XML n'est pas basée sur une mesure de similarité 'classique'. Elle s'appuie plutôt sur la contribution directe des termes définissant la requête modulée en fonction de

l'importance du terme dans la requête et éventuellement d'autres éléments tels que des contraintes de structure.

Les documents sont représentés sous forme de vecteurs <terme, nombre d'occurrences, chemin de localisation>. Les termes sont extraits automatiquement des éléments textuels avec leur nombre d'occurrences dans le composant et leur localisation dans les documents XML en utilisant une notation de type Xpath [4]. L'extraction de termes met en œuvre notamment la suppression des mots vides et des traitements optionnels comme la radicalisation. Un processus d'extraction automatique similaire est défini pour les requêtes représentées également sous formes de vecteurs <terme, nombre d'occurrences, contrainte de localisation, préférence>. Les notions de contrainte de localisation et de préférence sont précisées plus loin.

## 3.1. Fonction de score

La fonction de score qui estime la correspondance entre un élément XML et une requête tient compte principalement de trois facteurs :

- l'importance de chaque terme de la requête dans l'élément XML,
- l'importance de chaque terme dans la requête,
- le niveau global de représentation de la requête dans l'élément XML.

$$Score(T,E) = \left( \sum_{\forall t \in T} Occ(t,E) \cdot \frac{Occ(t,T)}{Occ(T) \cdot NbElts(t)} \right) \cdot \varphi^{\left(\frac{NbT(T,E)}{NbT(T)}\right)}$$

où T est la requête et E est un élément XML

*Occ(t,E)* Ce facteur mesure l'importance du terme t dans l'élément XML E. Il correspond au nombre d'occurrences du terme t dans E.

$\frac{Occ(t,T)}{Occ(T) \cdot NbElts(t)}$ Ce facteur mesure l'importance du terme t dans la représentation de la requête T. Occ(t,T) correspond au nombre d'occurrences du terme t dans la requête T et Occ(T) correspond au nombre total d'occurrences des termes de T. Le rapport de ces deux valeurs favorise la contribution des termes les plus fréquents dans la requête. NbElts(t) correspond au nombre d'éléments de la collection qui contiennent le terme t et reflète le pouvoir discriminant d'un terme. La division par NbElts(t) favorise les contributions des termes discriminants, c'est-à-dire apparaissant plus rarement dans la collection, suivant la même idée que l'idf.

$\varphi^{(\frac{NbT(T,E)}{NbT(T)})}$ Ce facteur mesure le niveau global de représentation de la requête dans l'élément XML. NbT(T,E) correspond au nombre de termes de la requête présents dans l'élément XML E. NbT(T) correspond au nombre de termes de la requête (cardinal de T). Le rapport de ces deux valeurs traduit la proportion de requête présente dans l'élément XML. Utiliser une puissance d'une constante permet d'accentuer l'écart entre les éléments où la requête est globalement très présente de ceux où elle est peu représentée. Le choix de la constante permet de faire varier l'influence de ce facteur dans la fonction de score.

Ces fonctions s'inspirent des fonctions de poids d'indexation de la littérature en recherche d'informations notamment les mesures de type *tf.idf* [22]. La définition de ces fonctions résulte de travaux et d'expérimentations réalisés dans le cadre de la catégorisation automatique de documents [1][2] et de la recherche d'information XML [11][23].

Le principe de la méthode est que chaque terme de la requête apparaissant dans un élément XML contribue au score de l'élément en fonction de sa présence et de son pouvoir discriminant. Un principe de propagation de score prend en compte la structure hiérarchique des documents XML.

La méthode permet également de prendre en compte la définition de préférences associées aux termes de la requête comme par exemple les notions de terme souhaité et terme non souhaité apparaissant dans le cadre INEX. De plus, cette méthode permet de moduler les scores calculés en fonction de contraintes de structure définies sur la requête. La méthode intègre le traitement des contraintes de structure associées au contenu recherché (contrainte de localisation) et des contraintes structurelles associées à la granularité de résultat recherché. Ces aspects n'étant pas utiles à l'étude menée dans cet article ils ne sont pas développés. Des détails peuvent être trouvés dans [11].

## 3.2. Propagation des scores

La propagation des scores permet de prendre en compte la structure hiérarchique des documents XML. L'hypothèse est qu'un élément XML contenant un composant pertinent est aussi pertinent et qu'il est d'autant plus pertinent qu'il contient plusieurs composants pertinents. Le score d'un composant est répercuté sur les éléments qu'il compose éventuellement à un degré moindre. Ce principe favorise la sélection de composants pertinents. Néanmoins, lorsqu'un élément est composé de

plusieurs composants pertinents, les scores des composants sont cumulés et répercutés sur l'élément composé. Ce principe privilégie donc les éléments composés de plusieurs éléments pertinents. Le score d'un composant sélectionné est propagé vers les éléments qu'il compose après application d'un facteur réducteur fonction de la distance entre le composant et l'élément composé (plus la distance est grande, plus le score propagé est réduit), comme suit :

$$ScoreAggrégé(T, E_A) = Score(T, E_A) + \sum_i (1 - \lambda \cdot \frac{d(E_A, E_i)}{d(E_R, E_i)}) \cdot Score(T, E_i)$$

$$Pour\ tout\ i\ tel\ que\ 1 - \lambda \cdot \frac{d(E_A, E_i)}{d(E_R, E_i)} > 0$$

où λ est un réel positif,

$E_A$, $E_R$ et $E_i$ sont des éléments XML tels que $E_A$ est ancêtre de $E_i$ dans le chemin associé et $E_R$ est la racine de l'arbre XML,

$d(E_X,E_i)$ est la distance entre $E_X$ et $E_i$ dans le chemin associé à $E_i$ correspondant au nombre d'ancêtres de $E_i$ jusqu'à $E_X$ inclus (ex. dans le chemin /article/bdy/s/ss1/p la distance entre p et bdy est 3),

$Score(T,E_A)$ est le score directement lié au contenu de $E_A$.

L'introduction de la constante λ permet de moduler la propagation des scores vers les nœuds ancêtres. Plus la valeur fixée pour λ est grande plus la propagation est limitée.

## 3.3. Couverture

L'objectif de la couverture est d'assurer que seuls sont sélectionnés les éléments dans lesquels la requête est suffisamment représentée. La couverture est un seuil relatif au pourcentage de termes de la requête qui apparaissent dans le texte d'un élément. Par exemple, une couverture de 50% implique qu'au moins la moitié des termes décrivant la requête doit apparaître dans un élément pour le sélectionner. La couverture est appliquée de la façon suivante sur la fonction de score :

Si $\frac{NbT(T,E)}{NbT(T)} < CT$ Alors $Score(T,E) = 0$

Sinon $Score(T,E)$ calculé suivant la définition de la fonction de score (cf 3.1)

où CT est le seuil de couverture.

Ce seuillage peut-être appliqué soit uniquement sur les scores directs des éléments possédant un contenu textuel soit sur les scores agrégés.

# 4. CRITERES DE DISTINCTION DES REQUETES

Au regard des expérimentations menées dans le cadre de campagnes d'évaluations comme INEX [14] pour la recherche d'information XML ou TREC [26] pour la recherche d'information, il est clair que les systèmes ne parviennent pas à traiter avec la même efficacité toutes les requêtes proposées.

Dans le cadre INEX, notre méthode n'échappe pas à la règle. Globalement notre méthode a donné en moyenne des résultats intéressants lors de la campagne INEX 2004, puisque classée au 13$^{ième}$ rang sur 70 soumissions des participants pour les requêtes définissant uniquement le contenu textuel souhaité et au 5$^{ième}$ rang sur 51 soumissions pour les requêtes qui combinent contenu et références explicites à la structure XML (cf. 4.1). Cependant, en regardant les résultats obtenus pour chaque requête des différences d'efficacité existent. C'est pourquoi nous avons cherché à identifier les caractéristiques des requêtes pour lesquelles notre méthode donne des résultats satisfaisants et les caractéristiques des requêtes pour lesquelles notre méthode donne de moins bons résultats. Le but est de pouvoir définir des types de requête et ainsi trouver si une configuration de la méthode peut être adaptée à chaque type de requête. De plus, la définition de types de requête peut permettre d'identifier les requêtes pour lesquelles notre méthode peut être appliquée et les requêtes pour lesquelles une autre approche peut être envisagée. La plupart des critères définis sont néanmoins transposables à d'autres contextes de RI.

## 4.1. Requêtes INEX

Le cadre INEX introduit deux types de requêtes, les requêtes de type CO (Content Only) qui décrivent uniquement le contenu souhaité des éléments XML recherchés et les requêtes de type CAS (Content And Structure) qui combinent contenu et références explicites à la structure XML. Les deux types de requêtes CO et CAS sont constituées de quatre parties : titre, description, narration et mots-clés. Le titre est considéré comme la requête de l'utilisateur ; les autres parties sont principalement destinées aux évaluateurs (jugements de pertinence). Une requête est constituée de mots simples et/ou de groupes de mots recherchés. Ces

éléments peuvent être préfixés pour indiquer des concepts à favoriser (préfixe +) ou des concepts non souhaités (préfixe -). Les requêtes de type CAS incluent des indications de structure sous formes de Xpath [4] précisant la localisation souhaitée pour les concepts recherchés et la granularité des éléments XML souhaités en résultat.

Exemples de requêtes :

- type CO : "query expansion" +"relevance feedback" +web
- type CAS : //article[about(.//p,object database)]//p[about(.,versioning)]

## 4.2. Critères

En analysant la définition d'une requête il est possible de recenser différents éléments constituants :

- les termes,
- les groupes de mots,
- les préférences (ou préfixes),
- les contraintes structurelles.

Ces constituants conduisent directement à la définition de critères possibles pour caractériser les requêtes à savoir :

- le nombre de termes constituant une requête,
- la présence ou non de groupes de mots ou bien le nombre de groupes,
- le nombre de mots simples,
- la présence ou non de préférences sur les mots voire le nombre de préférences positives et le nombre de préférences négatives,
- la présence ou non de contraintes de structure et plus précisément la présence de contraintes de granularité de résultat, la présence de contraintes sur les mots recherchés (contraintes de localisation) voire leur nombre. L'expression de contraintes de structure est complexe et peut engendrer une multitude de cas de figure. La définition de critères liés aux contraintes de structure réclame une étude approfondie qui dépasse le cadre de cet article.

Cette liste est bien entendu extensible en prenant en compte par exemple des notions linguistiques vis-à-vis des termes constituant les requêtes (par exemple nombre de noms, verbes ou sigles). Il est aussi possible d'introduire un critère relatif à la présence ou au nombre de mots composés.

Au-delà de ces critères immédiats définis précédemment il est possible de définir des critères supplémentaires comme par exemple :

- le nombre d'unités constituant une requête c'est-à-dire le nombre de mots simples plus le nombre de groupes de mots. Un utilisateur peut considérer que trouver une partie d'un groupe constitue déjà une certaine pertinence mais il peut également considérer un groupe de mot comme une unité à trouver intégralement,
- le nombre d'éléments de la collection contenant chaque terme de la requête et notamment le minimum des nombres d'éléments de la collection contenant un terme de la requête et le maximum des nombres d'éléments contenant un terme de la requête. Ce critère renseigne sur l'utilisation de termes plus ou moins discriminants dans la définition d'une requête.

Bien sûr, à partir de ces critères de base il est possible de définir des critères plus élaborés combinant plusieurs critères de base comme par exemple :

- la proportion de groupes de mots par rapport au nombre de termes ou par rapport au nombre d'unités si l'on suppose que l'utilisateur cherche à retrouver les unités qu'il a indiquées dans la définition de sa requête,
- l'écart entre le minimum des nombres d'éléments de la collection contenant un terme de la requête et le maximum des nombres d'éléments contenant un terme de la requête,
- la proportion de termes discriminants dans une requête,
- la proportion de termes ayant une préférence négative par rapport au nombre de termes.

Certains de ces critères ont été utilisés dans le cadre de travaux d'intégration de techniques de traitement automatique de langues (TAL) en recherche d'information dont [17] présente un panorama. Le nombre de termes peut par exemple distinguer les requêtes pour lesquelles certaines techniques de TAL peuvent améliorer les performances de systèmes de recherche d'information.

Dans cet article nous nous intéressons dans un premier temps au nombre de termes constituant une requête, au nombre de groupes de mots constituant une requête et à la proportion de groupes de mots par rapport au nombre d'unités. Les expérimentations accordent une grande place aux critères relatifs aux groupes de mots. La raison est notamment que notre méthode ne traitant pas de manière particulière les groupes de mots, ceux-ci peuvent donc constituer une limite.

# 5. EXPERIMENTATIONS

Les différentes expérimentations réalisées utilisent les jeux de tests fournis par l'environnement INEX.

## 5.1. Cadre INEX

La collection de documents utilisée correspond à celle fournie lors des campagnes d'évaluations INEX jusqu'en 2004 qui regroupe approximativement 12000 articles publiés par la 'IEEE Computer Society' de 1995 et 2002. La collection rassemble plus de 8 millions d'éléments XML de longueurs et de granularités variables (par exemple, titre, paragraphe ou article).

Les requêtes de type CO correspondant à la campagne d'évaluation INEX 2004 ont été utilisées pour étudier le comportement de différentes configurations de notre méthode au regard des critères choisis pour caractériser les requêtes (cf. 4.2). Pour les expérimentations présentées dans cet article nous avons utilisé le jeu de jugements de pertinence N°1 parmi les deux disponibles pour la campagne INEX 2004. Ces jeux diffèrent sur certaines requêtes jugées par des évaluateurs différents. Ceci n'a pas montré d'impact majeur sur les résultats obtenus pour les expérimentations officielles de la campagne INEX 2004 [14].

Les évaluations réalisées dans le cadre d'INEX 2004 sont basées sur les notions de rappel et de précision en prenant en compte également le degré de pertinence des composants retrouvés. Elles s'appuient sur le calcul de la probabilité qu'un document vu par un utilisateur soit pertinent. Différentes mesures sont calculables au travers de fonctions de quantifications qui décrivent différentes préférences d'utilisateurs. Pour chaque fonction de quantification, le taux de précision est calculé pour différentes valeurs du taux de rappel. La précision moyenne (MAP) est également calculée. Une mesure 'agrégée' correspondant à la moyenne des différentes mesures obtenues pour les fonctions de quantifications donne une mesure d'évaluation globale. Pour cette mesure 'agrégée', il est également possible de calculer la précision pour différentes valeurs de rappel ainsi que la précision moyenne. Cette précision moyenne (MAP) pour la mesure 'agrégée' (aggr) est celle sur laquelle s'appuient nos expérimentations. Les méthodes d'évaluation 'officielles' de la campagne INEX 2004 sont détaillées dans [25]. Les mesures d'évaluations ont été modifiées à partir de l'édition 2005 d'INEX [12]. Le changement de mesure a été motivé par le fait que les mesures utilisées jusqu'à l'édition

2004 ne tenaient pas compte du chevauchement des éléments dans les résultats lié à la structure hiérarchique des documents XML. Ainsi, les systèmes introduisant un chevauchement important pourraient être favorisés. Cette hypothèse n'a cependant pas pu être démontrée [28]. De plus, ce changement semble ne pas avoir eu d'impact notable sur les évaluations et le classement des systèmes lors d'INEX 2005. Par conséquent, compte tenu du manque de recul vis-à-vis des nouvelles mesures et du manque de stabilité des programmes les mettant en œuvre, nous avons préférés fonder notre étude sur le précédent environnement d'évaluation correspondant à INEX 2004.

## 5.2. Paramétrage des expérimentations

Différentes expérimentations utilisant notre méthode ont été réalisées sur le jeu de requêtes de type CO utilisé durant la campagne d'évaluation INEX 2004. Ces expérimentations possèdent une configuration commune d'un ensemble de paramètres (méthode d'indexation, coefficients associés aux préférences, constante $\varphi$) de la méthode résultant d'une phase d'entraînement réalisée sur les requêtes de type CO et CAS d'INEX 2003. La valeur de la constante $\varphi$ a notamment été fixée à 400 (cf. 3.1). L'efficacité de la méthode peut dépendre de l'indexation et ce paramètre est donc à étudier. Néanmoins, dans un premier temps, nous nous sommes orientés sur l'étude de l'influence de la couverture et de la propagation de score. Les configurations définies pour les expérimentations diffèrent donc sur les valeurs du seuil de couverture CT (cf. 3.3) et du coefficient de propagation $\lambda$ (cf. 3.2). Le libellé identifiant chaque expérimentation est construit comme suit de manière à identifier les valeurs choisies pour la couverture et la propagation :

CH0xCy où x correspond à une valeur de 0,x pour le coefficient de propagation de score $\lambda$ ($\lambda$=x/100) et y correspond à une valeur de y% de couverture (CT=y/100). Par exemple, l'expérimentation libellée CH025C35 est paramétrée avec une valeur de 0,25 pour le coefficient propagation de score $\lambda$ et avec une valeur du seuil de couverture CT égale à 0,35.

Il est à noter que plus le coefficient de propagation est grand plus faible est la propagation de score. De plus, plus le coefficient de couverture est élevé plus la couverture de la requête doit être grande pour qu'un élément soit retenu.

## 5.3. Résultats

Les expérimentations montrent dans un premier temps le comportement de notre méthode par rapport à des critères relatifs au nombre de termes constituant une requête, au nombre de groupes de mots définis et à la proportion de groupes de mots par rapport au nombre d'unités constituant la requête. Dans un deuxième temps, deux autres approches sont étudiées suivant ces mêmes critères.

### 5.3.1. Comportement de notre approche

| MAP, mesure : aggr | Nombre de termes de la requête | | | | |
|---|---|---|---|---|---|
| | 2-3 | 4-5 | 6-7 | 8-9 | 10-12 |
| MAP de CH005C0 | 0,0998 | 0,1056 | 0,0825 | 0,0450 | 0,0484 |
| MAP de CH005C35 | 0,1031 | 0,1087 | 0,0785 | 0,0598 | 0,0604 |
| MAP de CH005C50 | 0,1031 | 0,1039 | 0,0769 | 0,0341 | 0,0277 |
| MAP de CH025C35 | 0,1073 | 0,1092 | 0,0785 | 0,0697 | 0,0591 |
| MAP de CH065C35 | 0,1063 | 0,0927 | 0,0759 | 0,0777 | 0,0532 |
| Nombre de requêtes | 6 | 17 | 8 | 1 | 2 |

**TAB. 1 –** *Précision moyenne en fonction du nombre de termes*

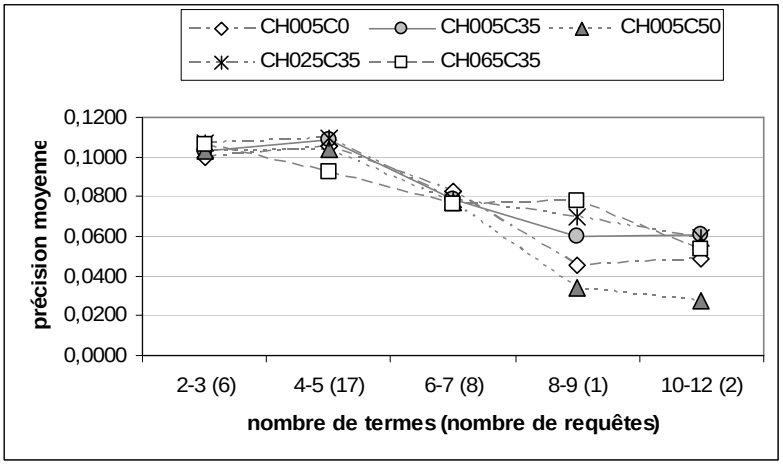


**FIG. 1 –** *Précision moyenne en fonction du nombre de termes*

Au regard des résultats synthétisés dans le Tableau 1 et représentés Figure 1, le premier constat est que le moteur de recherche semble plus efficace pour des requêtes constituées d'un nombre limité de termes (<=5) pour les configurations testées. Une explication peut-être que plus le nombre de termes augmente plus une partie de la requête présente dans un élément peut suffire à le juger pertinent. Notre méthode semble alors limitée pour gérer partiellement des requêtes. Les résultats montrent également que la configuration peut influencer les résultats obtenus. L'absence de couverture (CH005C0) ou trop de couverture (CH005C50) semble dégrader les résultats lorsque le nombre de termes est plus important. Une couverture trop faible entraîne le classement d'éléments où la requête est faiblement représentée mais où certains termes sont fortement présents. A l'inverse une couverture trop élevée supprime trop d'éléments pourtant apparemment jugés pertinents où la requête est trop partiellement présente. Cependant, compte tenu du nombre limité de requêtes de ce type cette constatation est à confirmer.

| MAP, mesure : aggr | Nombre de groupes de mots dans la requête | | | | |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 |
| MAP de CH005C0 | 0,0878 | 0,1337 | 0,0628 | 0,0763 | 0,0484 |
| MAP de CH005C35 | 0,0890 | 0,1392 | 0,0627 | 0,0684 | 0,0604 |
| MAP de CH005C50 | 0,0810 | 0,1437 | 0,0574 | 0,0669 | 0,0277 |
| MAP de CH025C35 | 0,0909 | 0,1404 | 0,0631 | 0,0683 | 0,0591 |
| MAP de CH065C35 | 0,0798 | 0,1288 | 0,0639 | 0,0641 | 0,0532 |
| Nombre de requêtes | 17 | 9 | 4 | 2 | 2 |

**TAB. 2 –** *Précision moyenne en fonction du nombre de groupes de mots*

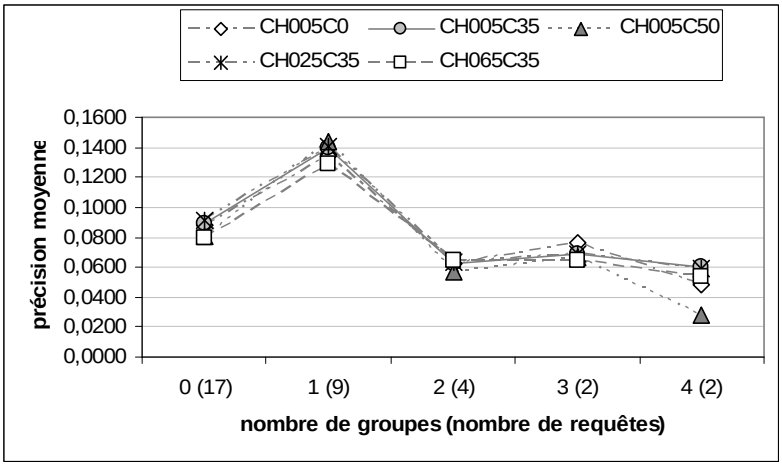


**FIG. 2 –** *Précision moyenne en fonction du nombre de groupes de mots*

Les résultats synthétisés dans le Tableau 2 et la Figure 2 montrent que la méthode de recherche semble plus efficace pour des requêtes constituées d'un nombre limité de groupes de mots (<=1). Les résultats montrent également qu'une combinaison appropriée de propagation de score et de couverture est préférable (l'expérimentation CH005C50 qui utilise une forte couverture donne majoritairement des résultats inférieurs ainsi que l'expérimentation CH065C35 qui propage peu les scores et utilise une couverture de 35%). Les résultats concernant des nombres de groupes de mots importants sont à considérer avec prudence compte tenu de l'échantillon limité. Les résultats plus faibles lorsque le nombre de groupes de mots augmente peuvent s'expliquer par le fait que les groupes de mots ne sont pas traités de manière spécifique mais comme des mots simples indépendants.

| MAP, mesure : aggr | Pourcentage de groupes de mots | | | | |
|---|---|---|---|---|---|
| | 0-0,2 | 0,2-0,4 | 0,4-0,6 | 0,6-0,8 | 0,8-1,0 |
| MAP de CH005C0 | 0,0895 | 0,1400 | 0,1031 | 0,0573 | 0,0699 |
| MAP de CH005C35 | 0,0915 | 0,1434 | 0,1092 | 0,0563 | 0,0703 |
| MAP de CH005C50 | 0,0838 | 0,1520 | 0,0964 | 0,0499 | 0,0703 |
| MAP de CH025C35 | 0,0932 | 0,1433 | 0,1107 | 0,0569 | 0,0702 |
| MAP de CH065C35 | 0,0820 | 0,1324 | 0,1002 | 0,0589 | 0,0658 |
| Nombre de requêtes | 18 | 5 | 5 | 4 | 2 |

**TAB. 3 –** *Précision moyenne en fonction de la proportion de groupes de mots*

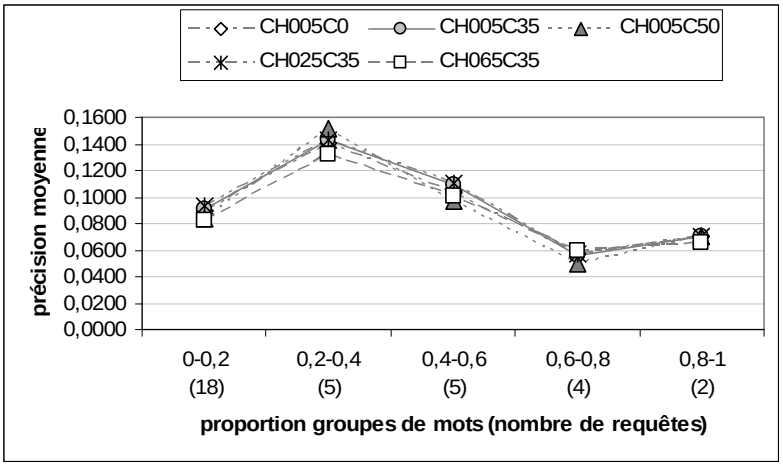


**FIG. 3 –** *Précision moyenne en fonction de la proportion de groupes de mots*

Le Tableau 3 et la Figure 3 montrent que la méthode de recherche semble plus efficace pour des requêtes pour lesquelles les groupes de mots représentent 20% à 60% des concepts recherchés. Au-delà, les résultats sont dégradés ce qui peut s'expliquer par le fait que notre méthode traite les groupes mots en tant que mots simples. Lorsqu'une requête est quasiment constituée de groupes de mots il semble que la notion de groupe de mots soit davantage à respecter par rapport aux requêtes constituées d'une part plus importante de mots simples.

### 5.3.2. Comportement d'autres approches

Au-delà d'identifier des critères permettant de distinguer des différences de comportements de notre méthode suivant les requêtes, il nous a semblé intéressant de regarder si ces critères mettaient en évidence des comportements différents pour d'autres approches. Pour cela, nous avons choisi d'étudier les expérimentations ayant obtenu les trois meilleurs rangs en moyenne toutes requêtes et mesures confondues globalement lors de la même campagne INEX 2004.

Le libellé identifiant chaque expérimentation indique le nom du participant ayant soumis l'expérimentation lors de la campagne INEX 2004. Ces expérimentations présentent également plusieurs caractéristiques intéressantes :

- deux expérimentations (nommées IBM Haifa1 et IBM Haifa2) correspondent à des variantes d'une même approche. Ceci permet de regarder si, comme observé pour notre méthode, les variantes d'une même approche présentent un comportement similaire au regard des critères étudiés,
- deux approches différentes sont représentées basées sur des modèles différents. De plus ces approches sont également différentes de la notre. L'approche correspondant aux expérimentations IBM Haifa1 et IBM Haifa2 (classées respectivement 1ère et 2ième) est fondée sur le modèle vectoriel, l'utilisation de plusieurs index à différents niveaux et la mesure de similarité Cosinus [15]. L'approche correspondant à l'expérimentation U. Waterloo (classée 3ième) [5] est fondée sur un modèle probabiliste et la mesure Okapi BM25 [21]. Ces différences permettent d'observer si des approches différentes présentent des différences de comportement vis-à-vis des critères étudiés. Ces différentes approches rejoignent néanmoins la notre dans le fait qu'elles ne traitent pas de manière particulière pour les groupes de mots.

Les données directement issues des systèmes choisis (IBM Haifa et U. Waterloo) et soumises lors de la campagne INEX 2004 ont été utilisées pour réaliser les analyses présentées dans cet article.

| MAP, mesure : aggr | Nombre de termes de la requête | | | | |
|---|---|---|---|---|---|
| | 2-3 | 4-5 | 6-7 | 8-9 | 10-12 |
| MAP de IBM Haifa1 | 0,1499 | 0,1375 | 0,1200 | 0,0871 | 0,0421 |
| MAP de IBM Haifa2 | 0,1478 | 0,1262 | 0,1137 | 0,0621 | 0,0384 |
| MAP de U. Waterloo | 0,0808 | 0,1398 | 0,0896 | 0,0834 | 0,0418 |
| MAP de CH025C35 | 0,1073 | 0,1092 | 0,0785 | 0,0697 | 0,0591 |
| Nombre de requêtes | 6 | 17 | 8 | 1 | 2 |

**TAB. 4 –** *Précision moyenne en fonction du nombre de termes*

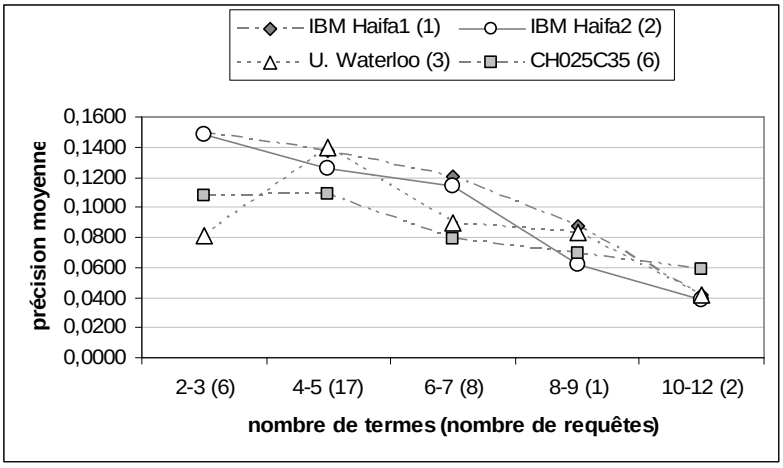


**FIG. 4 –** *Précision moyenne en fonction du nombre de termes*

Au regard des résultats synthétisés dans le Tableau 4 et représentés Figure 4, nous constatons que :

- toutes les approches suivent la même tendance face au nombre de termes de la requête : plus le nombre de termes augmente, plus l'efficacité de la méthode se dégrade. L'expérimentation 'U. Waterloo' suit cette tendance seulement à partir d'au moins 4 termes. En-dessous l'efficacité de cette approche semble également dégradée,
- les requêtes de moins de 4 termes sont celles pour lesquelles les différences d'efficacité entre approches sont les plus importantes. Notre approche 'CH025C35' qui se classerait globalement au 6$^{ième}$ rang atteint le 3$^{ième}$ rang pour ce type de requête,
- les variantes de la même approche 'IBM Haifa1' et 'IBM Haifa1' ont des comportements très similaires au regard du nombre de termes constituant les requêtes. Ceci rejoint le constat précédemment fait pour des variantes de notre approche (cf. 5.3.1).

| MAP, mesure : aggr | Nombre de groupes de mots dans la requête | | | | |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 |
| MAP de IBM Haifa1 | 0,1510 | 0,0988 | 0,1570 | 0,0997 | 0,0421 |
| MAP de IBM Haifa2 | 0,1424 | 0,0961 | 0,1256 | 0,1078 | 0,0384 |
| MAP de U. Waterloo | 0,1297 | 0,0988 | 0,0988 | 0,0865 | 0,0418 |
| MAP de CH025C35 | 0,0909 | 0,1404 | 0,0631 | 0,0683 | 0,0591 |
| Nombre de requêtes | 17 | 9 | 4 | 2 | 2 |

**TAB. 5 –** *Précision moyenne en fonction du nombre de groupes de mots*

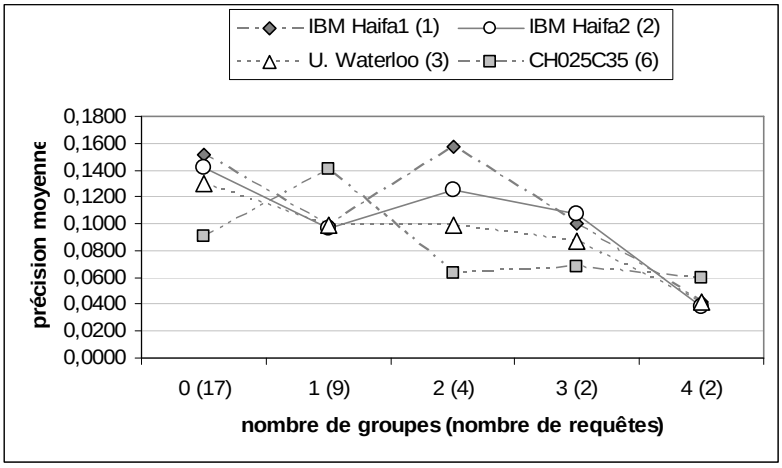


**FIG. 5 –** *Précision moyenne en fonction du nombre de groupes de mots*

Les résultats présentés dans le Tableau 5 et la Figure 5, montrent des différences de comportement entre les trois approches. Les approches 'IBM Haifa' et 'U. Waterloo' ont des comportements similaires exceptés au niveau des requêtes comportant 2 ou 3 groupes de mots. Notre approche présente un manque d'efficacité pour les requêtes constituées uniquement de mots simples ou constituées d'au moins 2 groupes de mots. Néanmoins, l'augmentation du nombre de groupes de mots ne semble pas affecter notre méthode contrairement aux autres approches. De plus, notre approche se distingue par une efficacité nettement plus importante pour les requêtes constituées de seulement 1 groupe de mots.

| MAP, mesure : aggr | Pourcentage de groupes de mots | | | | |
|---|---|---|---|---|---|
| | 0-0,2 | 0,2-0,4 | 0,4-0,6 | 0,6-0,8 | 0,8-1,0 |
| MAP de IBM Haifa1 | 0,1489 | 0,0772 | 0,1003 | 0,0980 | 0,2039 |
| MAP de IBM Haifa2 | 0,1397 | 0,0740 | 0,0996 | 0,0964 | 0,1563 |
| MAP de U. Waterloo | 0,1250 | 0,1245 | 0,0702 | 0,0923 | 0,0761 |
| MAP de CH025C35 | 0,0932 | 0,1433 | 0,1107 | 0,0569 | 0,0702 |
| Nombre de requêtes | 18 | 5 | 5 | 4 | 2 |

**TAB. 6 –** *Précision moyenne en fonction de la proportion de groupes de mots*

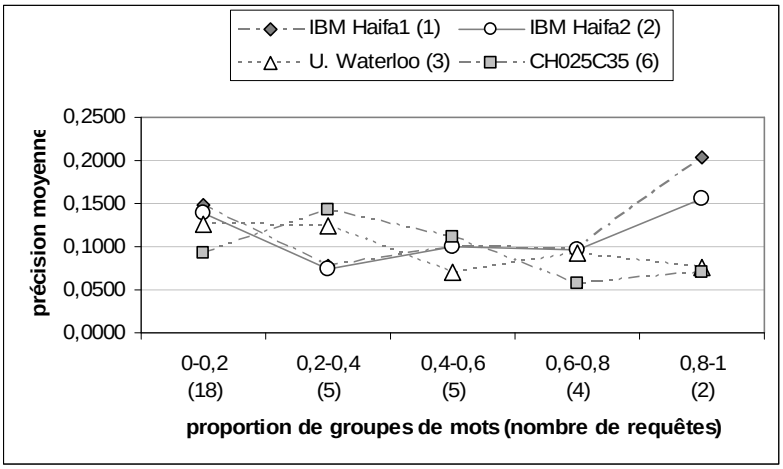


**FIG. 6 –** *Précision moyenne en fonction de la proportion de groupes de mots*

Le Tableau 6 et la Figure 6 montrent que les trois expérimentations les mieux classées globalement lors d'INEX 2004 ont principalement une efficacité supérieure pour les requêtes ayant une proportion de groupes de mots faible voire nulle c'est-à-dire inférieure à 20%. De même, les deux variantes du même système classées aux deux premiers rangs montrent des résultats très nettement supérieurs pour des requêtes constituées quasiment entièrement de groupes de mots c'est-à-dire une proportion de groupes de mots supérieure ou égale à 80%. A l'inverse, notre approche montre une meilleure efficacité pour une proportion de groupe de mots entre 20% et 60%.

## 5.4. Discussion

En analysant globalement les différents résultats obtenus et détaillés dans les différentes sections précédentes nous pouvons déduire les constatations suivantes :

- les critères les plus significatifs pour différencier des comportements de notre moteur de recherche sont le nombre de termes constituant la requête et le nombre de groupes de mots. Ces critères mettent en évidence que la méthode dans certaines configurations produit les meilleurs résultats pour des requêtes ayant moins de 6 termes et n'ayant pas plus d'un groupe de mots. Le critère de proportion de groupes de mots est moins significatif puisqu'il montre une faiblesse de la méthode à partir d'une proportion de groupes de mots de 60% ce qui correspond à seulement à 6 requêtes sur 34,
- les critères ne permettent pas réellement de dégager plusieurs configurations de la méthode qui soient efficaces pour des types de requêtes différents et donc qui pourraient être combinées pour améliorer les résultats. La meilleure configuration correspond à un coefficient de propagation de 0,25 et une couverture de 35%. Il est donc nécessaire de poursuivre cette étude suivant d'autres critères, avec d'autres configurations impliquant d'autres paramètres,
- les critères permettent d'identifier des comportements similaires pour différentes approches. Par exemple, les requêtes longues (requêtes constituées d'un nombre important de termes) conduisent à des résultats plus faibles (en termes de précision moyenne). De même, la présence de nombreux groupes de mots dégrade les résultats,
- les critères permettent d'identifier des plages de valeurs pour lesquelles différentes approches fournissent des résultats variables. Par exemple, le nombre de termes montre une efficacité supérieure pour une approche ('IBM Haifa') pour les requêtes de peu de termes (moins de 4). Ces critères permettent également de pointer des points forts ou des faiblesses des approches. Par exemple, la proportion de groupes de mots montre que l'efficacité notre méthode est supérieure pour une proportion entre 20% et 60% alors que d'autres approches sont supérieures lorsque la proportion est inférieure à 20% ou supérieure à 60%,
- les critères permettent d'identifier les approches ayant des comportements différents vis-à-vis des critères et notamment des comportements complémentaires. Par exemple, notre approche semble davantage complémentaire à l'approche 'IBM Haifa' que l'approche 'U. Waterloo'.

## 6. CONCLUSION

Nous avons présenté dans cet article une étude pour permettre de distinguer différents types de requêtes définies en recherche d'information XML. Le but était dans un premier temps d'améliorer notre propre méthode de recherche en identifiant les points faibles de notre approche. Il s'agissait de déterminer des types de requête pour lesquels notre méthode donne des résultats satisfaisants et des types de requêtes pour lesquels notre approche semble actuellement limitée. Le but était également de pouvoir déterminer quelle configuration de la méthode est la plus efficace pour chaque type de requête. Des études complémentaires impliquant d'autres configurations de notre méthode doivent être menées pour déterminer des configurations adaptées à chaque type de requêtes. Enfin, des analyses impliquant les résultats obtenus par d'autres approches ont permis de montrer des différences d'efficacité pour différents types de requêtes. Ces observations permettent d'envisager la possibilité de combiner différentes approches pour améliorer la recherche d'information XML. Des études complémentaires permettront d'affiner les critères et d'évaluer l'apport d'une combinaison de variantes de notre approche voire d'une combinaison d'approches différentes.

## Remerciements

Les recherches présentées dans cet article s'inscrivent dans le cadre du projet WS-Talk «WS-Talk: Web services communicating in the language of their user community», 6ième PRCD de l'Union Européenne (2002-2006), COOP-CT-2004 006026. Les idées exprimées dans ce papier sont cependant personnelles.

## 7. BIBLIOGRAPHIE

[1] J. Augé, K. Englmeier, G. Hubert et J. Mothe. Classification automatique de textes basée sur des hiérarchies de concepts. *Veille Stratégique Scientifique & Technologique (VSST'2001)*, Barcelone, p. 291-300, 2001.

[2] J. Augé, K. Englmeier, G. Hubert et J. Mothe. Catégorisation automatiques de textes basée sur des hiérarchies de concepts. *19èmes Journées Bases de Données Avancées (BDA'03)*, Lyon, p. 69-87, 2003.

[3] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler et Y. Yergeau. Extensible, Markup Language (XML) 1.0. (Third Edition). W3C Recommendation, http://www.w3.org/TR/REC-xml/, 2004.

[4] J. Clark et S. DeRose. XML Path Language (XPath). W3C Recommendation, http://www.w3.org/TR/xpath, 1999.

[5] C. L. A. Clarke et P. L. Tilker. MultiText Experiments for INEX 2004. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 85-87, 2005.

[6] C. J. Crouch, S. Apte et H. Bapat. An Approach to Structured Retrieval Based on the Extended Vector Model. *2nd INEX Workshop*, p. 89-93, http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf, 2003.

[7] T. Dkaki, G. Hubert, J. Mothe et E. Orain. Recherche de la nouveauté dans les textes : une tâche difficile. *Veille Stratégique Scientifique & Technologique (VSST'2004)*, Tome 2, p. 355-368, Toulouse, 2004.

[8] N. Fuhr et K. Großjohann. XIRQL: An XML query language based on information retrieval concepts. *ACM TOIS*, vol. 22, Issue 2, p. 313-356, 2004.

[9] T. Grabs et H.-J. Schek. Generating Vector Spaces On-the-fly for Flexible XML Retrieval. *ACM SIGIR Workshop on XML and Information Retrieval*, p. 4-13, 2002.

[10] D. Harman. Overview of the TREC 2002 Novelty Track. http://trec.nist.gov/pubs/trec11/t11_proceedings.html, 2002.

[11] G. Hubert. A voting method for XML retrieval. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 183-195, 2005.

[12] G. Kazai et M. Lalmas. INEX 2005 Evaluation Metrics. *INEX 2005 Workshop Pre-Proceedings*, http://inex.is.informatik.uni-duisburg.de/2005/pdf/inex-2005-preproceedings.pdf, p. 401-406, 2005.

[13] K. L. Kwok. An attempt to identify weakest and strongest queries. *ACM SIGIR Workshop on Predicting Query Difficulty*, 2005.

[14] S. Malik, M. Lalmas et N. Fuhr. Overview of INEX 2004. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 1–15, 2005.

[15] Y. Mass et M. Mandelbrod. Component Ranking and Automatic Query Refinement for XML Retrieval. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 73–84, 2005.

[16] V. Mihajlović, G. Ramírez, A. P. de Vries, D. Hiemstra et H. E. Blok. TIJAH at INEX 2004: Modeling Phrases and Relevance Feedback. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 276–291, 2005.

[17] F. Moreau et P. Sébillot. Contributions des techniques du traitement automatique des langues à la recherché d'information. Publication interne n°1690, IRISA, ftp://ftp.irisa.fr/techreports/2005/PI-1690.pdf, 2005.

[18] P. Ogilvie et J. Callan. Hierarchical Language Models for XML Component Retrieval. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX,* p. 224–237, 2005.

[19] J. Pehcevski, J. A. Thom et S. M. M. Tahaghoghi. Hybrid XML Retrieval Revisited. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX,* p. 153-167, 2005.

[20] B. Piwowarski, H.-T. Vu et P. Gallinari. Bayesian Networks and INEX'03. *2nd INEX Workshop*, Dagstuhl, Germany, p. 33-37, http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf, 2003.

[21] S. Robertson, S. Walker et M. Beaulieu. Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive. *7th Text Retrieval Conference (TREC-7)*, p. 253-264, 1998.

[22] G. Salton et C. Buckley. Term-weighting approaches in automatic text retrieval, Information Processing & Management, Volume 24, Issue 5, p. 513-523, 1988.

[23] K. Sauvagnat, G. Hubert, M. Boughanem et J. Mothe. IRIT at INEX 2003. *2nd INEX Workshop*, Dagstuhl, Germany, p. 142-148, http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf, 2003.

[24] B. Sigurbjörnsson, J. Kamps et M. de Rijke. Mixture Models, Overlap, and Structural Hints in XML Element Retrieval. *Advances in XML Information Retrieval, LNCS 3493, 3rd International Workshop INEX*, p. 196-210, 2005.

[25] A. P. de Vries, G. Kazai et M. Lalmas. Evaluation Metrics 2004. *Pre Proceedings of the 3rd INEX Workshop*, http://inex.is.informatik.uni-duisburg.de:2004/pdf/INEX2004PreProceedings.pdf, p. 249-250, 2004.

[26] E. M. Voorhees. Overview of TREC 2004. http://trec.nist.gov/pubs/trec13/ papers/, NIST, 2004.

[27] E. M. Voorhees. Overview of the TREC 2004 Robust Track. http://trec.nist.gov/pubs/trec13/papers/, NIST, 2004.

[28] A. Woodley et S. Geva. XCG Overlap at INEX 2004. *INEX 2005 Workshop Pre-Proceedings*, http://inex.is.informatik.uni-duisburg.de/2005/pdf/ inex-2005-preproceedings.pdf, p. 25-39, 2005.

[29] E. Yom-Tov, S. Fine, D. Carmel, A. Darlow et E. Amitay. Improving document retrieval according to prediction of query difficulty. *Working Notes of Text Retrieval Conference (TREC 2004)*, p. 393-402, 2004.