# Recherche de la pertinence et de la nouveauté dans les textes

**Taoufiq Dkaki (*, **), Josiane Mothe (*,***)**

*(*) Institut de Recherche en Informatique de Toulouse, 118 route de Narbonne, 31062 Toulouse Cedex 04, France*
*(**) ISYCOM-GRIMM, (***) Institut Universitaire de Formation des Maîtres*
*{dkaki,mothe}@irit.fr tél: 05 61 55 63 22*

RÉSUMÉ. *Les systèmes de recherche d'information s'intéressent à retrouver les documents pertinents par rapport à un besoin défini par un utilisateur. Certains systèmes se sont intéressés à mieux répondre au besoin de l'utilisateur en considérant un niveau de granularité plus petit que le document. Dans ces systèmes, les informations restituées à l'utilisateur ne correspondent pas aux documents mais aux passages susceptibles de correspondre au besoin exprimé. Cet article répond à la double tâche proposée dans le programme TREC : rechercher les passages pertinents et ceux qui apportent de la nouveauté. Nous présentons une nouvelle stratégie de sélection de passages qui permet d'être efficace sans utiliser de ressources extérieures. Nous discutons les résultats obtenus sur les collections de TREC utilisées en 2002 et 2003.*

ABSTRACT: *Information retrieval systems usually retrieve documents. Studies have been carried out in order to better answer users'needs and present systems that retrieve smaller chunks of texts. This paper presents an approach that aims at answering the TREC novelty track: retrieve relevant passages and detect what are the novel sentences among the relevant ones. We discuss the results we obtained on two collections provided by TREC.*

MOTS-CLÉS : *recherche de passages, détection de la nouveauté, représentation de requête, marqueur de pertinence et de non pertinence*

KEYWORDS: *passage retrieval, novelty, query relevance feedback*

## 1. Introduction

Traditionnellement, les travaux dans le domaine de la recherche d'information se basent sur la recherche de documents c'est à dire sur des unités d'indexation et de restitution correspondant aux documents entiers. Le choix d'une telle granularité est lié au fait que les premiers systèmes de recherche d'information manipulaient des documents courts. Il s'agissait de documents secondaires correspondant à des références aux documents que l'utilisateur pouvait consulter dans leur format papier d'origine, une fois sélectionnés et localisés. Actuellement, une grande partie des systèmes manipule les documents primaires au format électronique. Cette évolution a entraîné un changement de la taille des documents manipulés, qui est passé de quelques lignes à plusieurs pages voire plusieurs dizaines de pages. L'utilisateur peut alors ne pas être satisfait par une réponse d'un système qui lui indique qu'un document est pertinent mais qui ne l'aide pas à retrouver à l'intérieur du contenu, les éléments spécifiques répondant réellement à son besoin. Plusieurs types de réponses à ce problème existent. La simple mise en évidence (en surbrillance par exemple) des termes du document qui ont conduit à sa restitution peut guider l'utilisateur dans sa lecture ; on peut penser dans ce cadre aux travaux liés au principe de « Key Word in Context » (Albus, 1971), qui ont initier d'autres travaux (Mladenic, 1998).

La restitution de passages plutôt que des documents entiers correspond à une autre réponse à l'aide à la consultation d'information. Cette approche qui a vu le jour au début des années 90 avec l'essor du langage SGML (Standard Generalized Mark-up Language) permettait d'effectuer des recherches sur des parties de documents sémantiquement cohérentes issues d'un marquage structurel du document (Wilkinson, 1994), (Corral, 1995). Dans le cas de documents non structurés, d'autres travaux se sont intéressés à la recherche de passages ou de fenêtres de texte (Salton, 1994). Plus récemment, le programme d'évaluation TREC, TExt Retrieval Conference s'est intéré à un niveau de granularité plus fin puisqu'il s'agit du niveau de la phrase (Harman, 2002).

Une seconde évolution de la gestion électronique des documents concerne le volume des données manipulées. Par exemple Google indexe plus de 3,3 milliards de pages web différentes (http://www.searchenginewatch.com, septembre 2003). Cette évolution indique que les systèmes tendent à restituer des documents en grande quantité quelque soit la requête. Les utilisateurs reçoivent de moins en moins souvent une réponse du type 'aucun document ne correspond à votre requête'. Cela est d'autant plus vrai lorsque la recherche est effectuée par des non-spécialistes de la recherche documentaire ; les spécialistes ayant en particulier pour tâche de restreindre le nombre de réponses restituées par le système. Une conséquence est que les documents restitués peuvent contenir des informations redondantes amenant un sentiment de non-satisfaction de l'utilisateur. Dans le cadre de la tâche intitulée 'Novelty' apparue en 2002, TREC s'intéresse à ce problème en proposant d'évaluer des systèmes capables de détecter la nouveauté dans les textes pertinents. L'idée

retenue est de détecter, parmi les phrases pertinentes, lesquelles apportent des éléments nouveaux et de ne restituer que celles-ci.

Dans cet article, nous répondons donc à la double tâche proposée dans le programme TREC :

- la recherche des phrases pertinentes à partir de documents jugés pertinents,
- la recherche des phrases qui apportent de la nouveauté.

Nous étudions différentes stratégies et discutons les résultats obtenus sur les collections de TREC 2002 et 2003. Ces résultats sont comparés aux résultats obtenus par les autres groupes participants à TREC dans le cadre de cette tâche.

## 2. Travaux dans le domaine

### *2.1. Recherche des passages pertinents*

Un système de recherche d'information est basé sur trois processus principaux : l'indexation, la recherche des informations répondant à la requête de l'utilisateur et leur restitution. L'indexation s'attache à représenter chaque document par un ensemble de termes pondérés. Lors de la phase de recherche, un calcul de ressemblance entre ces représentations de documents et la requête permet de définir les documents à restituer. Cette restitution s'effectue en général par ordre décroissant de ressemblance entre la requête et les documents.

La restitution de passages plutôt que de documents entiers est apparue avec l'essor du langage SGML qui permettait de définir la notion de passage en s'appuyant sur la structure explicite du document. Dans (Wilkinson, 1994), (Corral, 1995) les unités documentaires (parties de documents) sont extraites en s'appuyant sur la DTD (Document Type Definition). Ces unités correspondent alors aux unités manipulées par le système à chaque étape : indexation, recherche et restitution. D'autres travaux se sont intéressés à la combinaison de ressemblances dites locales, c'est à dire calculées au niveau des passages, et de ressemblances globales, c'est à dire calculées au niveau des documents. (Salton, 1994) propose une recherche en deux étapes : les documents entiers sont indexés ; lors d'une recherche, les documents supposés pertinents sont alors indexés par passage et un nouveau calcul de ressemblance avec la requête permet de sélectionner les meilleurs passages. Ces travaux ont montré que le découpage optimal était le paragraphe. D'autres types de passages ont été définis dans le cadre de documents non structurés comme les fenêtres de taille fixe (Stanfill, 1992) ou les phrases (Harman, 2002).

TREC 2002 (Harman, 2002) a défini la tâche de recherche de passages au niveau de la phrase. Cette recherche de phrases pertinentes est réalisée à partir de documents jugés pertinents par des évaluateurs. Le problème se trouve certes

simplifié (il n'y a plus le risque de sélectionner une phrase issue d'un document non pertinent) mais ce choix permet de se focaliser sur un aspect précis de la recherche de passages.

La majorité des systèmes utilisés dans TREC ont considéré les phrases comme des documents et appliquent simplement les techniques existantes au niveau des phrases plutôt qu'au niveau des documents. Ainsi les phrases, considérées de façon individuelle, ont d'abord été indexées avant de calculer la ressemblance de ces phrases avec la requête. Bien que les requêtes TREC comprennent une partie description qui donne des informations sur les documents qui seront considérés comme pertinents et sur ceux qui seront considérés comme non pertinents, les participants à TREC considèrent cette description de façon globale et perdent donc des informations importantes. (Allan, 2003) a utilisé le modèle vectoriel. (Collins, 2002) a également considéré une mesure cosinus avec une pondération de type tf.idf et a complété la recherche en appliquant les techniques de réinjection de pertinence. Ils ont également étudié différents types de classificateur en se basant sur des caractéristiques lexicales et sémantiques issues de l'analyse des textes. (Schiffman, 2002) a choisi d'étendre la requête en ajoutant à la requête initiale des termes sémantiquement équivalents et des termes fortement co-occurrents avec les termes de la requête. (Zhang, 2002, 2003) combine la réinjection de pertinence aveugle avec une classification des phrases utilisant un algorithme SVM (Support Vector Machine). (Dkaki, 2002) a amené une nouvelle approche en caractérisant les termes d'indexation selon trois classes : très pertinents, pertinents, non pertinents. Dans cet article, cette dernière approche sert de point de départ. Nous la complétons en ajoutant une quatrième classe de termes : les termes fortement non pertinents qui sont détectés par une analyse plus fine des requêtes. De nouveaux paramètres ont également été introduits (pondération en particulier) et évalués. Les résultats sont fournis dans cet article.

### *2.2. Recherche de la nouveauté*

Eviter de fournir à l'utilisateur des informations redondantes (i.e. détecter la nouveauté dans les documents et ne fournir que les éléments nouveaux) correspond à une approche complémentaire dans l'aide proposée à l'utilisateur. Il s'agit, pour un thème donné, de savoir si deux phrases sont redondantes ou complémentaires par rapport à un sujet donné.

(Allan, 2003) a étudié différentes mesures. Les meilleurs résultats ont été obtenus en utilisant la mesure LMDiri qui se base sur la représentation des textes par le modèle de langage (Ponte, 1998) en lissant les représentations en fonction de la longueur des phrases. (Kazawa, 2002) sélectionne les phrases nouvelles parmi les phrases pertinentes en se basant sur la mesure de la pertinence marginale maximum (MMR) (Carbonell, 1998). Dans notre approche, la détection de la nouveauté est basée sur une fonction de décision calculée en combinant la similarité de la phrase

considérée avec chacune des phrases déjà traitées et avec une phrase abstraite correspondant à l'union des phrases déjà traitées.

## 3. Définition de la tâche et des collections de test

### *3.1. Recherche des phrases pertinentes et nouvelles*

Dans cet article nous nous intéressons à la recherche de passages potentiellement intéressants pour l'utilisateur en nous basant sur la tâche « nouveauté » telle que définie dans TREC (Harman, 2002). Cette tâche comprend deux types de sous-tâches :

- La recherche des phrases pertinentes à partir de documents connus comme étant pertinents,

- La recherche des phrases apportant des éléments d'information nouveaux par rapport au besoin (sous ensemble des phrases pertinentes).

### *3.2. Collections d'évaluation*

Les collections que nous utilisons pour l'évaluation de nos méthodes sont celles de TREC.

#### *3.2.1. Caractéristiques des collections*

En 2002, TREC a choisi de sélectionner 49 requêtes issues des requêtes 300-450 des collections TREC. Le NIST (National Institute of Standards and Technology) a sélectionné les documents effectivement pertinents pour chacune des requêtes, avec un maximum de 25 documents et les a fournis aux participants. Dans une seconde étape, des évaluateurs humains ont indiqué quelles phrases étaient effectivement pertinentes et quelles apportaient des éléments nouveaux. Les caractéristiques de cette collection sont fournies dans le tableau 1. En 2003, le même type de collection a été constitué. Les caractéristiques sont rapportées dans le même tableau.

| | NIST-2002 | NIST-2003 |
|---|---|---|
| Nombre de requêtes | 49 | 50 |
| Nbre moyen de documents pertinents par requête | 22,3 | 25 |
| Nbre moyen de phrases issues des documents par requêtes | 1321 | 796,4 |
| Nombre moyen de phrases pertinentes par requête | 27,9 | 311,14 |
| % moyen de phrases pertinentes | 2,1 | 39 |
| Nombre moyen de phrases nouvelles par requête | 25,3 | 204,5 |
| % moyen de phrases nouvelles | 90,9 | 65,7 |

**Tableau 1 :** Caractéristiques des collections de test de TREC

*3.2.2. Requête*

Une requête a la forme suivante :

```
Requête: 2
Titre: clone Dolly sheep                Type: event
Description: Cloning of the sheep Dolly
Narration: To be relevant information there must be specific
reference to 'Dolly' or 'the first cloned sheep' or 'large
animal.' References to Dolly's children are relevant if
Dolly's name is included.  Mention of the company that cloned
Dolly is not relevant if nothing more is said about Dolly.
References to the consequences of her being a clone are
relevant. Mention of Polly and Molly are not relevant.
```

**Figure 1 :** Exemple de requête

La partie ‘Titre’ simule une requête telle que pourrait la formuler un utilisateur. Il s’agit généralement d’un ensemble de mots-clés. La partie ‘Description’ précise la recherche. Elle est généralement formulée en langage naturel. La partie ‘Type’ n’a pas été utilisée dans nos expérimentations ; elle peut contenir la valeur ‘événement’ ou ‘opinion’, selon le type de requête. Enfin, la partie ‘Narration’ indique quelles informations seront considérées comme pertinentes ; elle simule donc le point de vue de l’utilisateur. Cette section peut comprendre des éléments positifs et des éléments négatifs (cf. figure 1). Cette formulation des requêtes TREC est donc plus complète qu’une requête telle qu’elle peut être exprimée sur les moteurs du Web par exemple. Cet aspect peut donc être critiqué. Cependant, nous faisons l’hypothèse que des mécanismes peuvent être mis en place pour récolter cette information de façon implicite ou explicite auprès de l’utilisateur.

Concernant l’analyse des requêtes, un des aspects originaux de notre approche est que nous avons plus particulièrement tiré profit de cette distinction entre

éléments considérés comme pertinents et éléments considérés comme non pertinents.

### *3.3. Critères d'évaluation*

#### *3.3.1. Mesures dans le cadre général*

Les critères d'évaluation que nous avons utilisés sont ceux définis par TREC et sont directement issus des critères communément utilisés pour évaluer les systèmes de recherche d'information : les taux de rappel et de précision. Ces deux taux évoluant en sens inverse, une mesure globale, la mesure F combinant rappel et précision permet une comparaison rapide des résultats obtenus par différents systèmes.

#### *3.3.2. Mesure concernant les phrases pertinentes*

Le calcul des taux est ramené au niveau de la phrase comme suit :

$$Rs = \frac{Nombre\ de\ \ phrases\ pertinentes\ et\ retrouvées}{Nombre\ de\ \ phrases\ pertinentes}$$

$$Ps = \frac{Nombre\ de\ \ phrases\ retrouvées\ et\ \ pertinentes}{Nombre\ de\ \ phrases\ retrouvées}$$

La mesure F est définie par : $$F1s = \frac{2 * Ps * Rs}{Ps + Rs}$$

#### *3.3.3. Mesure concernant les phrases nouvelles :*

Le même type de ratio est défini concernant l'évaluation de la nouveauté (« *phrase nouvelle* » remplace « *phrase pertinente* » dans les ratios précédents).

Comme l'évaluation prend en compte un ensemble de requêtes, les résultats que nous donnons dans les tableaux des sections 4 et 5 correspondent à la moyenne des résultats obtenus pour chacune des requêtes.

## 4. Détection des phrases pertinentes

### *4.1. Description de la méthode*

Comme dans le cadre des travaux soumis à TREC 2002 (Dkaki, 2002), notre approche consiste à considérer la phrase comme l'unité documentaire. La recherche de phrases pertinentes est donc basée sur les trois étapes suivantes :

#### *4.1.1. Indexation des phrases issues des documents*

Chaque phrase issue d'un document est considérée comme une unité documentaire et est indexée selon une approche classique :

- élimination des mots vides : les termes considérés comme non discriminants sont éliminés (articles, prépositions, etc.).

- radicalisation : les différentes variantes d'un terme sont ramenées à une forme unique ; nous utilisons pour cela une liste de mots issus du dictionnaire anglais à chacun desquels est associé sa racine. Cette liste est également utilisée dans LexiQuest (http://www.lexiquest.fr/). Elle contient 21291 entrées.

- pondération : un poids est associé à chaque terme ti pour une phrase Sj donnée. Ce poids est calculé comme suit : $Poids(t_i, S_j) = tf_{i,j}$ (1)

où $tf_{i,j}$ est la fréquence du terme $t_i$ dans la phrase $S_j$ .

#### *4.1.2. Indexation des phrases issues des requêtes*

Un paramètre important de notre méthode concerne les composantes de la requête utilisées. En effet, une requête est composée de trois parties distinctes : le titre, la description et la narration (cf. section 3.2.2). Nous combinons ces parties de différentes façons. En particulier, la partie narration est décomposée selon que les éléments expriment la pertinence ou la non pertinence des phrases retrouvées. Cette décomposition est basée sur la détection de marqueurs linguistiques. Dans l'exemple de la figure 1, la partie narration de la requête est décomposée en deux parties comme indiqué figure 2 :

```
NarrationPert: To be relevant information there must be
specific reference to 'Dolly' or 'the first cloned sheep' or
'large animal.' References to Dolly's children are relevant if
Dolly's name is included. References to the consequences of
her being a clone are relevant.
NarrationNonPert : Mention of the company that cloned Dolly is
not relevant if nothing more is said about Dolly. Mention of
Polly and Molly are not relevant.
```

**Figure 2 :** Partie narration de la requête Figure 1, décomposée en deux parties.

Une fois sélectionnées (la ou) les parties à considérer pour construire la représentation de la requête, les phrases qui les composent sont analysées et indexées selon les mêmes étapes : suppression des mots vides, radicalisation et pondération. Nous introduisons toutefois une représentation originale de la requête qui nous permet ensuite de pondérer différemment les termes selon la partie de la requête d'où ils proviennent. Ainsi une requête $Q_k$ est représentée par un ensemble de termes comme suit :

Soient $Q_k$ une requête et $t_i$ un terme,

$Q_k = QT_k \cup QD_k \cup QNP_k \cup QNN1_k \cup QNN2_k$ où $QT_k$ correspond à l'ensemble des termes extraits du *Titre* de la requête, $QD_k$ de la partie *Description*, $QNP_k$ de la partie *NarrationPert* et $QNN1_k \cup QNN2_k$ correspond aux termes extraits de la partie *NarrationNonPert*. Ils sont définis par $QNN2_k \cap (QT_k \cup QD_k \cup QNP_k \cup QNN1_k) = \phi$. Ainsi, $QNN1_k$ correspond aux termes qui apparaissent aussi dans une autre partie (Titre, Description ou NarrativePert ) alors que les termes de $QNN2_k$ apparaissent seulement dans la partie NarrationNonPert de la requête. Ces derniers sont spécifiques de la non-pertinencee.

Dans la suite, $tf_{i,k,P}$ est la fréquence d'apparition du terme $t_i$ dans la partie P de la requête $Q_k$, $P \in \{T, D, NP, NN1, NN2\}$ .

Le poids associé à un terme d'indexation pour la requête est alors calculé comme indiqué dans (2).

$$
\begin{aligned}
&soit \quad \omega_{i,k} = \sum_{P \in \{T,D,NP,NN1,NN2\}} \mu_P \cdot tf_{i,k,P} \\
&Poids(t_i, Q_k) = \omega_{i,k} \quad si \quad \omega_{i,k} \geq \tau_H \quad ou \quad \omega_{i,k} < 0 \\
&\qquad\qquad = \tau_L \quad si \quad 0 < \omega_{i,k} < \tau_H \\
&\qquad\qquad = 0 \qquad dans\ les\ autres\ cas \\
&où \quad \mu_P \ est\ un\ coefficient\ dépendant\ de\ la\ partie\ P\ considérée
\end{aligned}
\qquad (2)
$$

Cette formule de poids repose sur l'hypothèse qu'un terme de la requête doit posséder un poids proportionnel à sa fréquence dans chacune des parties de la requête ($w_{i,k}$) ; mais que chaque partie de la requête peut être considérée de façon différente. $\mu p$ permet donc de pondérer l'importance de chaque partie de la requête. Intuitivement, le titre aura plus d'importance que la partie descriptive, c'est-à-dire que $\mu_T \geq \mu_D$. Selon notre approche, un terme qui est trop peu fréquent dans la requête peut avoir un poids ramené à une constante, typiquement de valeur inférieure à sa fréquence réelle ($\tau_L \leq \omega_{i,k}$). Intuitivement, $\mu_{NN2}$ devrait être négatif pour diminuer l'importance des termes associés à la non pertinence. Enfin, cette fonction de pondération peut permettre d'annuler certaines parties de la requête (ne considérer que le titre et la description par exemple en ramenant les autres coefficients $\mu_P$ à une valeur nulle). Concrètement, les hypothèses que nous avons faites ont été validées par expérimentation.

Chaque terme est alors catégorisé dans un des groupes définis comme suit :

$$HQ_k = \{t_i / t_i \in Q_k \text{ et } Poids(t_i, Q_k) > \tau_L\}$$
$$LQ_k = \{t_i / t_i \in Q_k \text{ et } Poids(t_i, Q_k) = \tau_L \text{ et } Poids(t_i, Q_k) > 0\}$$
$$iQ_k = \{t_i / Poids(t_i, QP_k) = 0 \quad \forall P \in \{T, D, NR, NN1, NN2\}\}$$
$$IQ_k = \{t_i / Poids(t_i, Q_k) < 0\}$$

$HQ_k$ and $LQ_k$ correspondent aux termes caractérisant la pertinence des phrases. $HQ_k$ sont les termes très pertinents (ils ont un poids dans la requête important) et $LQ_k$ sont les termes moins pertinents (ils ont un poids dans la requête plus faible, mais cependant non nulle). $IQ_k$ correspond aux termes caractérisant la non-pertinence des phrases (présents dans la partie caractéristique de la non-pertinence des éléments qui seraient retrouvés). $iQ_k$ correspond typiquement à un terme d'indexation non présent dans la requête.

*4.1.3. Calcul de ressemblance et détection de la pertinence*

La ressemblance entre une phrase et la requête est calculée comme suit :

Soient $Q_k$ une requête, $S_j$ une phrase

$$Score(S_j, Q_k) = \sum \left( Poids(t_i, S_j) \cdot Poids(t_i, Q_k) \right) \qquad (3)$$

Où Poids ($t_i$, $S_j$) est le poids du terme $t_i$ dans la phrase $S_j$ en cours de traitement (issue d'un document) et Poids ($t_i$, $Q_k$) est le poids du terme $t_i$ dans la requête $Q_k$. Ces poids sont calculés par (1) et (2). Il s'agit là donc d'un calcul classique dans le modèle vectoriel. La taille des éléments considérés (phrases) justifie le fait que ce calcul ne soit pas normalisé.

Une phrase donnée $S_j$ est alors considérée comme pertinente ssi :

$$Score(S_j, Q_k) > \vartheta \quad (4)$$

$\vartheta$ peut être une fonction dépendant du nombre de termes très pertinents des requêtes (éléments de $HQ_k$) qui apparaissent dans la phrase $S_j$ et du nombre de termes peu pertinents des requêtes (éléments de HQk) qui apparaissent dans la phrase $S_j$ (Dkaki, 2004). $\vartheta$ peut être une constante (cf. section 4.2.1.).

*4.1.4. Extraction de groupes de mots*

L'extraction de groupes de mots (ou expressions) plutôt que des mots simples a pour but d'obtenir une représentation plus précise des contenus (Caropreso, 2001). Il semblerait donc utile de représenter les textes non pas par des termes simples mais par des groupes de mots directement issus de l'analyse des textes. Notre approche de détection des groupes de mots s'appuie sur une analyse statistique des segments répétés. Les groupes de mots sont constitués par des chaînes de termes apparaissant fréquemment en séquence. Cette extraction est effectuée avant la suppression des

mots vides qui peuvent être de bons marqueurs de groupes de mots. D'autre part, différentes variantes de mots sont prises en compte (forme singulier/pluriel, féminin/masculin). Par exemple les chaînes «des bases de données » et « une base de données » sont considérées comme équivalente. Ainsi, les textes (phrases, requêtes) sont représentés à la fois par des uni-termes et des groupes de mots. Il faut noter qu'un terme simple et un groupe de mots contenant ce mot peuvent co-exister dans les représentations obtenues. Dans le cas des groupes de mots, les mêmes fonctions de calcul de poids des termes sont appliquées, ainsi les formules (1) et (2) restent identiques ; simplement $t_i$ peut être soit un terme simple, soit un groupe de mots. Le calcul des phrases pertinentes reste identique.

*4.1.5. Réinjection de pertinence*

Le principe de réinjection de pertinence a été largement utilisé en recherche d'information (Rocchio, 1971), (Harman, 1992). Il consiste à reformuler de façon automatique la requête de l'utilisateur à partir d'éléments fournis par l'utilisateur: le système fourni à l'utilisateur les résultats de la recherche obtenus par le système à partir de la requête initiale. L'utilisateur doit alors indiquer la pertinence de ces éléments. Cette connaissance est utilisée pour ajouter à la requête les termes issus des documents pertinents, éventuellement repondérer des termes ou éliminer des termes de la requête (ceux responsables de la sélection de documents non pertinents). La réinjection de pertinence aveugle consiste à considérer que le système a retrouvé en début de liste des documents pertinents ; les $\delta$ premiers documents sont donc considérés comme pertinents pour reformuler la requête (Mitra , 1998). Nous avons appliqué ce principe au niveau des phrases.

La réinjection de pertinence se traduit dans notre modèle par la modification dans la formule (2) lorsque $tf_{i,k,P} > \varsigma$ où RP correspond au texte composé de l'ensemble des $\eta$ premières phrases retrouvées à partir de la requête initiale:

$$\omega_{i,k} = \sum_{P \in \{T, D, NP, NN1, NN2\}} \mu_P \cdot tf_{i,k,P} + \mu_{RP} \quad (5)$$

Ainsi, le poids des termes de la requête est modifié pour y ajouter une contribution issue de l'analyse des documents supposés pertinents, selon le même principe que celui de réinjection de pertinence défini par (Rocchio, 1971). Le coefficient ($\mu_{RP}$) permet ici de pondérer l'importance de la prise en compte des termes issus des documents 'pertinents' (l'équivalent de $\beta$ dans Rocchio). Dans le cas de la réinjection de pertinence, les formules (3) et (4) permettant de décider quelles phrases sont pertinentes (cf section 4.1.3.) restent identiques.

***4.2. Résultats***

Différentes valeurs de paramètres ont été étudiées. Les meilleurs résultats ont été obtenus avec les paramètres suivants :

*4.2.1. Sans réinjection de pertinence*

$$Poids\ (t_i, S_j) = tf_{i,j} \qquad (1)$$

$$\begin{aligned} \omega_{i,k} &= 4*tf_{i,k,T} + tf_{i,k,D} + tf_{i,k,NP} + tf_{i,k,NN1} - tf_{i,k,NN2} \\ Poids(t_i, Q_k) &= \omega_{i,k} \quad si \quad \omega_{i,k} \geq 3 \quad ou \quad \omega_{i,k} < 0 \\ &= 1 \quad si \quad 0 < \omega_{i,k} < 3 \\ &= 0 \qquad dans\ les\ autres\ cas \end{aligned} \qquad (6)$$

$$Score(S_j, Q_k) \geq 3$$

Ainsi, le titre doit être pondéré de façon plus significative que les autres parties (ces résultats sont conformes à nos attentes). La valeur 3 correspond au type de valeur que l'on pouvait imaginer de façon intuitive dans la mesure où les éléments que nous traitons sont courts (la fréquence d'apparition des termes dans $S_j$ est généralement de 1.

| | *TREC 2002* | | | *TREC 2003* | | |
|---|---|---|---|---|---|---|
| | *Ps* | *Rs* | *F1s* | *Ps* | *Rs* | *F1s* |
| Meilleur TREC | 0,23 | 0,34 | 0,235 | 0,60 | 0,79 | 0,619 |
| Officiel TREC | 0,15 | 0,49 | 0,190 | 0,64 | 0,58 | 0,526 |
| Meilleur résultat sans réinjection de pertinence | 0,14 | 0,50 | 0,190 | 0,62 | 0,65 | 0,561 |

**Tableau 2:** Evaluation de la sélection des phrases pertinentes à partir des collections

Dans le tableau 2, la ligne « Meilleur TREC » correspond au meilleur résultat qui a été obtenu sur l'ensemble des données envoyées par les différents groupes participant à cette tâche de TREC en 2002 et 2003. Il s'agit des résultats dont le détail figure dans (Zhang, 2002) et (Zhang, 2003). Dans ce même tableau, la ligne 'Officiel TREC' correspond aux résultats obtenus sur les tests qui ont été envoyés pour notre participation au programme en 2002 et 2003. Ces résultats ont été obtenus avec une fonction (2) qui ne prenait pas en compte les termes issus de la partie que nous avons qualifiée de « Narration non pertinente ». D'autre part, certaines valeurs de paramètre étaient différentes. Ces résultats correspondaient aux rangs 4 sur 13 participants en 2002 et 5 sur 11 en 2003. Enfin, la dernière ligne de ce tableau reporte les résultats que nous avons obtenus en utilisant les paramètres optima pour la collection TREC 2003, c'est-à-dire ceux qui nous ont permis d'obtenir les meilleurs résultats finaux sur cette collection (cf. section 4.2.2). Il faut noter que les paramètres optima pour la collection 2002 sont sensiblement différents, mais que l'amélioration dans ces cas n'est pas significative puisque nous obtenons une mesure F de 0,195 (au lieu de 0,190). On peut donc dire que les paramètres optima sont relativement stables (sur les collections existantes actuellement).

*4.2.2. Résultats avec réinjection de pertinence*

Comme indiqué dans la section 4.1.5, la prise en compte des principes de réinjection de pertinence modifie simplement le calcul de $w_{i,k}$. Les meilleurs résultats ont été obtenus par le calcul suivant, avec :

$$\omega_{i,k} = 4*tf_{i,k,T} + tf_{i,k,D} + tf_{i,k,NP} + tf_{i,k,NN1} - tf_{i,k,NN2} + 1 \; si \; tf_{i,k,RP} \geq 4 \quad (7)$$

et i est un terme qui apparaît dans au moins une requête.

$$\omega_{i,k} = 4*tf_{i,k,T} + tf_{i,k,D} + tf_{i,k,NP} + tf_{i,k,NN1} - tf_{i,k,NN2} \quad \text{sinon}$$

où RP est composé des 10 premières phrases retrouvées à partir de la requête initiale. Les autres paramètres restent inchangés. Ces résultats sont cohérents avec les résultats de la littérature lorsque la recherche s'effectue sur des documents entiers et en se basant sur la réinjection aveugle (typiquement 10 à 12 documents correspond à un optimum). La valeur optimale 4 montre qu'il ne faut pas donner trop d'importance aux termes issus des documents par rapport aux termes issus de la requête initiale.

| | *TREC 2002* | | | *TREC 2003* | | |
|---|---|---|---|---|---|---|
| | *Ps* | *Rs* | *F1s* | *Ps* | *Rs* | *F1s* |
| Meilleur TREC | 0,23 | 0,34 | 0,235 | 0,60 | 0,79 | 0,619 |
| Sans réinjection | 0,14 | 0,50 | 0,190 | 0,62 | 0,65 | 0,561 |
| Avec Réinjection | 0,14 | 0,50 | 0,191 | 0,59 | 0,75 | 0,593 |

**Tableau 3:** Evaluation de la sélection des phrases pertinentes avec réinjection

Les deux premières lignes du tableau 3 reprennent les lignes 1 et 3 du tableau 2 à des fins de comparaison.

Concernant la collection TREC 2003, l'utilisation de la réinjection de pertinence aveugle améliore le rappel d'environ 15% ; la précision est dégradée, mais dans une moindre mesure. Globalement, la mesure F est améliorée. Il faut noter que ces résultats positionnent notre approche au 3ième rang (sur onze) par rapport aux participants de TREC en 2003. La différence entre le meilleur résultat (Zhang, 2003) et le résultat obtenu par notre méthode concernant la mesure F est seulement de 4%. Concernant la collection 2002, cette approche nous positionne en quatrième position sur treize. En revanche, dans le cas de cette collection, les résultats sont stables. Une analyse plus poussée est nécessaire pour interpréter ce résultat.

## 5. Détection des phrases nouvelles

Les phrases nouvelles sont celles qui contiennent de nouveaux éléments ou qui abordent un nouvel aspect de la requête non encore rencontré dans les phrases déjà retrouvées.

### *5.1. Méthode de détection*

Pour décider si une phrase Sj doit être considérée comme nouvelle, nous calculons la ressemblance entre cette nouvelle phrase et les autres phrases considérées comme nouvelles par le système. Nous calculons ensuite la ressemblance entre la nouvelle phrase et une phrase virtuelle correspondant à l'union des phrases déjà considérées comme nouvelles.

Soit $\Pi = \{S_1, S_2, ...S_n, \}$ l'ensemble des phrases déjà traitées et considérées comme nouvelles par le système et $S^{'} = \bigcup_{i \in \{1,...,n\}} S_i$, $S^{'}$ est une phrase virtuelle composée de toutes les phrases de Π,

Pour décider si une phrase s est nouvelle, nous calculons les similarités suivantes:

$Sim(s, S^{'}) = \alpha_s$ et pour $i \in \{1,...,n\}$ $Sim(s, s_i) = \omega_{s,i}$ où $s_i$ est une phrase que le système a déjà traitée et qu'il a considéré comme nouvelle. Nous utilisons le produit scalaire comme fonction $Sim$.

Nous considérons ensuite les q précédentes phrases les plus ressemblantes à s.

Pour $i \in \{1,...,n\}$ $P_{s,i}$ est la série de phrases obtenue en ordonnant Π par ordre décroissant de $\omega_{s,i}$ : $\beta_s = \sum_{i \in \{1,...q\}} Sim(s, P_{s,i})$

Le principe général de la décision sur la nouveauté ou le caractère redondant d'une phrase (formule 8) est que la phrase en cours de traitement doit être suffisamment différente de la phrase virtuelle constituée des phrases déjà choisies comme pertinentes et suffisamment différentes des phrases sélectionnées comme nouvelles. La seconde condition est un cas particulier de la première condition et permet de prendre en compte le fait que des phrases similaires dans un document ont plus de chance de parler du même sujet. Ainsi,

*s* est considérée comme redondante (non nouvelle) si et seulement si:

$$\alpha_s \geq \tau_1 \text{ and } \beta_s \geq \tau_2 \qquad (8)$$

Ces deux coefficients sont définis plus loin.

### *5.2. Evaluation*

Pour évaluer les méthodes de détection de la nouveauté, il est important que les phrases soient considérées dans un ordre fixe quelle que soit la méthode de détection. Dans l'évaluation ci-après, l'ordre des documents correspond à celui donné par le NIST et l'ordre des phrases est celui de leur occurrence dans les documents. Il est important de noter que, selon les évaluateurs du NIST, la majorité (plus de 90,9 %) des phrases pertinentes a été considérée comme nouvelle dans la collection de 2002 alors que seulement 65,7 % des phrases l'ont été en 2003. La tâche de détection de la nouveauté est plus facile en utilisant la collection 2003 qu'en utilisant la collection 2002.

Les résultats que nous présentons ont été obtenus avec les paramètres suivants :

- $\tau_1$ =1 et $\tau_2$ =0,6

| | *TREC 2002* | | | *TREC 2003* | | |
|---|---|---|---|---|---|---|
| | *Pn* | *Rn* | *F1n* | *Pn* | *Rn* | *F1n* |
| Meilleur TREC | 0,22 | 0,30 | 0,217 | 0,466 | 0,74 | 0,505 |
| Résultats officiels | 0,16 | 0,22 | 0,157 | 0,45 | 0,55 | 0,426 |
| Meilleurs résultats (non officiel) | 0,14 | 0,49 | 0,187 | 0,43 | 0,71 | 0,477 |

**Tableau 4 :** Evaluation de la sélection des phrases pertinentes

Ces résultats positionnent notre approche quatrième sur treize pour la collection 2002 et troisième sur onze pour la collection 2003.

## 6. Conclusion et perspectives

Dans cet article, nous avons présenté une nouvelle méthode de détection de passages pertinents. Le niveau de granularité auquel nous nous sommes intéressés correspond à la phrase. Cette problématique est récente et complète des études actuelles sur la recherche de composants XML (eXtended Mark-up Language), elle complète également des études précédentes qui s'intéressent à des niveaux de granularité différents. Nous avons également présenté une méthode de détection de la nouveauté. Cette problématique a été introduite dans les évaluations à grande échelle en 2002. Nous avons montré sur les deux collections de test existant dans le domaine, chacune composée d'environ 50 requêtes, que notre approche permettait de bien se positionner par rapport à la littérature du domaine.

Ces travaux nous amènent toutefois à deux types de questionnement :

- Le niveau de la phrase est-il pertinent pour détecter la pertinence et la nouveauté? Certains travaux plus anciens indiquaient que le niveau du paragraphe

était pertinent (Salton, 1994). Il est alors souhaitable de mettre à disposition de l'utilisateur une interface de consultation qui permet de re-situer les résultats dans leur contexte plus global (Corral, 1995). Une étude spécifique devrait être menée afin de déterminer la satisfaction de l'utilisateur non pas en terme de pertinence des contenus mais en terme de pertinence de granularité. Ces problématiques commencent à être abordées en particulier dans le cadre du projet INEX.

- Peut-on être satisfait des résultats obtenus ? Si de façon relative (i.e. comparé à la littérature du domaine) la réponse est incontestablement « oui », de façon absolue la réponse est cependant « non ». Par rapport à la collection 2002 (plus difficile que la collection 2003), seulement la moitié des phrases pertinentes est restituée alors même que ces phrases sont issues de documents pertinents. Ces résultats seraient donc probablement fortement dégradés sans cette connaissance à priori sur la pertinence des documents. Le comportement de la méthode sur la collection 2003 est meilleur puisque environ 80% des phrases pertinentes sont retrouvées.

Nos travaux futurs pour améliorer les résultats absolus s'orientent selon différents axes complémentaires. Le premier concerne l'extension de notre méthode à d'autres sous tâches liées à la détection de nouveauté. Il s'agit d'évaluer l'efficacité de la méthode en fonction du niveau de connaissance de départ : ensemble de documents retrouvés vs ensemble de documents pertinents, ensemble des phrases retrouvées par le système vs phrases jugées pertinentes, etc. Le deuxième concerne l'adaptation des techniques de réinjection (supervisée par l'utilisateur ou aveugle) pour prendre en compte la spécificité de la notion d'ordre des résultats.

## 7. Références


J. S. Albus, J. Yeh," Introduction to the Key Word In Context Index (KWIC) to the ACM IS & R symposium", p 225-284, 1971.

J. Allan, C. Wade, A. Bolivar, "Retrieval and Novelty Detection at the Sentence Level", *26ème Conf. ACM, Research and Development in Information Retrieval*, p 314-321, 2003.

J. Carbonnel, J. Goldstein, "The use of MMR, Diversity-Based Reranking for Reordering Document and Producing Summaries", *21ème Conférence ACM-SIGIR, Research and Development in Information Retrieval*, p 335-336, 1998.

M-F. Caropreso, S. Matwin, F. Sebastiani, A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization, dans Amita G. Chin (ed.), Text Databases and Document Management: Theory and Practice, Idea Group Publishing, Hershey, US, p 78-102, 2001.

K. Collins-Thompson, P. Ogilvie, Y. Zhang, J. Callan, I"nformation filtering, Novelty detection, and named-page finding", *actes de Text Retrieval Conf.*, p 107-118, 2002.

M.-L. Corral, J. Mothe, “How to retrieve and display long structured documents *?”*, *actes du congrès Basque International Workshop on Information Technology*, BIWIT’95, p 10-19, 1995.

T. Dkaki, J. Mothe, “Novelty track at IRIT-SIG”, *actes de Text Retrieval Conference*, p 332-336, 2002.

T. Dkaki, J. Mothe, “Combining Positive and Negative Query Feedback in Passage Retrieval”, *RIAO*, à paraître 2004.

V. Geroimenko, C. Chen, Visualizing the Semantic Web XML-based Internet and Information Visualisation, Springer, ISBN 1-85233-576-9, 2002.

D. Harman, “Relevance feedback revisit”, *actes de la 15ième Conference ACM-SIGIR, Research and Development in Information Retrieval*, p 1-10, 1992.

D. Harman, “Overview of the TREC 2002 novelty track”, *actes de Text Retrieval Conference*, p 46-55, 2002.

H. Kazawa, T. Hirao, H. Isozaki, E. Maeda, “A machine learning approach for QA and Novelty tracks: NTT system description”, *actes de Text Retrieval Conf.*, p 472-475, 2002.

M. Mitra, A. Singhal, C. Buckley `Improving automatic query expansion', *actes de la 21ième Conf. ACM-SIGIR Conference on Research and Development in Information Retrieval*, p 206-214, 1998.

D. Mladenic, “Feature Subset Selection in Text-Learning*”, European Conf. on Machine Learning*, p 95-100, 1998.

J.M. Ponte, W.B. Croft, “A language modelling approach to information retrieval*”, actes de la 21ième Conf. ACM-SIGIR, Research and Development in Information Retrieval*, p 275-281, 1998.

J. Rocchio. Relevance feedback information retrieval, G. Salton ed., The Smart retrieval system| experiments in automatic document processing, p 313-323. Prentice-Hall, Englewood Cli s, NJ, 1971.

G. Salton, J. Allan, C. Buckley, “Automatic structuring and retrieval of large text files”, *communication de l’ACM*, 37(2), p 97-108, 1994.

B. Schiffman, “Experiments in Novelty Detection at Columbia University*”, actes de Text Retrieval Conference*, p 188-196, 2002.

C. Stanfill, D.L. Waltz, Statistical methods, artificial intelligence, and information retrieval, Text-based intelligent systems: current research and practice in information extraction and retrieval, Ed. P.S. Jacobs, p 215-226, 1992.

R. Wilkinson, “Effective retrieval of structured documents*”, actes de la 17ième Conference ACM-SIGIR, Research and Development in Information Retrieval*, p 311-317, 1994.

M. Zhang, R. Song, C. Lin, S. Ma, Z. Jiang, Y. Jin, Y. Liu, L. Zhao, et S. Ma, “Expansion-based technologies in finding relevant and new information”: *THU TREC2002: Novelty Track Experiments*, *actes de Text Retrieval Conference*, p 586-590, 2002.

M. Zhang, C. Lin, Y. Liu, L. Zhao, L. Ma, S. Ma, “THUIR at TREC 2003: Novelty, Robust, Web and HARD”, *actes de Text Retrieval Conference*, p 137, 2003.