Une approche de recherche d'information structurée fondée sur la correction d'erreurs à l'indexation des documents

Arnaud Renard[1, 2] Sylvie Calabretto[1, 2] Béatrice Rumpler[1, 2]
(1) Université de Lyon, CNRS
(2) INSA-Lyon, LIRIS, UMR 5205, F-69621 Villeurbanne Cedex
arnaud.renard@insa-lyon.fr, sylvie.calabretto@insa-lyon.fr,
beatrice.rumpler@insa-lyon.fr

RÉSUMÉ

Dans cet article, nous nous sommes intéressés à la prise en compte des erreurs dans les contenus textuels des documents XML. Nous proposons une approche visant à diminuer l'impact de ces erreurs sur les systèmes de Recherche d'Information (RI). En effet, ces systèmes produisent des index associant chaque document aux termes qu'il contient. Les erreurs affectent donc la qualité des index ce qui conduit par exemple à considérer à tort des documents mal indexés comme non pertinents (resp. pertinents) vis-à-vis de certaines requêtes. Afin de faire face à ce problème, nous proposons d'inclure un mécanisme de correction d'erreurs lors de la phase d'indexation des documents. Nous avons implémenté cette approche au sein d'un prototype que nous avons évalué dans le cadre de la campagne d'évaluation INEX.

ABSTRACT

Structured Information Retrieval Approach based on Indexing Time Error Correction

In this paper, we focused on errors in the textual content of XML documents. We propose an approach to reduce the impact of these errors on Information Retrieval (IR) systems. Indeed, these systems rely on indexes associating each document to corresponding terms. Indexes quality is negatively affected by those misspellings. These errors makes it difficult to later retrieve documents (or parts of them) in an effective way during the querying phase. In order to deal with this problem we propose to include an error correction mechanism during the indexing phase of documents. We achieved an implementation of this spelling aware information retrieval system which is currently evaluated over INEX evaluation campaign documents collection.

MOTS-CLÉS : Recherche d'information, dysorthographie, correction d'erreurs, xml.

KEYWORDS: Information retrieval, misspellings, error correction, xml.

1 Introduction

Les documents produits dans un cadre professionnel doivent satisfaire à un niveau minimum de qualité et font l'objet de multiples cycles de relecture et correction permettant d'y parvenir. Cela constituait auparavant le principal mode de production d'informations néanmoins cette pratique a fortement évolué et à l'échelle d'Internet, il s'agit désormais d'un mode de production de l'information qui peut être considéré comme marginal. En effet, la plupart des documents sont créés par des utilisateurs hors de tout cadre professionnel. Ces derniers sont donc davantage

Actes de la conférence conjointe JEP-TALN-RECITAL 2012, volume 2: TALN, pages 519–526,
Grenoble, 4 au 8 juin 2012.

susceptibles de commettre des erreurs en employant un lexique qu'ils ne maîtrisent pas toujours et qui peut s'avérer inadapté au sujet traité. Par ailleurs, le contenu publié sur Internet n'est pas soumis à un contrôle de qualité : les blogs ont popularisé l'auto-publication de masse à la fois gratuite et immédiatement disponible. Il est donc légitime dans ce cas d'émettre des réserves sur la qualité des documents et autres informations produits dans ce cadre (Subramaniam *et al.*, 2009). Les systèmes de RI constituent les principaux points d'accès aux informations d'Internet. Ils sont affectés par les erreurs (Kantor et Voorhees, 2000) dont la correction constitue un axe d'amélioration important qu'il convient d'étudier (Varnhagen *et al.*, 2009).

Dans la section 2 nous présenterons la RI dans les documents (semi-)structurés XML ainsi que les travaux tentant de mêler RI et correction d'erreurs. Dans la section 3, nous présenterons notre approche intégrant la gestion de la correction des erreurs durant la phase d'indexation des documents. Nous analyserons les résultats de l'évaluation de notre système de RI sans et avec prise en charge des erreurs sur la campagne d'évaluation INEX dans la section 4. Enfin, nous conclurons et nous présenterons nos perspectives d'évolution en section 5.

2 Contexte général et positionnement

2.1 Recherche d'information structurée

Les documents XML constituent un des formats de diffusion de l'information les plus répandus sur internet. Nous allons dans un premier temps modéliser ces documents dont la structure explicite est plus complexe que de simples documents textuels "plats". Un document XML structuré ds peut être représenté par un arbre dans lequel on peut distinguer 3 types de nœuds différents : les nœuds feuilles nf_i représentant le contenu textuel, les nœuds internes ni_i correspondant aux éléments ainsi que leurs attributs na_i.

```
<?xml version="1.0" encoding="UTF−8"?>
<article>
  <name id="1337">Lorem...</name>
  <body>
    ...ipsum dolor sit amet,
    <emph>consectetur</emph>
    adipiscing elit.
  </body>
</article>
```

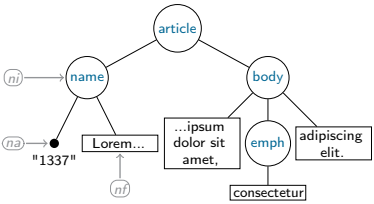

FIGURE 1 – Document XML (à gauche) et sa représentation arborescente (à droite).

Les informations textuelles sont présentes principalement dans les nœuds feuilles qui sont les nœuds à indexer en priorité et qui constitueront le niveau de granularité le plus fin de notre système de RI. Il diffère en cela des systèmes de RI classiques dont la granularité correspond au document. Plusieurs approches de la littérature (Kamps *et al.*, 2009) permettent la prise en compte de cette granularité plus fine mais aussi de la structure des documents. Nous proposons de nous appuyer sur une adaptation du modèle vectoriel de (Salton, 1971) ainsi que sur l'approche employée par XFIRM (Sauvagnat et Boughanem, 2005) qui introduit une méthode de propagation de la pertinence au travers de la structure des documents.

2.1.1 Pondération des nœuds feuilles (orientée contenu)

Lors de l'évaluation d'une requête le score relatif à la pertinence des nœuds feuilles est calculé directement, tandis que les scores des nœuds internes sont propagés dynamiquement à partir des nœuds feuilles à travers l'arborescence du document. Cela permet de retourner une liste ordonnée des nœuds (sous-arbres) les plus pertinents pour la requête.

Le score d'un nœud feuille s_{nf} vis-à-vis d'une requête textuelle r composée d'une séquence de n termes (ou mots-clés) $t_1, ..., t_n$ se calcule selon la formule suivante :

$$s_{nf}(r) = \sum_{i=1}^{n} p_{t_i}^{r} \times p_{t_i}^{nf} \tag{1}$$

dans laquelle, $p_{t_i}^{r}$ et $p_{t_i}^{nf}$ sont respectivement les poids du i-ème terme t_i dans la requête r (évalué lors de l'interrogation), et dans le nœud feuille nf (évalué lors de l'indexation). Afin d'adapter le modèle vectoriel de Salton aux documents XML structurés, nous avons choisi un système de pondération qui reflète l'importance locale des termes dans les nœuds feuilles (tf) [1] et globale dans les documents (idf) [2] ainsi que dans les éléments (ief) [3] de la collection.

$$p_{t_i}^{r} = tf_{t_i}^{r} \qquad p_{t_i}^{nf} = tf_{t_i}^{nf} \times idf_{t_i} \times ief_{t_i} \tag{2}$$

où, $tf_{t_i}^{r}$ et $tf_{t_i}^{nf}$ sont respectivement la fréquence du terme t_i dans la requête r et dans le nœud feuille nf. La fréquence correspond au nombre d'occurences du terme t_i respectivement dans la requête r (dénoté par $|t_i^r|$) et dans nf (dénoté par $|t_i^{nf}|$), divisé par le nombre de termes respectivement dans la requête r (dénoté par $|r|$) et dans le nœud feuille nf (dénoté par $|nf|$).

$$tf_{t_i}^{r} = \frac{|t_i^r|}{|r|} \qquad tf_{t_i}^{nf} = \frac{|t_i^{nf}|}{|nf|} \tag{3}$$

et, idf_{t_i} (resp. ief_{t_i}) représente la fréquence inverse du terme t_i dans les documents (resp. les nœuds feuilles). $|D|$ (resp. $|NF|$) est le nombre total de documents (resp. nœuds feuilles) de la collection et $|d_{t_i}|$ (resp. $|nf_{t_i}|$) le nombre de documents (resp. nœuds feuilles) qui contiennent le terme t_i.

$$idf_{t_i} = log\left(\frac{|D|}{|d_{t_i}|+1}\right) + 1 \qquad ief_{t_i} = log\left(\frac{|NF|}{|nf_{t_i}|+1}\right) + 1 \tag{4}$$

2.1.2 Pondération des nœuds internes (orientée structure)

Lorsqu'un nœud feuille est pertinent vis-à-vis d'une requête, les nœuds internes ancêtres de ce dernier le sont également dans une certaine mesure du fait qu'ils englobent ce dernier. Le score des nœuds feuilles peut ainsi être propagé de proche en proche à leurs nœuds ascendants (selon une fonction d'agrégation) jusqu'au nœud racine qui représente le document dans son intégralité.

$$s_{ni}(r) = |NF_{ni}^{s_{nf}(r)>0}|. \sum_{nf_k \in NF_n} \alpha^{dist(ni,nf_k)-1} \times s_{nf_k}(r) \tag{5}$$

1. tf : term frequency (fréquence du terme dans un contexte : requête, élément, ou document).
2. idf : inverse document frequency (fréquence inverse du terme dans les documents).
3. ief : inverse element frequency (fréquence inverse du terme dans les éléments).

où, α compris dans l'intervalle $[0..1]$ représente le facteur d'atténuation de l'importance du nœud feuille nf_k vis-à-vis du nœud interne ni et $dist(ni, nf_k)$ représente la distance entre le nœud interne ni et le nœud feuille nf_k dans la structure arborescente du document. Ainsi, les termes qui apparaissent près de la racine d'un sous-arbre sont plus pertinents pour l'élément racine que ceux qui apparaissent à un niveau plus profond du sous-arbre.

Et $|NF_{ni}^{s_{nf}(q)>0)}|$ représente le nombre de nœuds feuilles du nœud interne qui sont pertinents car un nœud qui contient plus de nœuds pertinents peut être considéré comme plus pertinent.

En présence d'erreurs, le calcul des scores des nœuds feuilles et notamment le facteur $p_{t_i}^{nf}$ de la formule 1 est impacté car le $tf_{t_i}^{nf}$ (cf. formule 2) est amoindri voire annulé dans certain cas ce qui diminue la pertinence du nœud. Il est donc important de considérer la correction des erreurs.

2.2 Correction des erreurs dans les systèmes de RI

La plupart des approches de correction d'erreurs associées aux systèmes de RI considèrent uniquement la correction des requêtes. tels que les travaux de (Sitbon *et al.*, 2007), ou encore le "Did you mean..." introduit par Google qui n'est donc pas adapté. Certains des travaux liés à la campagne d'évaluation TREC [4] considèrent la correction des documents.

La campagne TREC-5 Confusion track a rendu disponibles différentes versions d'une collection de plus de 55000 documents contenant respectivement des taux d'erreurs de 0%, 5%, et 20%. L'article de synthèse de la campagne (Kantor et Voorhees, 2000) présente les différentes approches pour la gestion des erreurs suivies par 5 des participants. Néanmoins, ils ont pu constater une dégradation des performances de tous les systèmes de RI en présence de documents corrompus contenant des erreurs essentiellement dues à la non correspondance entre les termes de la requête et les termes par lesquels les documents ont été indexés. Le même phénomène d'augmentation des silences à l'interrogation et de perte de précision même à de faibles taux de corruption des documents (taux d'erreurs de 3%) a été observé par (Ruch, 2002). La campagne TREC-6 Spoken document retrieval track (Voorhees *et al.*, 2000) considère des documents issus de transcriptions de même que (Gravier *et al.*, 2011).

Dans le cadre de TREC-5, trois systèmes s'appuient sur l'expansion de requêtes en y ajoutant des versions altérées des termes qui la composaient. Cela présente l'inconvénient d'introduire du bruit supplémentaire dans les résultats du système de RI lorsque le nombre de variations des termes de la requête ajoutées à la requête initiale augmente. Deux autres systèmes ont suivi des approches différentes et ont essayé de corriger directement les erreurs présentes dans les documents ce qui semble apporter un gain plus important. Cela constitue un point de départ intéressant dans l'étude de la robustesse des systèmes de RI face aux erreurs.

3 Proposition : Construction d'index corrigés

Notre approche consiste à corriger les erreurs lors de la phase d'indexation du système de RI et plus précisément pendant l'analyse du contenu textuel des documents. Notre proposition s'appuie

4. TREC : Text REtrieval Conference

donc sur deux sous-systèmes : un *système de RI XML* fondé sur le modèle XFIRM présenté en section 2.1, et un *système de correction d'erreurs* qui y est intégré.

En effet, le modèle XFIRM ne permet pas la prise en compte des erreurs, c'est pourquoi les fonctions de calcul de la pondération des termes doivent être modifiées pour en tenir compte. De plus, un système de correction d'erreurs est nécessaire afin d'identifier les termes erronés et d'identifier les termes qui doivent leur être substitués dans l'index.

Supposons les deux phrases suivantes *p1* et *p2* appartenant respectivement à deux documents XML *ds1* et *ds2* simples car comportant un seul élément à leur racine respectivement *nf1* et *nf2* :

p1 : "The trees are green." ***p2*** : "Green paper is made of teer."

Lors de la construction de l'index, les termes sont lemmatisés (mis sous une forme standard : noms au singulier, ...) puis filtrés en fonction d'une liste de mots non significatifs ("stop-words"). L'index construit à partir de ces deux documents est ainsi représenté dans la table 1.

Terme	Document	Élément	tf	idf	ief
green	ds1	nf1	0,5	0,82	0,82
	ds2	nf2	0,33		
paper	ds2	nf2	0,33	1	1
teer	ds2	nf2	0,33	1	1
tree	ds1	nf1	0,5	1	1

TABLE 1 – Index des documents *ds1* et *ds2* (les facteurs *idf* et *ief* sont égaux car les documents ne comportent chacun qu'un seul élément).

Ainsi, une recherche comportant les mots-clés ***tree*** et ***paper*** aboutira aux scores suivants pour chacun des nœuds des deux documents :

$$\begin{aligned} s_{ds1}(r) = s_{nf1}(r) &= p^r_{tree} \times p^{nf1}_{tree} + p^r_{paper} \times p^{nf1}_{paper} = 0,25 \\ s_{ds2}(r) = s_{nf2}(r) &= p^r_{tree} \times p^{nf2}_{tree} + p^r_{paper} \times p^{nf2}_{paper} = 0,165 \end{aligned} \tag{6}$$

Comme cela peut être constaté sur cet exemple, le document *ds1* obtient un score de $0,25$ supérieur au score de $0,165$ obtenu par *ds2*. Néanmoins, on s'aperçoit bien en lisant les 2 phrases que *p1* (et donc *nf1* et *ds1*) devrait moins bien répondre à la requête que *p2* car elle ne contient pas paper alors que c'est le cas de *p2* (et donc *nf2* et *ds2*). Pour pallier cela, le système de correction d'erreurs est utilisé afin d'associer chaque erreur à sa correction avec un degré de confiance δ tel que $t_{err} \overset{\delta}{\longmapsto} t_{cor}$. Cela permet ainsi de détecter que le terme teer noté t_{err} constitue un terme erroné et qu'il doit être remplacé par le terme original tree noté t_{cor}.

Afin de prendre en compte les occurrences potentielles des termes issus de la correction, il est nécessaire de modifier les formules permettant l'obtention de la pondération des termes dans les nœuds des documents (cf. formule 2) à savoir : le tf (cf. formule 3), l'idf et l'ief (cf. formule 4).

$$tf^{nf}_{t_i} = \frac{|t^{nf}_i| + \sum_{e=1}^{|t^{nf}_{cor}|} \delta_e}{|nf|} \tag{7}$$

où, $\sum_{e=1}^{|t^{nf}_{cor}|} \delta_e$ est le nombre (pondéré par la confiance δ_e) de termes erronés t^{nf}_{err} dont la correction t^{nf}_{cor} est égale au terme original t^{nf}_i.

$$idf_{t_i} = log\left(\frac{|D|}{|d_{t_i}|+\sum_{e=1}^{|d_{t_{cor}}|}\delta_e+1}\right)+1 \qquad ief_{t_i} = log\left(\frac{|NF|}{|nf_{t_i}|+\sum_{e=1}^{|nf_{t_{cor}}|}\delta_e+1}\right)+1 \tag{8}$$

où, $\sum_{e=1}^{|d_{t_{cor}}|}\delta_e$ (resp. $\sum_{e=1}^{|nf_{t_{cor}}|}\delta_e$) est le nombre (pondéré par la confiance δ_e) de documents $d_{t_{err}}$ (resp. d'éléments $nf_{t_{err}}$) contenant des termes erronés dont la correction $d_{t_{cor}}$ (resp. $nf_{t_{cor}}$) est égale au terme original d_{t_i} (resp. nf_{t_i}).

Ainsi, si on reprend l'exemple précédent en considérant un degré de confiance plutôt modéré δ de 60% dans la correction (en pratique ce degré est déterminé par le score attribué à t_{cor} par le système de correction d'erreurs), on obtient l'index corrigé selon les formules 7 et 8 présenté dans le tableau 2 :

Terme	Document	Élément	tf	idf	ief
green	ds1	nf1	0,5	0,82	0,82
	ds2	nf2	0,33		
paper	ds2	nf2	0,33	1	1
tree	ds1	nf1	0,5	0,89	0,89
	ds2	nf2	0,2		

TABLE 2 – Index modifié des documents *ds1* et *ds2* (les facteurs *idf* et *ief* sont égaux car les documents ne comportent chacun qu'un seul élément).

Une recherche comportant les mots-clés ***tree*** et ***paper*** aboutira aux scores suivants pour chacun des nœuds des deux documents :

$$\begin{aligned} s^{cor}_{ds1}(r) = s^{cor}_{nf1}(r) = p^r_{tree} \times p^{nf1}_{tree} + p^r_{paper} \times p^{nf1}_{paper} = 0,19 \\ s^{cor}_{ds2}(r) = s^{cor}_{nf2}(r) = p^r_{tree} \times p^{nf2}_{tree} + p^r_{paper} \times p^{nf2}_{paper} = 0,25 \end{aligned} \tag{9}$$

Par conséquent, le document *ds2* sera mieux classé que le document *ds1* (et cela bien que le degré de confiance dans les corrections qui a été choisi soit relativement faible), ce qui est correct compte tenu du fait que c'est ce premier qui est le plus pertinent des deux documents. L'approche proposée a servi de support à l'implémentation de nos prototypes *SnAIRS/SAIRS (Spelling (non-)Aware Information Retrieval System)* évalués ci-dessous.

4 Évaluation

Nos prototypes *SnAIRS*/*SAIRS* ont été évalués sur la collection de documents du track ad-hoc de la campagne d'évaluation INitiative for the Evaluation of XML retrieval (INEX) de 2008. Cette campagne comporte une collection de 659387 documents XML issus de Wikipedia associée à 70 requêtes évaluées. Le track ad-hoc est composé de 3 taches : *focused*, *relevant in context* et *best in context*, qui sont associées à différentes métriques permettant de les évaluer. L'objectif poursuivi suite à la participation à de telles campagnes est d'évaluer le système de RI complet *sans* puis *avec* correction d'erreurs (s'appuyant sur Aspell (Atkinson, 2011)) lors de l'indexation des documents. De cette façon, il est possible d'obtenir à la fois des indicateurs globaux sur les performances de notre système de RI (en comparant ses résultats à ceux obtenus par d'autres systèmes évalués lors de la campagne), mais aussi des indicateurs locaux nous permettant d'estimer l'impact relatif de la correction d'erreurs sur les résultats de notre système de RI.

Prototype	*SnAIRS*	*SAIRS*	Δ (%)
Volume index (Go)	*8,0*	*6,9*	*-13,75*
Durée req. min. (ms)	*2*	*1*	*-50*
Durée req. max. (ms)	13320	27279	+104,80
Durée req. moy. (ms)	605	1139	+88,26
Durée req. 1er quartile (ms)	*4*	*4*	*0*
Durée req. mediane. (ms)	5	6	+20
Durée req. 3e quartile (ms)	16	32	+100
Durée req. total (ms)	41775	78657	+88,29

TABLE 3 – Propriétés de *SnAIRS* (sans correction) et *SAIRS* (avec correction).

Bien que la collection de documents INEX ne contienne qu'un faible taux d'erreurs (les documents issus de Wikipedia sont de relativement bonne qualité), on peut constater dans la table 3 que le volume occupé par l'index est beaucoup plus important pour les mêmes documents lorsque ces derniers contiennent des erreurs même en faible quantité. Ce comportement peut s'expliquer par le fait que les erreurs constituent autant de variations des termes qui viennent augmenter le nombre d'entrées différentes dans l'index. On pourrait penser qu'un index plus petit devrait permettre une exécution plus rapide des requêtes. Bien que cela ne soit pas visible (on constate une dégradation et non pas un gain) dans la table 3, c'est effectivement le cas mais cela est contrebalancé par le fait qu'il y a un nombre plus important de correspondances dans l'index et donc un nombre plus important de résultats à retourner ce qui demande plus de temps.

La taille de l'index et le temps de réponse ne sont pas les seuls facteurs impactés par les erreurs, c'est aussi le cas de la pertinence des résultats. Les systèmes ont ainsi été évalués sur la tache *focused* qui est la plus classique car elle est dédiée à la recherche des éléments (parties de documents) les plus pertinents dans les premiers rangs des résultats de la requête. Cette tache est évaluée en fonction de la précision interpolée à 1% de rappel (iP[.01], la métrique principale), mais aussi de la moyenne des précisions interpolées (MAiP) sur les 101 points de rappel.

Participant	iP[.00]	iP[.01]	iP[.05]	iP[.10]	MAiP
SnAIRS	0.3073	0.2894	0.1788	0.1501	0.0499
SAIRS	0.3592	0.3141	0.1967	0.1694	0.0598

TABLE 4 – Résultats de *SnAIRS* (sans correction), *SAIRS* (avec correction).

On peut observer sur la table 4 que *SnAIRS* obtient une précision inférieure à *SAIRS* aux différents niveaux de rappels considérés par la campagne INEX et notamment pour la mesure officielle d'iP[.01]. La correction des erreurs à l'indexation permet donc d'obtenir une précision accrue dans les premiers niveaux de rappels. Ces résultats sont prometteurs (de nombreux paramètres peuvent être améliorés) bien qu'ils soient pour l'instant relativement éloignés du Top-10 d'INEX (Kamps *et al.*, 2009) dont les systèmes plus aboutis intègrent des mécanismes tel que l'expansion de requêtes leur permettant de mieux satisfaire aux requêtes "pauvres" composées d'un seul mot-clé.

5 Conclusion et perspectives

Dans cet article nous avons considéré un problème qui touche de façon transverse l'ensemble des applications amenées à manipuler des informations de qualité variable. Nous avons ainsi

considéré le cas des informations textuelles qui sont souvent considérées de fait comme étant "propres". Nous avons proposé une solution à ce problème pour les systèmes de RI structurés en section 3 qui pourrait être étendue à la plupart des systèmes de RI car cette dernière consiste à y intégrer un *système de correction d'erreurs* lors du processus d'indexation. Nous avons dans un premier temps identifié les contraintes spécifiques imposées par les systèmes de RI vis-à-vis des systèmes de corrections d'erreurs, et nous les avons évalués dans (Renard *et al.*, 2011). La correction d'erreurs à l'indexation présente des avantages (cf. table 3) et permet de construire des index plus représentatifs du contenu réel des documents ce qui aboutit à de meilleurs résultats (cf. table 4) que sans correction d'erreurs. De plus, la collection de documents basée sur Wikipedia ne contient que peu d'erreurs et il serait intéressant de corrompre volontairement cette dernière afin de mieux mettre en lumière l'apport de notre proposition.

Références

Atkinson, K. (2011). Correcteur Aspell. `http://aspell.net`. [consulté le 15/01/2012].

Gravier, G., Guinaudeau, C., Lecorvé, G. et Sébillot, P. (2011). Exploiting speech for automatic TV delinearization : From streams to cross-media semantic navigation. *EURASIP JIVP*, 2011(0).

Kamps, J., Geva, S., Trotman, A., Woodley, A. et Koolen, M. (2009). Overview of the INEX 2008 Ad Hoc Track. *In* Geva, S., Kamps, J. et Trotman, A., éditeurs : *Advances in Focused Retrieval*, volume 5631 de *Lecture Notes in Computer Science*, pages 1–28. Springer-Verlag.

Kantor, P. B. et Voorhees, E. M. (2000). TREC-5 Confusion Track : Comparing Retrieval Methods for Scanned Text. *Information Retrieval*, 2(2):165–176.

Renard, A., Calabretto, S. et Rumpler, B. (2011). An evaluation model for systems and resources employed in the correction of errors in textual documents. *In* Morvan, F., Tjoa, A. M. et Wagner, R. R., éditeurs : *8th International Workshop on Text-based Information Retrieval in conjunction with the 22nd International Conference DEXA 2011*, pages 160–164, Toulouse, France. IEEE Computer Society.

Ruch, P. (2002). Using contextual spelling correction to improve retrieval effectiveness in degraded text collections. *In 19th international conference on Computational linguistics-Volume 1*, volume 1, page 7. Association for Computational Linguistics.

Salton, G. (1971). *The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice Hall.

Sauvagnat, K. et Boughanem, M. (2005). Using a Relevance Propagation Method for Adhoc and Heterogeneous Tracks at INEX 2004. *In* Fuhr, N., Lalmas, M., Malik, S. et Szlavik, Z., éditeurs : *Advances in XML Information Retrieval*, volume 3493 de *Lecture Notes in Computer Science*, pages 499–532. Springer-Verlag.

Sitbon, L., Bellot, P. et Blache, P. (2007). Traitements phrastiques phonétiques pour la réécriture de phrases dysorthographiées. *In 14ème conférence TALN*, Toulouse, France.

Subramaniam, L. V., Roy, S., Faruquie, T. A. et Negi, S. (2009). A Survey of Types of Text Noise and Techniques to Handle Noisy Text. *Language*, pages 115–122.

Varnhagen, C. K., McFall, G. P., Figueredo, L., Takach, B. S., Daniels, J. et Cuthbertson, H. (2009). Spelling and the Web. *Journal of Applied Developmental Psychology*, 30(4):454–462.

Voorhees, E. M., Garofolo, J. et Sparck Jones, K. (2000). TREC-6 Spoken Document Retrieval Track. *Bulletin of the American Society for Information Science and Technology*, 26(5):18–19.