Étude Comparative des Algorithmes de Segmentation Thématique Pour la Langue Arabe

Fouzi Harrag [1], Mohamed BenMohammed [2]

[1] Département d'informatique
Université Farhat Abbas
Sétif – Algérie
hfouzi2001@yahoo.fr

[2] Département d'informatique
Université Mentouri,
Constantine, Algérie
ben_moh123@yahoo.com

Résumé. Le besoin d'avoir un système de segmentation thématique des textes arabes a pour but d'améliorer les fonctionnalités de la Recherche d'Information Arabe (RIA). La segmentation thématique des textes a été utilisée pour améliorer la précision des processus subséquents telle que les systèmes de résumé automatique, les systèmes de Question/Réponses et les systèmes de recherche d'information. Dans cet article, nous présentons une étude comparative des algorithmes *TextTiling* et *C99* pour la segmentation thématique des textes arabes. Nous évaluons la performance de ces deux algorithmes en utilisant les mesures classiques Rappel et Précision et la méthode des Jugements des Lecteurs récemment introduite.

1 Introduction

La segmentation thématique est une nouvelle technique pour l'amélioration de l'accès à l'information, elle peut être définie comme la tâche de subdivision d'un document en plusieurs paragraphes thématiquement cohérents. En recherche d'information par exemple, avoir des documents thématiquement segmentés peut résulter en la récupération des segments de texte courts et pertinents qui correspondent directement à la requête d'un utilisateur au lieu de longs documents examiné avec soin par l'utilisateur pour trouver l'objet de son intérêt. Avoir des documents thématiquement segmentés peut aussi aider dans la tâche de résumé automatique des textes puisque un meilleur résumé peut être obtenu de la fusion des différents segments constituant le document [7]. Au temps où un nombre considérable de recherches a été consacré à l'étude cette technique pour les langues anglaise et française, peu l'ont étudié pour d'autres langues et presque personne, à l'exception de [7] et [12], n'a étudié cette technique pour langue arabe. Le manque de recherche dans ce domaine nous a poussés à adopter les deux algorithmes de segmentation thématique *TextTiling* et *C99* pour une telle langue. Cet article est organisé comme

suit: la Section 2 présents un état de l'art dans le domaine; la Section 3 présents une vue d'ensemble des approches implémentés; les résultats et leur discussion sont rapportées dans la Section 4; finalement la Section 5 conclut l'article.

2 Travaux antérieurs

Les approches qui adressent le problème de segmentation thématique peuvent être classées en deux classes : les approches à base de connaissance et les approches à base de mot. Les systèmes à base de connaissance, comme dans [11], exige un grand effort manuel de l'ingénierie de connaissance pour la création d'une base de connaissance (réseau sémantique et/ou de Frames). Ceci est seulement réalisable dans quelques domaines très restreints. Pour dépasser cette limitation, et pour traiter une grande quantité de textes, les approches à base de mot ont été développées. [13] et [20] fait usage de la distribution des mots dans un texte pour trouver une segmentation thématique. Ces travaux sont bien adaptés à des textes techniques ou scientifiques caractérisés par un vocabulaire spécifique.

Pour traiter des textes narratifs ou explicatifs tels que les articles des journaux, les approches [17] et [22] sont basées sur la cohésion lexicale calculée à partir d'un réseau lexical. Ces méthodes dépendent de la présence du vocabulaire du texte à l'intérieur de leur réseau. Donc, pour éviter toute restriction de domaines dans tels genres de textes, [20] a présenté une méthode mixte qui augmente un système basée sur la distribution des mots, en utilisant les connaissances représentés par un réseau lexical de co-occurrences construit automatiquement à partir d'un corpus.

Les autres approches Existantes de segmentation thématique peuvent être classées dans deux groupes principaux: les approches à base de cohésion lexicale et les approches à base d'attributs. Les approches à base de cohésion lexicale dépendent de la tendance des unités thématiques à lier ensemble. En outre, les approches qui mesurent ce type de cohésion peuvent être divisées en deux catégories: les approches à base de similarité où les modèles de répétitions syntactiques sont utilisés pour indiquer la cohésion et les approches à base de chaînes lexicales où autres aspects de cohésion lexicale (comme les relations entre termes) sont aussi analysé [3].

3 Approches implémentés

Dans cette section, deux algorithmes de segmentation thématique des textes sont décrits: *TextTiling* [13] et *C99* [5]. Les deux systèmes sont basés sur la cohésion lexicale. L'algorithme *TextTiling* utilise la mesure de similarité *Cosine* entre les vecteurs des termes pour mesurer la densité de la cohésion entre blocs adjacents. L'algorithme C99 utilise aussi la mesure de similarité *Cosine* pour déterminer des ressemblances parmi les phrases du texte puis il projette ceux-ci graphiquement. Il applique alors des techniques de traitement d'image pour déterminer des frontières thématiques.

3.1 Pré-traitement des textes

L'étape de pré-traitement traite les flux d'entrée en enlevant les étiquettes et les ponctuations et en transformant les termes en lemmes. En premier lieu, nous allons construire des blocs de texte appelés « séquences lexicales ». Le texte de l'entrée est simplement une séquence de caractères avant le pré-traitement. C'est la responsabilité du pré-processor de transformer cette séquence en unités sémantiques dans la phase d'analyse lexicale. Ces unités peuvent être des mots simples tels que les mots programme et création, ou des expressions composées telles que Les États-Unis (par opposition à États et Unis).

3.2 L'Algorithme TextTiling

L'algorithme *TextTiling*, pour la découverte des structures thématiques en utilisant la répétition des termes, se décompose de trois parties principales [13]:

- Le découpage physique.
- Détermination de la similarité.
- Identification des frontières.

C'est l'un des travaux fondateurs dans le domaine de la détection de thème, *TextTiling* réalise le découpage d'un texte en unités de discours multi paragraphe cohérentes qui reflète la structure thématique du texte cf. Figure 1. Cet algorithme utilise la fréquence lexicale indépendamment du domaine et la distributivité pour reconnaitre l'interaction de thèmes simultanés multiples. Elle se base sur un modèle d'espace vectoriel qui détermine la similarité entre des groupes voisins de phrases et place une coupure entre des blocs voisins dissimilaires.

La première étape est le découpage physique Elle se base sur une mesure de similarité lexicale. Les lemmes issus du texte prétraites sont groupes en pseudo phrases, c'est-a-dire un ensemble de lemmes adjacents (20 dans l'article), qui sont elles-mêmes regroupées en bloc de Taille fixée par l'utilisateur (cf. Figure 1). Cette taille des segments est variable, elle peut aller de 3 à 5 pseudo phrases a un paragraphe. En général, on prend la moyenne de la longueur des Paragraphes. Les paragraphes réels ainsi que les phrases ne sont pas pris car leur longueur Peut être fortement irrégulière conduisant à des comparaisons déséquilibrées.

La deuxième étape est le calcul de la similarité entre blocs adjacents La similarité entre des blocs de pseudo phrase adjacents est calculée cf. Figure 1 par Une mesure du cosinus cf. Equation 1 : étant donne des blocs de textes b1 et b2,

$$Score(i) = \frac{\sum_t W_{t,b1} W_{t,b2}}{\sqrt{\sum_t W_{t,b1}^2 \sum_t W_{t,b2}^2}} \quad (1)$$

Où t s'étend à l'ensemble des termes dans le document et $w_{t,b1}$ est le poids *tf.idf* assigné au terme t dans le bloc *b1*. *tf.idf* correspond au nombre de lemmes communs et au

nombre de fois qu'ils apparaissent dans le texte tout entier. Donc, si le score de la similarité entre deux blocs est élève, alors non seulement les Blocs ont des termes en commun, mais les termes qu'ils ont en commun sont relativement rares en ce qui concerne le reste du document. L'évidence de la réciproque n'est pas aussi concluante : si des blocs adjacents ont une mesure de similarité faible, cela ne signifie pas nécessairement qu'ils ne se tiennent pas ensemble ; cependant, en pratique cette évidence négative est souvent justifiée.

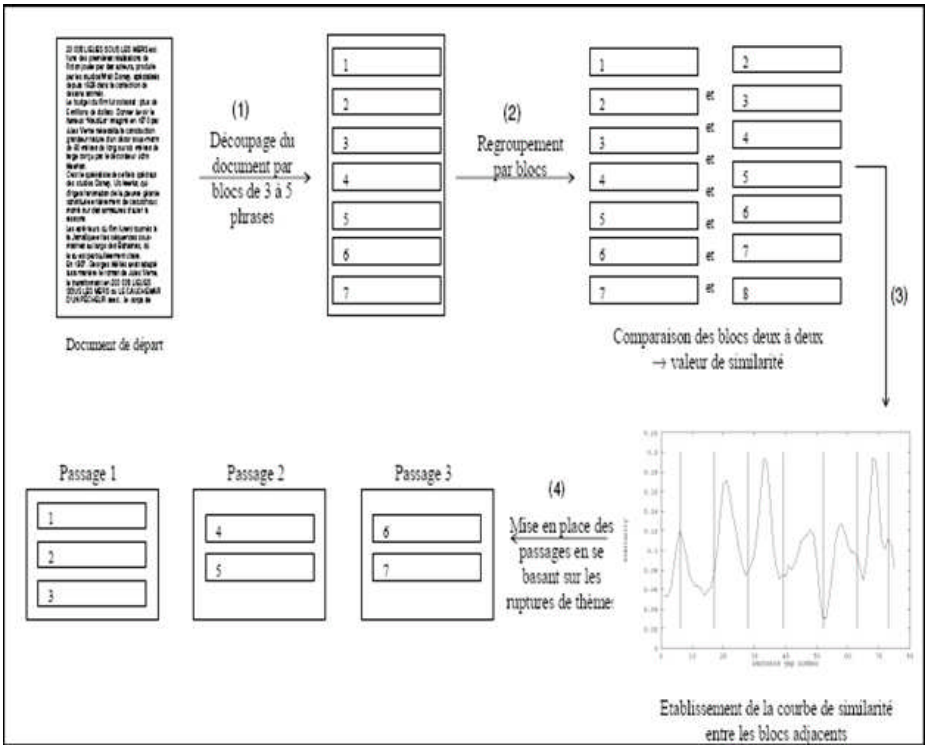

Fig. 1. Principe de l'algorithme *TextTiling*

La troisième étape est l'extraction des zones thématiques, à partir de ce score, le calcul d'un score de cohésion (ou de profondeur) est effectue qui quantifie la similarité entre un bloc et les blocs voisins. En terme de graphe de score de Similarité, un score de cohésion peut être représente comme la somme des différences entre le sommet du pic et les creux des vallées voisines. Le calcul des scores de cohésion procède comme suit:

- on commence au premier creux entre 2 blocs et,
- on mémorise le score de similarité associée avec les blocs de chaque cote du creux.
- On vérifie le score de similarité du creux précédant,

- Si c'est plus haut, on continue et on examine le score de similarité du creux précédant.
- On continue jusqu'à ce que le score soit plus bas que celui déjà examiner.
- Ensuite, on soustrait le score de similarité du creux initial avec le score maximum de similarité rencontre.
- Cette procédure est répétée pour les creux entre les blocs suivant le premier creux.
- Enfin, la somme des deux différences est calculée.

Cette valeur est le score de cohésion pour le premier creux examine, les scores de cohésion ne sont calcules que pour les creux qui sont des minimaux locaux pour la fonction de similarité. Les limites, c'est-a-dire les zones de changements de thèmes, sont déterminées en localisant les portions les plus basses des vallées dans le graphique résultant. En d'autres termes, les creux avec de fort score de cohésion sont sélectionnes comme les endroits de rupture de thèmes. Cette coupure est ajustée a la fin d'un paragraphe. Ceci permet d'éliminer les coupures très proches l'une de l'autre.

3.3 L'algorithme C99

Cet algorithme proposé par [5] utilise une mesure de similarité entre chaque unité textuelle. L'idée de base de cette méthode est que les mesures de similarité entre des segments de textes courts sont statistiquement insignifiantes, et que donc seul des classements locaux (voir ci-dessous) sont à considérer pour ensuite appliquer un algorithme de catégorisation sur la matrice de similarité.
Dans un premier temps, une matrice de similarité est donc construite, représentant la similarité entre toutes les phrases du texte à l'aide de la mesure de similarité *Cosinus*, calculée pour chaque paire de phrases du texte, en utilisant chaque mot commun entre les phrases, et après « nettoyage » du texte : suppression des mots vides et lemmatisation.
On effectue ensuite un « classement local », en déterminant pour chaque paire d'unités textuelles, le rang de sa mesure de similarité par rapport à ses $m \times n - 1$ voisins, $m \times n$ étant le masque de classement choisi. Le rang est le nombre d'éléments voisins ayant une mesure de similarité plus faible, conservé sous la forme d'un ratio r afin de prendre en compte les effets de bord.

$$r = \frac{rang}{\#\ de\ voisins\ dans\ le\ masque} \quad . \qquad (2)$$

Enfin, la dernière étape détermine les limites de chaque segment de la même manière que l'algorithme *Dotplotting* [24] emploie la maximisation. En effet on cherche à déterminer quelle configuration offre la plus grande densité, en recherchant une nouvelle limite thématique à chaque étape.
Les segments sont alors représentés par des carrés le long de la diagonale de la matrice de similarité modifiée avec les classements locaux. Pour chaque segment de la répartition proposée à une étape de la segmentation on considère son aire notée a_k et

son poids s_k qui est la somme des tous les rangs des phrases qu'il contient. On calcule alors la densité D de la configuration avec :

$$D = \frac{\sum_{k=1}^{m} s_k}{\sum_{k=1}^{m} a_k} \quad . \qquad (3)$$

L'algorithme s'arrête lorsque la densité de la meilleure répartition proposée est suffisamment faible, ou si le nombre de frontières thématiques est déjà déterminé, lorsqu'il est atteint.

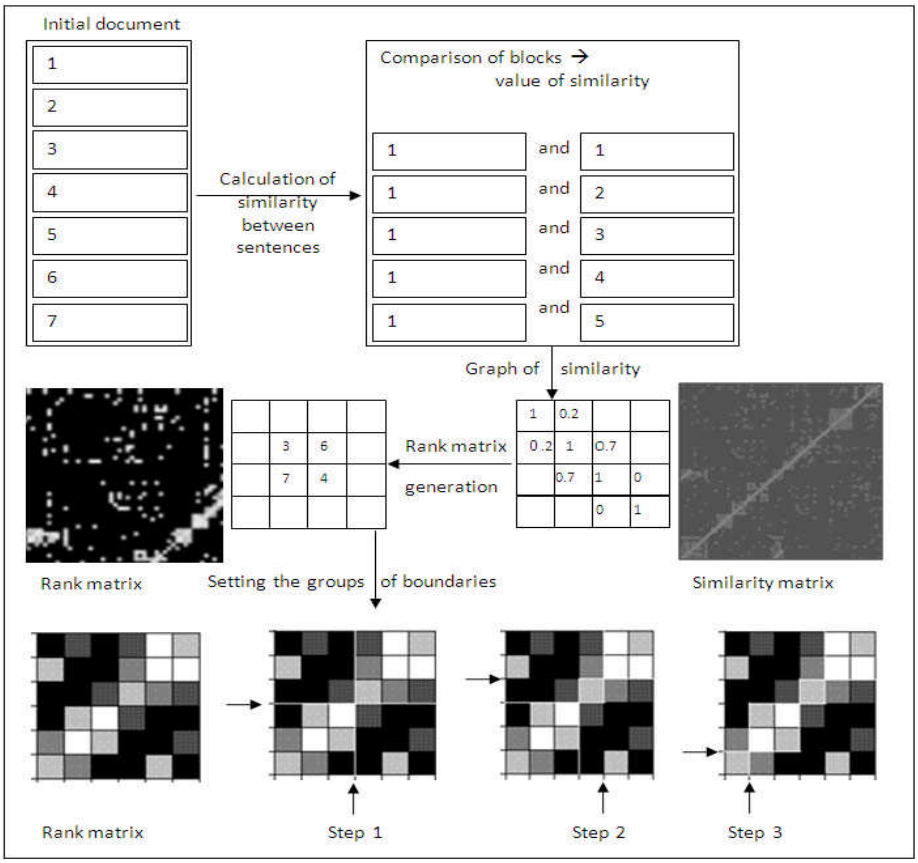

Fig. 2. Principe de l'algorithme *C99*.

4 Résultats et discussion

4.1 Critères d'évaluation

L'évaluation de la segmentation thématique peut se faire de plusieurs manières :

- Par comparaison avec des jugements humains : aucun corpus segmenté de taille suffisante n'est cependant disponible à ce jour ; des propositions ont été faites pour la constitution d'un tel corpus et pour évaluer la qualité des jugements humains [4] [13] [24].
- Par rapport à des marques déposées par l'auteur du texte (cette procédure n'est pas fiable car toute segmentation est subjective [24], la position des marques de segmentation dépend du point de vue du lecteur) ;
- Par rapport à des marques « certaines » à retrouver (limites entre documents d'un corpus par exemple);
- Par son impact sur une tâche particulière (évaluation fonctionnelle), la recherche d'informations par exemple.

4.2 Le Corpus d'évaluation

Pour l'évaluation des deux algorithmes *TextTiling* et *C99*, on se base sur les jugements de sept lecteurs, chaque lecteur parmi les sept a fait la lecture et la segmentation manuelle de 5 textes arabes traitant des sujets de deux domaines différents (Littérature, Médecine). Les textes utilisés pour cette évaluation ont une longueur moyenne entre 600 et 2000 mots. Les lecteurs ont été invités simplement à délimiter les paragraphes auxquels il y a un changement de thème, cette délimitation restera subjective pour chaque lecteur.

4.3 Méthode de Jugements des Lecteurs:

Le schéma de la figure (Fig.3) montre les limites faites par les sept lecteurs sur les textes. Ce schéma nous aide à illustrer les tendances générales des évaluations des lecteurs, et également à montrer où/et combien de fois ils sont en accord ou en désaccord. Par exemple, tous les lecteurs sauf le quatrième ont marqué une frontière au paragraphe 7. Ce lecteur en désaccord avec les autres a délimité la frontière au paragraphe 10. L'ensemble des frontières pour lesquelles les lecteurs sont tous en accord sont les suivants: {12, 20, 22, 31, 33, 37, 38, 50}. Par contre, il y a un désaccord pour les frontières suivantes: {1, 15, 18, 41,43, 44, 45 …}.

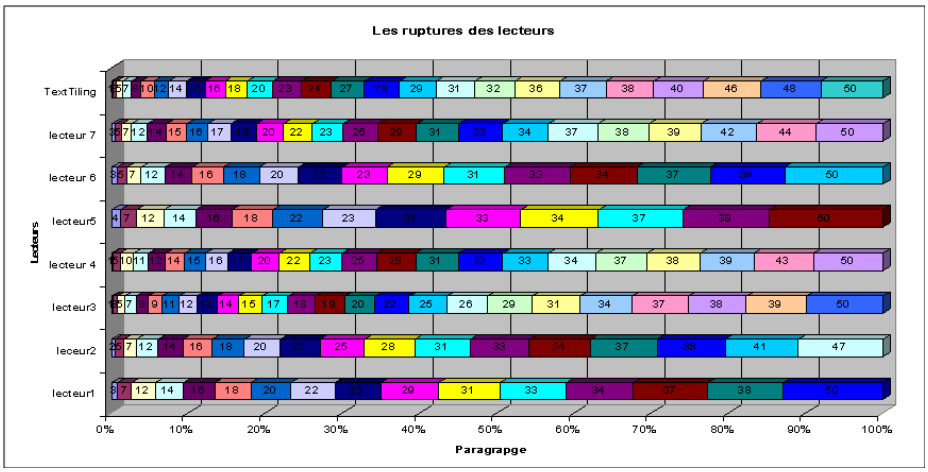

Fig. 3. Les ruptures mises par les lecteurs et l'algorithme TextTiling

D'après [24], si quatre ou plus sur sept lecteurs marquent la même frontière, la segmentation s'avérée. Mais, deux années après [18], ont montré que trois lecteurs sont considérés suffisamment pour classifier ce point comme une frontière "principale". [4] et [14] précisent l'importance de tenir en compte l'accord fortuit et prévu en calculant si les lecteurs convenir de manière significative. A cette fin, Ils conseillent d'utiliser le coefficient de *Kappa* (*K*). S'accorder à [4], *K* mesure par paires l'accord parmi un ensemble de lecteurs faisant des catégories de jugements, calculant selon l'équation (4)

$$K = \frac{P(A) - P(E)}{1 - P(E)} \quad . \qquad (4)$$

Où P (A) est la proportion de fois que les lecteurs conviennent et P(E) est la proportion de fois où on s'attendrait à ce qu'ils conviennent par hasard. Le coefficient peut être calculé en faisant par paires des comparaisons contre un expert ou en comparant à une décision de groupe. [4] déclare également que si K > 0.8 ceci signale que la segmentation est bonne, et si K > 0. 67 et K < 0.8 cela permet de donner des conclusions expérimentales acceptables. Les coefficients trouvés par [14] se sont étendus du 0.43 au 0.68 pour trois lecteurs, et ceux trouvées par [4] sont étendus du 0.65 à 0.90 pour quatre lecteurs segmentant des phrases.

Dans notre évaluation, nous concéderons que trois jugements en accord sont acceptables pour considérer la frontière juste. A partir de la figure (Fig.3) l'ensemble des frontières acceptables est le suivant : {1, 3, 5, 7, 12, 14, 15, 16, 18, 20, 22, 23, 29, 31, 33, 34, 37, 38, 50}. A partir du schéma de la même figure on peut calculer le coefficient Kappa comme il est montré dans le tableau 1 ci-dessous, la comparaison de nos résultats avec celles obtenus par Hearst [13] à partir de l'application de

l'algorithme *TextTiling* sur un corpus anglais a montré que notre segmentation est acceptable.

Table 1. Résultats de calcul du coefficient Kappa

P(A)	P(E)	K	K (H)	Remarque
0.7894	0.2106	**0.7332**	0.647	**Acceptable**

4.4 Méthode de Rappel / Précision:

Dans l'expérience suivante, les deux mesures rappel et précision, classiquement utilisés dans la recherche d'information, détaillés dans [1], ont aussi été employés pour évaluer les algorithmes de segmentation. Dans le contexte de segmentation thématique, la précision est définie comme:

$$P = \frac{Nombre\ de\ frontières\ correctement\ détectées\ par\ le\ système}{nombre\ \ totale\ \ de\ frontières\ générées\ par\ le\ système} \quad . \quad (\mathbf{5})$$

Tandis que le rappel est défini comme:

$$R = \frac{Nombre\ de\ frontières\ correctement\ détectées\ par\ le\ système}{nombre\ \ totale\ \ des\ \ frontières\ de\ référence} \quad . \quad (\mathbf{6})$$

Les valeurs de Rappel et Précision pour les deux algorithmes nous donnent une idée générale sur l'échec de ces deux mesures traditionnelles de la recherche d'information dans la tache d'évaluation des performances des systèmes de segmentation [11]. Le tableau 2 présente les valeurs de rappel et de précision pour cinq textes du corpus de référence segmentés par l'algorithme *TextTiling*. On voit bien que les valeurs de rappel pour cet algorithme sont très basses, allant de 0.00 jusqu'à 0.60, tandis que les valeurs de précision sont hautes, allant de 0.40 jusqu'à 1.00.

Table 2. Rappel et Précision pour 5 textes segmentés avec l'algorithme *TextTiling*

Texte	Nombre total de frontières	Nombre de frontières en accords	Rappel	Précision
1	6	6	0.00	1.00
2	4	4	0.00	1.00
3	3	2	0.33	0.66
4	5	2	0.60	0.40
5	1	1	0.00	1.00

Cependant, ces valeurs ne prennent pas en compte le fait que l'algorithme *TextTiling* malgré qu'il échoue dans la détection correcte des frontières, il ne manque de détecter toutes les frontières. Le tableau 3 présente les valeurs de rappel et de

précision pour les cinq textes segmentés par l'algorithme C99. On remarque que l'algorithme C99 a de hautes valeurs du rappel, 0.33, 0.40, 0.50 et 1 respectivement, Alors que Les valeurs de précision sont entre 0.50 et 0.66.

Table 3. Rappel et Précision pour 5 textes segmentés avec l'algorithme C99

Texte	Nombre total de frontières	Nombre de frontières en accords	Rappel	Précision
1	6	3	0.50	0.50
2	4	2	0.50	0.50
3	3	2	0.33	0.66
4	5	3	0.40	0.60
5	1	0	1.00	0.00

Le tableau 4 présente les résultats de comparaison entre les deux algorithmes et les jugements des lecteurs. Pour les algorithmes, *TextTiling* a la meilleure valeur pour la précision; il dépasse 0.84 mais il a la plus mauvaise valeur pour rappel qui est égale 0.15. C99 a la plus mauvaise valeur de précision 0.45 mais il a la meilleure valeur pour le rappel; il dépasse 0.54. TextTiling et C99 paraissent avoir des difficultés à s'adapter avec le nombre de frontières à découvrir; la longueur du texte a un grand impact sur leur nombre de frontières détectées. L'algorithme C99 paraît être plus effectif aux textes arabes.

Table 4. Comparaison des algorithmes avec les jugements des lecteurs

Segmentation	Rappel	Précision
TextTiling	0.18	**0.81**
C99	**0.54**	0.45
Les jugements des lecteurs	0.15	0.84

5 Conclusion

Dans cet article, une analyse comparative de deux algorithmes de segmentation thématique des textes arabes est présentée. Pour évaluer les performances de chaque algorithme sur des corpus arabe, chacun a été appliqué sur un ensemble de textes arabes et les résultats ont été comparés. Nous avons confirmé dans cet article que la tâche de segmentation est dure à évaluer parce que les objectifs peuvent varier. Globalement l'algorithme TextTiling paraît être plus adapté à la langue arabe que celui de C99. Pour aller plus loin dans les expérimentations, nous devrions essayer un nouvel algorithme qui mélange une méthode supervisée avec une autre non supervisée, et faire de nouvelles comparaisons entre les approches statistiques et linguistiques. Finalement, notre travail montre qu'avec seulement des petites améliorations, les algorithmes existants pour segmenter des textes anglais, sont adaptables pour les textes arabes.

Références

1. R. Baeza-Yates and B. Ribeiro-Neto, “Modern Information Retrieval”. Addison-Wesley, ACM Press, 1999.
2. D. Beeferman, A. Berger, and J. Lafferty, “Statistical models for text segmentation,” Machine Learning, vol. 34, pp. 177 - 210, 1999.
3. T. Brants, F. Chen, and I. Tsochantaridis, “Topic-based document segmentation with probabilistic latent semantic analysis,” presented at CIKM, McLean, Virginia, USA, 2002.
4. J. Carletta. “Assessing agreement on classification tasks: The kappa statistic”. Computational Linguistics, 22(2):249-254. 1996.
5. F. Choi, “Advances in domain independent linear text segmentation,” presented at the first conference on North American chapter of the Association for Computational Linguistics (NAACL), Seattle, Washington, 2000.
6. K. Darwish, “Building a Shallow Arabic Morphological Analyzer in One Day,” Proceedings of the workshop on Computational Approaches to Semitic Language, in the 40th Annual Meeting of the Association for the Computational Linguistics, (ACL-02), pp. 47 - 54. 2002.
7. M. A. El-Shayeb, S. R. El-Beltagy and A. Rafea, “Comparative Analysis of Different Text Segmentation Algorithms on Arabic News Stories,” Proc. IEEE International Conference on Information Reuse and Integration, pp. 441 - 446, Aug, 2007.
8. O. Ferrat, B. Grau and N. Masson, “Thematic segmentation of texts: two methods for two kinds of texts,” In Proceedings of the 36th Annual Meeting of the ACL, 1998.
9. M. Galley, K. McKeown, E. Fosler-lussier, and H. Jing. Discourse segmentation of multi-party conversation. In: Proceedings of the 41st Annual Meeting of ACL, Sapporo, Japan, 2003.
10. G. Grefenstette, and P. Tapanainen. What is a word, what is a sentence? Problems of tokenization. In: Proceedings of the 3rd Conference on Computational Lexicography and Text Research (COMPLEX-94), Budapest, Hungary, 1994.
11. B. J. Grosz and C. L. Sidner, “Attention, Intentions and the Structure of Discourse,” Computational Linguistics, vol. 12, pp. 175 - 204, 1986.
12. Hasnah, “Full Text Processing and Retrieval: Weight Ranking Text Structuring, and Passage Retrieval for Arabic Documents,” Ph.D. thesis, Illinois Institute of Technology. 1996.
13. M. A. Hearst, “TextTiling: Segmenting text into multiparagraph subtopic passages,” Computational Linguistics, vol. 23, pp. 33 - 64, 1997.

A. Isard and J. Carletta “Replicability of transaction and action coding in the map task corpus”. In Johanna Moore and Marilyn Walker, editors, Empirical Methods in Discourse: Interpretation & Generation, AAAI Technical Report SS-95~06. AAAI Press, Menlo Park, CA. 1995.

14. M. Y. Kan, J. L. Klavans, and K. R. McKeown, “Linear segmentation and segment relevance,” presented at the International Workshop of Very Large Corpora (WVLC 6), Montreal, 1999.
15. D. Kauchak and F. Chen, “Feature-based segmentation of narrative documents,” presented at the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing, Ann Arbor, MI, USA, 2005.
16. H. Kozima, “Text Segmentation Based on Similarity between Words,” In Proceedings of ACL'93, pp. 286 - 288, Ohio, Japan, 1993.
17. D. J. Litman and R. J. Passonneau. “Combining multiple knowledge sources for discourse segmentation”. In Proceedings of the 33rd Meeting of Association for Computational Linguistics., pages 108-115, June. 1993.

18. O. Manabu and H. Takeo, “Word sense disambiguation and text segmentation based on lexical cohesion,” presented at The International Conference on Computational Linguistics, Kyoto, Japan, 1994.
19. N. Masson, “An Automatic Method for Document Structuring,” In Proceedings of the 18th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, 1995.
20. G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, “Five papers on Wordnet,” Cognitive Science Laboratory, Technical report 1990.
21. J. Morris and G. Hirst, “Lexical cohesion computed by thesaurus relations as an indicator of the structure of text,” Computational Linguistics, vol. 17(1), pp. 21 - 48, 1991.
22. D.D. Palmer and M. A. Hearst, “Adaptive sentence boundary disambiguation,” In Proceedings of the 4th Conference on Applied Natural Language Processing, Stuttgart, Germany, October. 1994.
23. J. R. Passonneau and D. J. Litman. “Intention-based segmentation: Human reliability and correlation with linguistic cues”. In Proceedings of the 31st Annual Meeting, pages 148-155. 1993.
24. J. Reynar, “Topic Segmentation: Algorithms and Application,” Ph.D. thesis, Computer and Information Science. University of Pennsylvania, Pennsylvania, USA, 1998.
25. N. Stokes, J. Carthy, and A. F. Smeaton, “SeLeCT: a lexical cohesion based news story segmentation system,” AI Communications, vol. 17, pp. 3 - 12, 2004.