El Younoussi Yacine
Laboratoire Systèmes d'Information Multimédia et Mobiles (SI3M)
Ecole Nationale Supérieure de l'Informatique et Analyse des Systèmes Maroc.
Doukkali Sdigui Abdelaziz & Belahmer Habib
Laboratoire Alkhawarizmi de Génie Informatique (LAGI)
Ecole Nationale Supérieure de l'Informatique et Analyse des Systèmes, Maroc.

La racinisation de la langue arabe par les automates à états finis (AEF)

Résumé

Par sa richesse morphologique et syntaxique, la langue arabe est considérée parmi les langues les plus difficiles à traiter dans le domaine de recherche d'information. Cela est dû, notamment, aux diverses difficultés rencontrées dans sa racinisation, qui n'a pas encore connu une approche standard. Nos travaux se situent sur cet axe de recherche. On essaye de développer une nouvelle approche de racinisation de la langue arabe simple et efficace.

Abstract

Because of its morphological and syntactic richness, the arabic language is considered as one of the most difficult languages to deal within the search information domain. This is due to the difficulties met during stemming which doesn't have a standard approach. Our work is about this search area. We try to develop a simple and efficient stemming approach of the arabic language.

Mots-clés: Racinisation, la langue arabe, extraction de l'information, traitement automatique de la langue arabe, automate à état fini.
Keywords: Stemming, arabic language, information retrieval, automatic treatment of arabic language.

1. Introduction

La racinisation est le processus d'extraction des racines des mots. Une définition courte, certes, mais qui dissimule beaucoup de complexité lorsqu'il s'agit de la langue arabe.

Par ses propriétés morphologiques et syntaxiques, la langue arabe est considérée comme une langue difficile à maîtriser dans le domaine du traitement automatique des langues [1] [2]. A la différence des autres langues comme, le français ou l'anglais, dont les étiquettes grammaticales proviennent d'une approche distributionnelle caractérisée par une volonté "d'écarter toute considération relative au sens", les étiquettes de l'arabe viennent d'une approche où la sémantique côtoie le formel lié à la morphologie du mot, sans référence à la position de ce dernier dans la phrase [3]. Ce phénomène est matérialisé par la notion de schèmes et de fonctions qui occupent une place importante dans la grammaire de l'arabe.

Cette langue a suscité un sentiment de défi auprès de plusieurs chercheurs, qui ont voué leurs travaux à sa maîtrise partielle, chose qui a donné lieu à la mise au point de plusieurs approches de sa racinisation. Plus précisément, trois approches coexistent depuis maintenant plusieurs années: la construction manuelle de dictionnaire, la racinisation légère (light stemming en anglais) et l'analyse morphologique.

2. Contexte du travail

Dans le contexte de l'augmentation importante des documents numériques, notamment dans le cadre du WEB, le domaine de la recherche d'information rencontre un nouveau défi. Le problème n'est plus l'accès à l'information, mais plutôt la recherche et le filtrage des informations réellement pertinentes.

Une nouvelle solution est proposée dans cet article. Les résultats de ces travaux s'inscrivent dans le cadre d'un projet de recherche baptisé GENAUM[1]: Génération Automatique de Moteurs de Recherche, initié par l'équipe de recherche et de développement ASTROLAB du laboratoire AlKhawarizmi de l'Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes de Rabat. Ce projet consiste à développer une architecture hybride de génération de moteurs pour la recherche d'informations à travers les moteurs de recherche arabes tout en combinant les trois notions de la recherche sur le Web: les annuaires, les moteurs de recherche et les méta moteurs de recherche.

Dans le cadre de ce projet, dont le but principal est l'accroissement de la pertinence, la racinisation occupe une place très importante, tant elle intervient au niveau du traitement des requêtes, des résumés automatiques et au niveau de la catégorisation du contenu. Désormais, dans le cadre de la langue arabe, une place importante devrait être réservée à la problématique de la racinisation.

3. Etat de L'art

Comme nous l'avons déjà mentionné, trois différentes approches de racinisation de la langue arabe ont été identifiées. La construction manuelle de dictionnaire, la racinisation légère (light stemming en anglais) et l'analyse morphologique.

Les premiers travaux sur la racinisation utilisaient la construction manuelle de dictionnaires. Al-Kharashi et Evens ont travaillé avec des petites collections de texte pour lesquelles ils ont construit manuellement des dictionnaires de racines et de pseudos racines (stems) pour chaque mot [4]. Cette approche est évidemment impraticable pour les corpus de taille réaliste.

[1] GENAUM est un projet financé par Maroc Télécoms

La racinisation légère (ou light stemming en anglais) est un processus d'enlèvement de préfixes et/ou suffixes qui génère une pseudo-racine (stem en anglais), sans se préoccuper des infixes ou de reconnaître les schèmes (patterns en anglais) pour retrouver la racine [5]. Tous les algorithmes qui ont été élaborés pour cette approche, notamment par Kareem DARWICH, Aitao CHEN et Shereen KHOJA [6] [7] [8], ont le même défaut. Ils se basent sur une liste prédéterminée contenant un ensemble de préfixes et suffixes censés être enlevés au cas où ils sont retrouvés attachés au mot dont on cherche la racine (voir tableau 2). Cette liste a été construite dans la plupart des cas en se basant sur des calculs statistiques et la connaissance de la langue arabe. Le fait de se baser sur cette liste d'affixes, engendre pas mal de résultats erronés ou par fois ambigus, cela est dû, soit à l'enlèvement des lettres considérées comme affixes alors qu'elles sont originales (بعيد (loin) ➔عيد (fête) selon l'algorithme de A.CHEN), soit on n'enlève pas certains affixes parce qu'ils ne sont pas pris en compte par l'algorithme (ودخل et il est entré➔ ودخل selon l'algorithme de DARWISH). En outre, cette approche ne se soucie pas de rechercher la racine, c'est-à-dire qu'elle n'enlève pas les infixes des mots (كاتب (écrivain) ➔كاتب). Tous ces problèmes diminuent l'efficacité de cette technique de racinisation ; d'où le recours à d'autre approche, telle que l'analyse morphologique.

Préfixes	Suffixes
وال، فال، بال بت، يت، لت، مت، وت، ست، نت، بم، لم، وم، كم، فم، ال، لل، وي، لي، في، وا، فا، لا، با	ات، وا، ون، وه، ان، تي، ته، تم، كم، هم، هن، ها، ية، تك، نا، ين، يه ة، ه، ي، ا

Tableau 1: Liste des préfixes et suffixes enlevés par Karim DARWISH dans la racinisation légère

Il est souvent supposé que la racinisation légère n'est qu'un chemin rapide et grossier pour se rapprocher de l'analyse morphologique, et que la meilleure façon de faire la racinisation serait d'exécuter une analyse morphologique correcte et par la suite d'extraire les racines des mots pour rechercher de l'information.

Les mots arabes se divisent en trois catégories: noms, verbes, et particules [19]. Les noms et les verbes sont dérivés d'un ensemble restreint de racines (autour de 10.000) composées généralement de 3, 4, ou même 5 lettres [20]. Les noms et les verbes arabes sont dérivés des racines en leur appliquant des schèmes, pour produire les pseudo-racines ou stems en anglais. L'application des schèmes implique souvent l'ajout des infixes, la suppression ou le remplacement des lettres de la racine. Le tableau 2 montre quelques mots dérivés de la racine كتب (écrire) en appliquant trois schèmes différents.

Le mot	Schème correspondant
كتب (Il a écrit)	فعل
كتاب (Livre)	فعال
كاتب (écrivain)	فاعل

Tableau 2: Quelques schèmes appliqués sur la racine كتب «écrire».

Plusieurs algorithmes d'analyse morphologique ont été élaborés [11, 12, 13, 14]. De tels analyseurs tentent de retrouver la racine ou les racines possibles, comme celui de Karim DARWISH. L'analyse morphologique se fait sur deux étapes. La première, consiste à appliquer une racinisation légère. Dans la deuxième, on essaye d'appliquer la correspondance entre la pseudi-racine, engendrée au cours de la première étape, et un schème, afin d'extraire la racine.

Le défaut de cette approche, comme la précédente, (la racinisation légère) est qu'elle se base sur l'enlèvement des affixes, en se basant sur une liste prédéterminée, avant d'appliquer la notion de schème. En outre, lorsqu'il s'agit d'un mot contenant une lettre faible (ا و ي) ou double (مدّ tendre), le processus d'extraction de racines, devient de plus en plus difficile. Les lettres faibles sont des lettres qui peuvent être, soit conservées, soit remplacées ou même éliminées lors de leurs déclinaison. Le Tableau 3 donne un exemple de dérivation du mot قال (dire). [21]

Le caractère ا est remplacé par	قال	Dire
ا	قال	Dire
و	يقول	Il dit
ي	قيل	Il a été dit
Ø	قل	Dis

Tableau 3: Exemple de déclinaison du verbe irrégulier قال (dire)

4. Notre approche: Racinisation par automates à états finis

Comme nous avons pu découvrir dans la section précédente, les anciennes techniques de racinisation, notamment la racinisation légère et l'analyse morphologique ont des limites qui affaiblissent le processus d'extraction des racines. Dans cet article nous proposons une autre approche de racinisation basée sur les automates à états finis (AEF).

4.1. Automate à états finis (AEF)

Un automate à états finis **(AEF)** est défini par:

- un ensemble fini E d'états
- un état e_0 distingué comme étant l'état initial
- un ensemble fini T d'états distingués comme états finaux (ou états terminaux)

- un alphabet ∑ des symboles d'entrée
- une fonction de transition, Δ, qui à tout couple formé d'un état et d'un symbole de Σ fait correspondre un ensemble (éventuellement vide) d'états: $\Delta (e_i, a) = \{ei1, e_{i2}, \ldots e_{in}\}$

4.2. Racinisation avec les AEF

Les schèmes de la langue arabe, sont une sorte de gabarit appliqué sur une racine pour en extraire une dérivée. Alors, si on veut extraire la racine à partir de la dérivée, il nous suffit de chercher le schème correspondant à notre mot, et ensuite, éliminer toutes les lettres additives. Les approches précédentes éliminent les préfixes et les suffixes, éventuellement ajoutés aux mots, en appliquant une racinisation légère, ce qui entraîne parfois des erreurs et des ambiguïtés.

a. Prétraitement:

Le prétraitement est l'étape qui précède la racinisation, elle consiste à faire une série de modifications sur la requête utilisateur, c'est une sorte de standardisation. En se référant aux travaux antérieurs, nous avons adopté le prétraitement suivant :

- Enlever la ponctuation
- Enlever le signe diacritique «chadda» (ّ)
- Enlever les mots vides (stopwods en anglais), qui ne portent aucune information, par exemple les pronoms et les prépositions (التي, هنا, أين, كيف, لكنكم, هاتان)
- Appliquer une normalisation légère:
- Remplacer إ, آ, et أ au début d'un mot par ا
- Remplacer ى à la fin d'un mot par ي
- Remplacer ة à la fin d'un mot par ه

b. Traitement:

Avec notre technique de racinisation, les schèmes sont représentés par des automates à états finis, et l'enlèvement des affixes se fait automatiquement et sans recours à une liste prédéterminée.

Nous avons conçu 5 automates (figures 1, 2, 3,4 et 5) qui représentent presque tous les schèmes de la langue arabe, et cela revient à :

- Au cours de notre recherche et selon notre connaissance de la langue arabe, nous avons constaté qu'à partir d'un seul mot nous pouvons déduire une ou plusieurs racines possibles
- Pour diminuer le taux d'erreur et d'ambiguïté

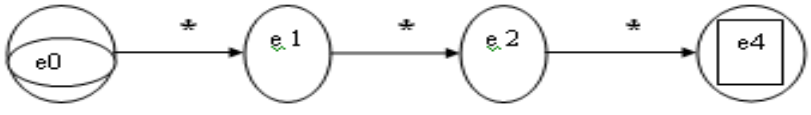

Figure 1: Automate à états finis représentant le schème فعل

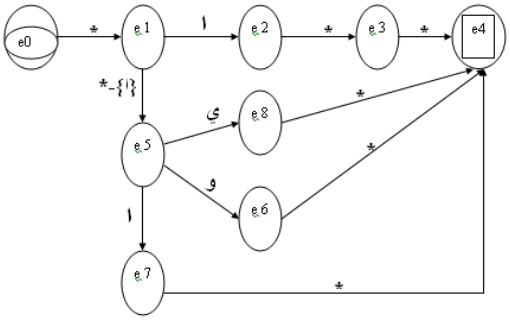

Figure 2: Automate à états finis représentant les schèmes فعيل, فعال, فعول, فاعل
Remarque: Pour passer de l'état e1 à l'état e5, il faut que la lettre en question soit différente de la lettre ا (*-{ا}).

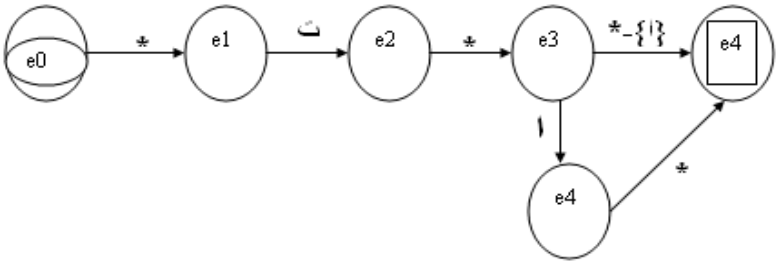

Figure 3: Automate à états finis représentant les schèmes فتعل, فتعال

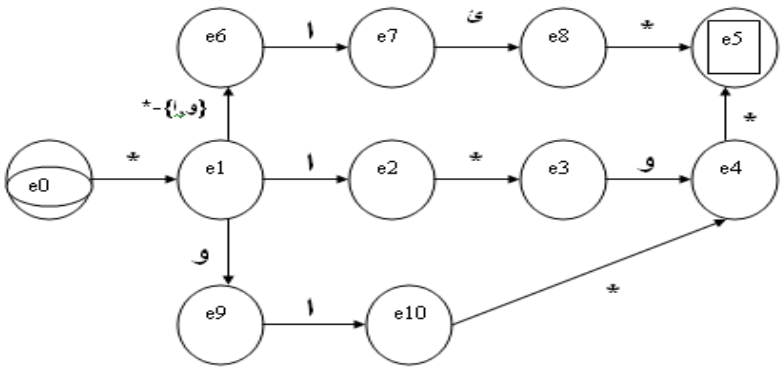

Figure 4: Automate à états finis représentant les schèmes فعائل, فواعل, فاعول

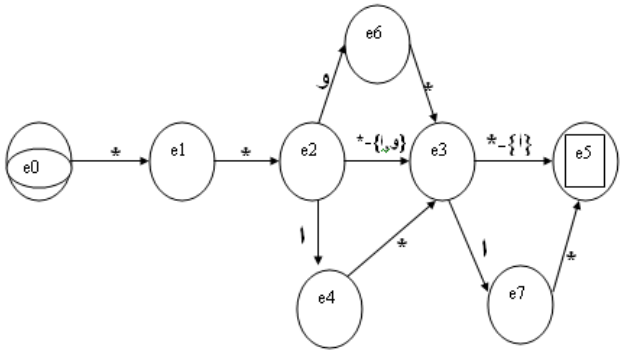

Figure 5: Automate à états finis représentant les schèmes فعلل, فعالل, فعلال, فعوعل

Après avoir appliqué le prétraitement au mot dont on cherche la racine, on le fait entrer dans les cinq automates. Dans chaque automate, si le mot parvient à trouver un chemin pour atteindre l'état final, on récupère la racine et on vérifie sa validité en consultant la liste des racines valides créée par Kareem DARWISH, contenant environ 10.000 racines. Si la racine trouvée n'est pas valide ou que l'on n'arrive pas à atteindre l'état final, la première lettre sera considérée comme préfixe et on recommence, une fois la première lettre enlevée.

Si on prend par exemple l'automate représentant le schème فاعل (Figure 6), il se compose de 5 états, e0, e1, e2, e3 et e4, avec e0 est l'état initial et e4 est l'état final. $\sum$ ={*,ا} avec, * peut représenter n'importe quelle lettre. Δ(e0,*)=e1, Δ(e1,ا)=e2, Δ(e2,*)=e3, Δ(e3,*)=e4. Voici donc la représentation graphique de cet automate :

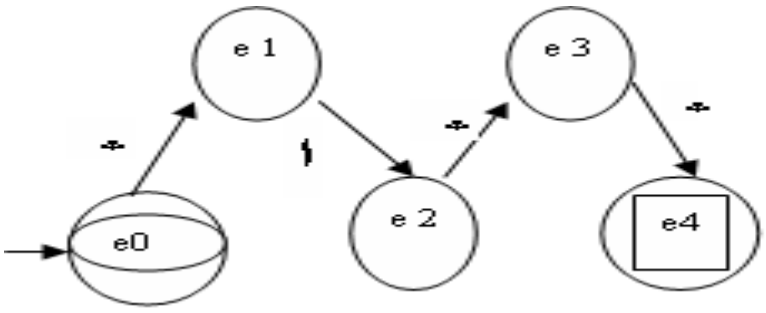

Figure 6: Automate à état fini correspondant au schème فاعل

4.3. Exemples

الكاتبون (les écrivains) : il ne va pas passer dans l'automate du schème فاعل, alors on enlève sa première lettre أ, ce qui donne لكاتبون. On rentre le nouveau mot dans

l'automate. Une fois encore le mot ne parvient pas à atteindre l'état final. On enlève la lettre ل, et on recommence. Maintenant il l'atteint, avec la lettre ب, et le suffixe ون, ne sera pas pris en considération, il s'élimine automatiquement. La racine retournée est **كتب**.
فقراء (des pauvres) est dérivé à partir du schème فعل en ajoutant le suffixe اء, parmi les racines retournée la racine **فقر**
والحوادث (et les accidents) est dérivé à partir du schème فواعل en ajoutant le préfixe وال, **حدث** est l'une des racines retournées.
وسيانتقلان est dérivé à partir du schème افتعل en ajoutant le préfixe وسي et le suffixe ان, l'automate convenable retourne la racine **نقل**

4.4. Cas d'échec

La richesse et la complexité de la langue arabe compliquent de plus en plus l'opération de la recherche des racines. En fait, selon le contexte, le même mot peut être dérivé parfois de plus qu'une racine. Par exemple, le tableau 4 montre cinq différentes racines à partir desquelles le mot إيمان a pu être dérivé [10].

Notre approche de racinisation prend en considération ce que nous venons de mentionner ci-dessus, puisque le mot dont on cherche la racine rentre dans les cinq automates, ce qui veut dire qu'on aura au plus cinq racines possibles pour le même mot. Le problème est que dans la plupart des cas notre approche retourne plus qu'une racine, alors qu'une seule est suffisante, et le reste est pratiquement hors contexte.

Racine	Signification
أمن	Sécurité
أيم	Deux personnes pauvres
مأن	Est-ce qu'il va nous donnez un support
يمن	Convention
يمأ	Est-ce qu'elles vont pointer à

Tableau 4: Les racines possibles du mot ايمان

Notre algorithme de racinisation est l'un des algorithmes qui rencontrent des problèmes lorsqu'il s'agit des mots contenant une ou plusieurs lettres faibles (ا و ي).

5. Conclusion et perspectives

Nous avons essayé, à travers cet article, de présenter brièvement les différentes approches de racinisation de la langue arabe ainsi que la notre. La technique que nous proposons se base sur les automates à états finis dans le processus d'extraction de racines, dans le but de minimiser le taux d'erreur et d'ambiguïté, dû généralement à l'enlèvement des affixes. Nous nous concentrons actuellement sur le

développement et l'amélioration de notre technique de racinisation tout en essayant de surmonter les différents problèmes rencontrés. Nous travaillons en parallèle sur la compilation d'un corpus d'évaluation qui nous permettra d'évaluer et de comparer en même temps notre approche par rapports aux autres.

Références

[19]. Abdul-Al-Aal, A., An-Nahw Ashamil: 1987, Cairo, Egypt: Maktabat Annahda Al- Masriya.

[13]. Al-Fedaghi, S.S. and Al-Anzi, F.S.:1989, A new algorithm to generate Arabic root-pattern forms

[8]. Aljlayl, M., Beitzel, S., Jensen, E., Chowdhury, A., Holmes,D., Lee, M., Grossman, D. et Frieder**,** O.: 2001 IIT at *TREC-10.In TREC 2001*. Gaithersburg: NIST.

[1]. Aljlayl, M. and Frieder, O.: 2002, On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach.

[4]. Al-Kharashi, I. et Evens.M.: 1994, Comparing words, stems, and roots as index terms in an Arabic information retrieval system.

[14]. Beesley, K.R.: 1996 Arabic finite-state morphological analysis and generation. In *COLING-96*.

[15]. Brent, M.R.: 1999, Speech segmentation and word discovery: A computational perspective.

[7]. Chen, A. and Gey, F.: 2002, Building an Arabic Stemmer for Information Retrieval.

[12]. Darwish, K., Doermann, D., Jones, R., Oard, D. and Rautiainen, M.: 2001 *TREC-10* experiments at Maryland: CLIR and video. In TREC 2001. Gaithersburg: NIST.

[10] Darwish, Kareem: 2003, Probabilistic Methods for Searching OCR-Degraded Arabic Text.

[6]. Darwish, Kareem: 2003, Building a Shallow Arabic Morphological Analyzer in One Day.

[3]. Débili, F., Achour, H., Souici, E.**:** 2002, La langue arabe et l'ordinateur : de l'étiquetage grammatical à la voyellation automatique.

[21]. Fouad, Douzidia, Soufiane: 2004, Rés Comme nous avons pu découvrir dans la section précédente, les anciennes techniques de racinisation, notamment la racinisation légère et l'analyse morphologique ont des limites qui affaiblissent le processus d'extraction des racines. Dans cet article nous proposons une autre approche de racinisation basée sur les automates à états finis (AEF).

[17]. Goldsmith, J.: 2000, Unsupervised learning of the morphology of a natural language.

[18]. Goldsmith, J., Higgins, D. and Soglasnova, S.: 2001, Automatic language-specific stemming in information retrieval.

[11]. Khoja, S. and Garside, R.: 1999, Stemming Arabic text. Computing Department, Lancaster University, Lancaster,.

[2]. Larkey, L.S., Ballesteros, L. and Connell, M.: 2002, Improving Stemming for Arabic Information Retrieval: Light Stemming and Cooccurrence Analysis.

[5]. Larkey, Leah., Ballesteros, Lisa. et Margaret, E.: *CONNELL,* Light stemming for Arabic information retrieval.

[20]. Manzour, Ibn Lisan, Al-Arab.

[16]. Marcken, C.: 1995, Unsupervised language acquisition. PhD thesis, MIT, Cambridge.

[9].Roeck, A.N. and Al-Fares, W.: 2000, A morphologically sensitive clustering algorithm for identifying Arabic roots. *In Proceedings ACL-2000*. Hong Kong,.