Dépendances syntaxiques et méthodes de détection de passages pour une segmentation sur le locuteur et le thème

Loïc Maisonnasse, Caroline Tambellini

Laboratoire CLIPS IMAG – Université Joseph Fourier
385, rue de la bibliothèque - BP 53
38041 Grenoble cedex9
loic.maisonnasse@imag.fr
caroline.tambellini@imag.fr

Mots-clés : découpage thématique, dépendance syntaxique
Keywords: topic segmentation, syntactic dependancy

Résumé Dans cet article, nous présentons les différentes méthodes que nous avons utilisées dans le cadre de la campagne de fouilles de texte DEFT'05. Dans un premier temps, nous présentons les différents aspects du système que nous avons développé pour effectuer la tâche de fouilles de texte. Nous présentons notamment la façon dont nous avons extrait les unités d'indexation, l'apprentissage du poids associé à ces unités, le calcul de la correspondance entre phrase et locuteur et la segmentation thématique. Nous présentons ensuite nos résultats, sur la base desquels nous envisageons d'éventuelles améliorations.

Abstract In this article, we show the methods we used for the DEFT'05 evaluation campaign. In a first time, we describe the different system aspect that we developed in order to complete the text-mining task. More particularly, we present the indexation unit extraction, the learning process used to weight these units, the matching function between sentences and speaker and the topic segmentation. Finally, we present our results at DEFT'05 and we consider some improvements.

1 Introduction

La tâche de DEFT'05 consiste à détecter des phrases de F. Mitterrand dans des allocutions de J. Chirac. Pour ce faire, nous disposons d'un corpus composé d'allocutions de J. Chirac au sein desquelles des portions d'allocutions de F. Mitterrand ont été insérées. Nous proposons ici de déterminer l'auteur d'une phrase à partir d'éléments représentatifs de la syntaxe. Il semble en effet intéressant de savoir s'il est possible de spécifier l'auteur d'une phrase par rapport aux tournures et aux constructions de phrases qu'il utilise. Nous avons également

voulu étudier la piste des changements thématiques pour déterminer les allocutions de F. Mitterrand.

Nous présentons ici notre apprentissage sur les dépendances syntaxiques, puis l'algorithme de détection des ruptures. L'ensemble de nos apprentissages a été effectué sur le corpus de la tâche 3 (celle contenant toutes les informations).

2 Apprentissage sur les dépendances syntaxiques

2.1 Méthode

Pour prendre en compte la syntaxe, nous utilisons des éléments constitutifs de la phrase qui capturent la forme de celle-ci. Le résultat d'un analyseur syntaxique est utilisée, plus particulièrement une analyse en dépendance extraite par l'analyseur 'Xerox Incremental Parser' (XIP) (Aït-Mokhtar et al., 2002). Dans le but de manipuler une structure moins complexe que l'arbre de dépendance, nous considérons le résultat de l'analyse comme un ensemble de dépendances. Chacune de ces dépendances est caractérisée par son type et la liste des lemmes qu'elle relie. Par ce formalisme, la phrase *'le chat mange la souris'* est représentée par : *'{SUBJ(chat,manger), OBJ(souris,manger)}'*

Pour tester l'approche, le corpus fourni pour l'apprentissage a été divisé en deux parties. La première (de l'allocution 100 à l'allocution 5) est utilisée pour l'apprentissage, le reste est utilisé pour l'évaluation. La figure ci-dessous décrit le processus d'apprentissage :

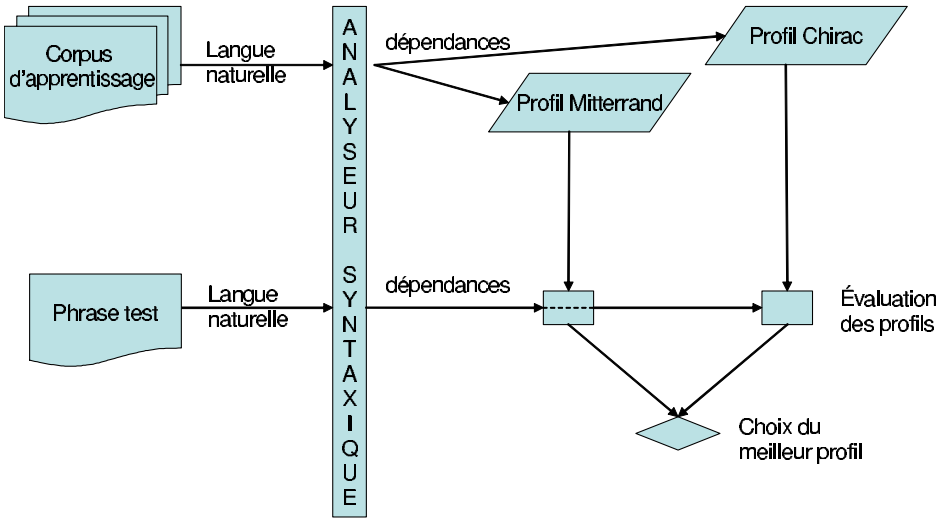

Figure 1 : Processus d'apprentissage

Les dépendances syntaxiques sont utilisées pour la détection de phrase. La première phase de notre approche consiste donc à effectuer une analyse en dépendances sur les phrases du corpus. A partir des résultats de cette analyse, la liste des dépendances syntaxiques de chaque phrase est extraite. Ces dépendances servent de base à un apprentissage permettant d'établir deux profils : un pour F. Mitterrand et un pour J. Chirac. A l'intérieur de ces deux profils, un poids est affecté à chaque dépendance en fonction de sa capacité à distinguer le profil.

Une fois les deux profils créés, chaque phrase du corpus d'évaluation est à son tour analysée. Pour chaque phrase, les dépendances extraites sont stockées sous la forme d'un vecteur et sont pondérées par rapport à leur fréquence. La similarité de ce vecteur par rapport à chaque profil est calculée. Au final, le profil fournissant la meilleure similarité est considéré comme celui correspondant à la phrase.

2.2 L'apprentissage

Nous avons évalué différents types d'apprentissages tout en nous positionnant à différents niveaux de granularité sur le corpus d'apprentissage. Dans un premier temps, nous avons concaténé l'ensemble des phrases de chaque politicien au sein de deux documents. Les dépendances extraites de ces deux documents sont stockées sous la forme de deux vecteurs représentant les profils de chaque président. Le poids de ces dépendances est calculé selon les deux pondérations ltc et lnc présentée ci-dessous :

Ltc	$w_{i,j} = \frac{\ln(f_{i,j}+1) * \log(2/df_i)}{\sqrt{\sum_i f_{i,j}^2}}$
Lnc	$w_{i,j} = \frac{\ln(f_{i,j}+1)}{\sqrt{\sum_i f_{i,j}^2}}$

Figure 2 : Pondération pour l'apprentissage global

où : $w_{i,j}$ est la pondération finale de la dépendance i pour le président j,

$f_{i,j}$ la fréquence de la dépendance i dans le discours de j,

df_i le nombre de documents contenant i (ici 1 ou 2)

Nous avons ensuite calculé les poids pour les dépendances à l'aide d'un apprentissage phrase par phrase. Le poids d'une dépendance résulte de sa répartition dans les phrases pertinentes et non pertinentes d'un profil. Ce calcul est effectué soit par la formule de *Rocchio* soit par la formule utilisée dans (Brouard, 2002) (*N*), ces deux formules sont présentées ci-dessous :

Rocchio	$w_{i,j} = \alpha \frac{\lvert Qi \cap Pj \rvert}{\lvert P_j \rvert} - \beta \frac{\lvert Qi \cap \overline{Pj} \rvert}{\lvert \overline{P_j} \rvert}$, $\alpha = \beta = 1$
N	$w_{i,j} = \frac{\lvert Q_i \cap P_j \rvert}{\lvert P_j \rvert} * \frac{\lvert Q_i \cap P_j \rvert}{\lvert Q_i \rvert}$

Figure 3 : Pondération pour l'apprentissage sur les phrases et les allocutions

Où : P_j Ensemble des documents pertinents pour le président j

Q_i Ensemble des documents contenant la dépendance i

Dans un troisième apprentissage, le poids des dépendances n'est plus calculé phrase par phrase mais par rapport au regroupement par allocution des phrases de chaque président. Les deux mêmes formules d'apprentissage que précédemment sont utilisées.

Une dernière méthode consiste à regrouper les apprentissages de différentes granularités.

2.3 Evaluation

Nous avons appliqué les méthodes précédentes sur notre corpus d'apprentissage. Pour les différentes méthodes, les profils obtenus ont été utilisés pour extraire les phrases de F. Mitterrand du corpus d'évaluation. Lors de cette évaluation les résultats suivants ont été obtenus :

	pondération	fscore
apprentissage global	*ltc*	0,3138
	lnc	0,2400
apprentissage par phrase	*N*	0,1843
	Rocchio	0,1814
apprentissage par allocution	*N*	**0,3286**
	Rochhio	0,2411

Figure 4 : Résultats des différents apprentissages

Le meilleur résultat est celui obtenus avec une pondération N sur les allocutions avec un F-score proche de 0,33. Les moins bons résultats sont ceux obtenus à l'aide de l'apprentissage sur les phrases, cela semble être dû au fait que les phrases prises seules constituent de trop petits éléments d'apprentissage. L'apprentissage global donne de bons résultats notamment par l'utilisation de la pondération *ltc*. En considération de ces résultats, nous avons regroupé l'apprentissage global avec l'apprentissage sur les allocutions, l'apprentissage global est utilisé comme poids initial dans le nouvel apprentissage. Les résultats obtenus sont présentés dans le tableau suivant :

	coef	F-score
Ltc et N	1	0,3195
ltc et Rocchio	1	0,3340

Figure 5 : Regroupement apprentissage global et apprentissage sur les phrases

La pondération globale ltc combinée avec l'apprentissage basé sur la pondération N sur les allocutions fournit des résultats inférieurs à ceux obtenus par la simple formule N. Ces deux formules ne sont donc pas complémentaires. Au contraire, l'apprentissage global ltc combinée avec un apprentissage sur les allocutions de type Rocchio améliore les résultats de base et dépasse les résultats obtenus à l'aide de l'apprentissage N sur les allocutions.

3 Diffusion

Au sein de l'évaluation DEFT'05, les phrases de F. Mitterrand insérées dans les allocutions de J. Chirac sont regroupées. Notre apprentissage ne tient pas compte de cette caractéristique en traitant chaque phrase indépendamment. Nous avons donc mis en place une méthode qui prend en compte le score relatif au locuteur des phrases voisines dans le calcul du score d'une phrase.

3.1 Méthode

Une fois l'ensemble des phrases évaluées par l'une des méthodes d'apprentissage, le score de chaque phrase est recalculé par rapport aux scores des phrases voisines à l'aide de fonctions de diffusion (Huang, 97). Le calcul s'effectue à l'aide de l'équation suivante pour laquelle nous avons testé deux fonctions de diffusion différentes (*A* et *B*) basées sur des cosinus :

$Pt_i = P_i + \sum_{j\in[1,N]} f(j)*(P_{i-j} + P_{i+j})$	
A	$f(j) = \cos(\frac{j*\pi}{2N})$
B	$f(j) = \cos(\frac{j*\pi}{N}) + 0.5$

Figure 6 : Fonctions de diffusion

Où Pt_i Poids final de la i-éme phrase (par ordre de lecture) pour un président
P_i Poids de la i-éme phrase pour un président obtenu par apprentissage
N Taille de la fenêtre des phrases voisines prises en compte

3.2 Evaluation

Pour tester les fonctions de diffusion et la taille de la fenêtre, nous avons utilisé les résultats obtenus lors de l'apprentissage à l'aide de la combinaison Rocchio et ltc. Les résultats de cet apprentissage ont donc été recalculés à l'aide des fonctions précédentes et en faisant varier la taille de la fenêtre utilisée.

	taille de la fenêtre			
fonction de lissage	5	6	7	8
A	0,6664	0,6736	0,6691	-
B	-	0,6696	**0,6742**	0,6731

Figure 7 : Résultats du lissage (F-score)

La fonction de lissage B donne de meilleurs résultats. La taille de fenêtre qui semble la plus adaptée est la taille 7.

4 Découpage thématique

Nous avons voulu étudier la piste des changements thématiques pour déterminer les allocutions de F. Mitterrand. Pour ce faire, nous avons utilisé la méthode du TextTiling de Hearst (Hearst, 1997).

4.1 Principe du TextTiling

Le système de TextTiling (Hearst, 1997) est un système qui recherche les ruptures de thèmes et les identifie lorsqu'un bloc du document présente un moins grand nombre de mots traitant du thème.

Le système de TextTiling (Figure 8) découpe tout d'abord (1) le document en blocs composés d'un nombre fixe de phrases (3 à 5 phrases généralement). Ensuite, (2) toutes les paires des blocs adjacents de texte sont comparées et une valeur de similarité leur est attribuée. (3) La suite résultante des valeurs de similarités, après être mise sous forme de graphe et aplanie, est examinée pour déterminer les pics et les vallées sur le graphique. (4) Des valeurs de similarités élevées, impliquant que les blocs adjacents se suivent de façon logique, sont susceptibles de former des pics, tandis que des valeurs de similarités faibles, indiquant une potentielle limite entre les blocs, créent des vallées. Un pic correspond donc à deux blocs fortement liés thématiquement alors qu'une vallée correspond à une rupture de thème. Chaque vallée est donc considérée comme une rupture de thèmes et correspond à une limite entre deux blocs thématiquement différents.

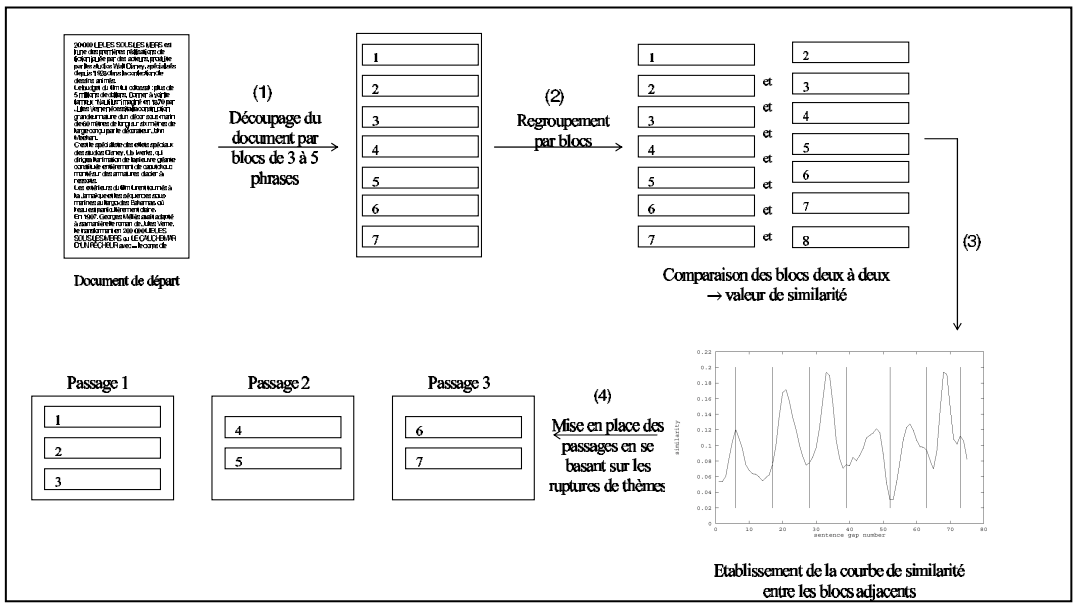

Figure 8 : Méthode du TextTiling

Il a été démontré que ce type d'analyse donne de bons résultats sur des textes dont les termes caractéristiques des thèmes développés ne possèdent pas de synonymes.

4.2 Adaptation au contexte DEFT'05

Afin de déterminer les changements thématiques, nous avons implémenté une adaptation de la méthode du TextTiling. Nous avons gardé le principe du TextTiling et nous l'avons adapté au contexte des tours de parole. Nous détaillons ici les principales étapes du processus :

- Découpage du document en blocs (1)
- Calcul de similarité des blocs pris 2 à 2 (2)
- Détermination du seuil permettant d'identifier les ruptures thématiques (3)
- Détermination des ruptures (4)

Le nombre de phrases constituant les blocs est de 15 tours de parole. Cette valeur a été déterminée suite à différentes expérimentations avec des blocs composés de 10 à 20 tours de parole. L'utilisation de 15 tours de parole pour former un bloc donne les meilleurs résultats. Cette valeur se justifie car les allocutions de F. Mitterrand ont une longueur moyenne de 18 tours de paroles et la majorité des allocutions ont une longueur proche de 15 tours de paroles. Une fois ces blocs de 15 tours de paroles formés (1), la similarité entre les blocs pris deux à deux est calculée (2) :

$$sim(a,b) = \frac{\sum_{t=1}^{n} w_{t,a} w_{t,b}}{\sqrt{\sum_{t=1}^{n} w_{t,a}^2 \sum_{t=1}^{n} w_{t,b}^2}}$$

où t varie pour tous les termes du document et $w_{t,a}$ est le poids tf.idf[1] assigné au terme t dans le bloc a.

Une fois cette similarité calculée, il faut fixer le seuil qui permettra d'identifier les ruptures. Pour ce faire, nous avons effectué plusieurs expérimentations et nous avons obtenus les meilleurs résultats en prenant un seuil correspondant aux 50 % des valeurs de similarités les plus faibles. Plus précisément, nous calculons les valeurs de similarité des blocs pris 2 à 2 (soit 1810 valeurs de similarité dans notre cas), puis nous les ordonnons par ordre décroissant, on se base alors sur la valeur médiane de similarité (soit la valeur de similarité à la position 905, une fois les valeurs ordonnées par ordre décroissant). Cette valeur correspond au seuil (3). Enfin, pour chaque valeur inférieure au seuil, on considère qu'il existe une rupture thématique entre les deux blocs correspondant à cette valeur de similarité (4). Si la valeur de similarité entre un bloc A et un bloc B est inférieure au seuil, la première phrase du bloc B est renvoyée dans un fichier résultat pour être ensuite utilisée dans le processus de détermination des phrases de F. Mitterrand (Figure 9). La première phrase du bloc B sert donc de délimiteur de passages.

---

[1] $\text{Tf.idf} = \frac{\text{nombre d'apparitions du terme dans le bloc}}{\text{nombre de blocs contenant le terme}}$

Les passages ainsi obtenus sont utilisés pour modifier le score des phrases. Le score de chaque passage est calculé en effectuant la moyenne des scores des phrases qu'il contient. Le score final d'une phrase résulte de la somme pondéré entre son score initial et le score du passage dans lequel elle se situe.

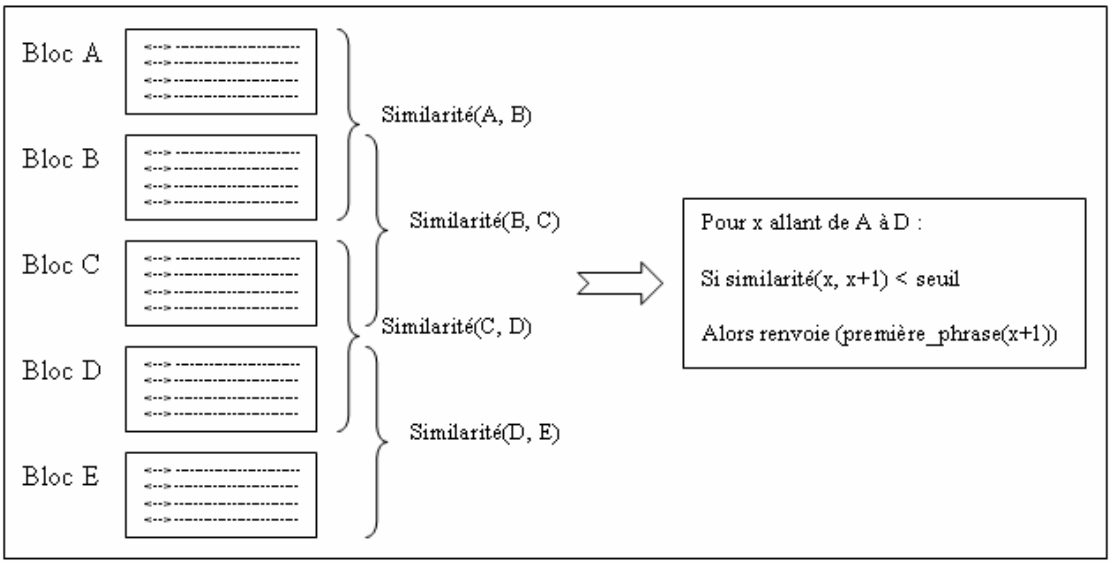

Figure 9 : Principe de détermination des ruptures

4.3 Limites de la méthode

Cette méthode de découpage thématique connaît quelques limites. Tout d'abord, pour avoir de bons résultats, il faut que les allocutions des différents présidents soient thématiquement bien différentes. En effet, pour qu'une rupture thématique soit déterminée il faut que l'on rencontre une différence de similarité entre deux blocs consécutifs. D'autre part, compte tenu du fait que l'on compare des blocs, la rupture thématique ne pourra être déterminée qu'entre deux blocs, or celle-ci peut avoir effectivement lieu au milieu d'un bloc. La méthode du TextTiling ne nous permet pas de le détecter. On constate donc que le choix de la taille du bloc est un problème important dans le bon fonctionnement du système.

5 Soumissions

5.1 Processus

Nos soumissions à l'évaluation DEFT'05 ont consisté à enchaîner les différents modules présentés dans les parties précédentes. Notre approche n'ayant été évaluée que pour la tâche 3, les mêmes exécutions ont été soumises pour les trois tâches (Figure 10).

- L'exécution 1 consiste en un apprentissage global en ltc couplé avec un apprentissage sur les allocutions de type Rocchio (voir partie 2). Sur cet apprentissage, une diffusion basée sur la fonction B avec une taille de fenêtre 7 est appliquée.

- L'exécution 2 est similaire à la première mais l'apprentissage de type Rocchio est effectué sur les phrases.
- L'exécution 3 est similaire à celle de la tâche 1, à la différence prés que les résultats de l'apprentissage sont d'abord modifiés en prenant en compte le poids global des passages détectés. La diffusion n'est effectuée qu'après cette étape.

Exécution 1	Exécution 2	Exécution 3
Ltc + Rocchio (allocutions) + diffusion	Ltc + Rocchio (phrases) + diffusion	Ltc + Rocchio (allocutions) + découpage thématique + diffusion

Figure 10 : Récapitulatif des trois exécutions

5.2 Résultats

Les résultats des trois tâches sont comparés aux moyennes de l'évaluation DEFT'05 comme présentés dans le tableau suivant :

tâche	exécution	précision	rappel	F-score
1	moyenne	-	-	0,6229
	1	0,7477	**0,7549**	**0,7513**
	2	0,9265	0,4216	0,5795
	3	**0,9415**	0,2669	*0,4159*
2	moyenne	-	-	0,6738
	1	0,7533	**0,7563**	**0,7548**
	2	**0,9246**	0,4300	0,5871
	3	0,7943	0,6725	0,7283
3	moyenne	-	-	0,6902
	1	0,7534	**0,7568**	**0,7551**
	2	**0,9268**	0,4337	0,5909
	3	0,7923	0,6725	0,7275

Figure 11 : Résultats d'évaluation DEFT'05

Les meilleurs résultats obtenus dans les trois tâches sont ceux obtenus à l'aide de l'exécution 1. L'utilisation des phrases à la place des allocutions donne un f-score largement inférieur, mais la précision obtenue est améliorée. La troisième exécution donne des résultats intermédiaires sauf pour la première tâche où le résultat est faible. Il est intéressant de remarquer que nos résultats sont stables sur les trois tâches. Cette stabilité peut s'expliquer par le fait que notre système utilise plus la syntaxe que les informations tel que les noms et les dates et que par conséquent celui-ci est peu affecté par leur suppression.

6 Conclusion et perspectives

Notre participation à DEFT'05 nous a permis d'évaluer l'intérêt d'une approche basée sur des éléments représentatifs de la syntaxe pour la détection de phrase ainsi que l'intérêt de la détection de passages dans un contexte de fouilles de texte. Les résultats obtenus montre que dans ce cas les dépendances syntaxiques permettent d'obtenir de bonnes extractions. Filtrer les dépendances extraites dans l'objectif de conserver les plus discriminantes permettrait d'améliorer le résultat. On peut remarquer également que la présence ou non d'informations telles que des dates ou des noms a peu d'impact sur une telle méthode. Toutefois, la prise en compte de telles informations dans un module supplémentaire, permettrait d'améliorer notre système. La détection de passages permet d'améliorer la précision du système. Une détection plus précise des ruptures permettrait d'augmenter cette précision. Une première solution serait l'utilisation d'un pré-traitement du corpus utilisant un anti-dictionnaire et une fonction de stemming.

Références

AÏT-MOKHTAR S., CHANOD J.P., ROUX C. (2002), Robustness beyond shallowness : Incremental Deep Parsing. Actes de *the Natural Language Engineering Journal on Robust Methods in Analysis of Natural Language Data*, Cambridge University Press, pp. 121-144.

BROUARD C. (2002), *RELIEFS : un système d'inspiration cognitive pour le filtrage adaptatif de documents textuels*, Actes de Revue des Sciences et Technologies de l'Information, vol7, no1/2, pp157-182.

HEARST M.A. (1997), TextTiling: Segmenting Text into Multi-paragraphe Subtopic Passage, Actes de *Computational Linguistics*, pp. 33-64.

HUANG C., Principle of information diffusion, Actes de *Fuzzy Sets and Systems*, 91, p. 69-90, 1997.