Construction d'un corpus arabe à partir du Web dans le but d'identifier les mots-outils ou *tokens*

Dhaou Ghoul[1]

[1] STIH – dhaou.ghoul@gmail.com

Abstract

In this paper, we present a method to build a large corpus for the Arabic from the Web. Our goal is to have at hand a good resource that allows us to identify the different forms of speech of the Arabic language and specific word tools or tokens. To do this, first we have implemented a Perl script that can extract html files from a URL. Then, we cleaned these files for raw text. The detection and identification of tokens is done through a tool called kawâkib.

Résumé

Dans cet article, nous présentons une méthode de construction d'un vaste corpus pour l'Arabe à partir du Web. Notre objectif est d'avoir sous la main une grande ressource qui nous permet d'identifier les différentes formes du discours de la langue arabe et spécifiquement les mots outils ou *tokens*[1]. Pour ce faire, tout d'abord nous avons implémenté un script Perl qui permet d'extraire à partir d'une adresse URL des fichiers html. Ensuite, nous avons nettoyé ces fichiers pour obtenir des textes bruts. La détection et l'identification des *tokens* se fait grâce à un outil appelé *kawâkib*.

Mots-clés : TAL, corpus, grammaire, *token*, *script*, Web, langue arabe

1. Introduction

Malgré les différents travaux effectués dans le domaine du traitement automatique de la langue arabe et bien qu'elle soit parlée par presque 280 millions de personnes, il reste toujours compliqué de trouver assez de ressources gratuites à propos de cette langue. Dans le cadre de notre projet de thèse (Mogador[2]), nous avons décidé de créer notre propre corpus à partir du web dans le but de localiser les différents tokens en arabe littéraire afin de créer une grammaire pour chaque *token*. Avec le développement de l'internet et de ses services, le web est devenu une grande source de documents dans différentes langues et différents domaines. Cette source alliée à des supports de stockage permet la construction rapide de corpus (Meftouh et al, 2007).

Notre projet de thèse consistera à étudier et classifier les tokens de la langue arabe afin de les modéliser par des automates. La finalité étant de proposer pour chaque *token* une grammaire qui décrit les attentes syntaxiques de ce dernier.

Dans ce papier, nous présentons la méthode pour construire un vaste corpus de langue arabe à partir du web dans le but d'identifier les différentes formes du discours (nom, verbe, mots-outils). Notons bien que dans notre travail, nous nous intéressons à l'identification des mots-outils.

[1] Les mots qui n'appartiennent pas au lexique arabe et n'obéissent pas à la dérivation morphologique de l'arabe.

[2] http://halshs.archives-ouvertes.fr/docs/00/91/20/09/PDF/Mogador_Jaccarini_Gaubert.pdf

Cet article est organisé comme suit : la section 2 présente notre corpus ainsi que la méthode de sa construction et la section 3 présente l'outil *kawakib* qui nous permet de détecter et identifier les *tokens*.

2. Construction du corpus

L'utilisation du Web comme base pour la constitution de ressources textuelles est très récente. Ces dernières années ont ôté de travaux tentant d'exploiter ce type de données. Dans une perspective de traduction automatique (Resnik et al., 1998) étudient la possibilité d'utiliser les sites Internet proposant les informations en plusieurs langues pour constituer des corpus parallèles bilingues.

(Ghani et al., 2001) exposent l'idée de construction de corpus, à partir du web, par interrogation automatique de moteurs de recherche. Ils exploitent cette idée pour la constitution de corpus de langues minoritaires.

Dans une tout autre approche (Issac et al., 2001) mettent au point un logiciel pour la constitution d'un corpus de phrases dans le but d'étudier le comportement de l'une des formes de discours (non, verbe, *token*), des noms prédicatifs marquant la localisation et le déplacement, afin de mesurer si l'introduction des prépositions dans les requêtes en recherche d'information permet d'améliorer la précision.

Pour obtenir un vocabulaire suffisant qui permette de nous donner une idée précise sur les critères linguistiques (leurs propriétés grammaticales) des formes de discours et en particulier les mots-outils et étant donné le manque de ressources arabes en accès libre, nous avons essayé de réaliser notre propre corpus. La réalisation de notre corpus est portée sur l'extraction des différents articles à partir d'un site web. Le site que nous avons choisi est le site du journal électronique «Alwatan»[3] (الوطن) en 2004. La question qui se pose ici est : comment peut-on avoir un corpus directement utilisable par des linguistes à partir d'une adresse web de type URL ?

La procédure de constitution de notre corpus est divisée en trois grandes étapes :

- Etape1 : le but de cette étape est de construire la liste des noms de fichiers html disponibles sur le site du journal de la presse (الوطن). Pour ce faire, nous avons implémenté un script Perl qui prend en entrée l'adresse web visée et génère en sortie l'ensemble des fichiers html regroupés par domaine.
- Etape2 : cette étape nous permet d'extraire les textes bruts. Pour l'atteindre, comme à la première étape, nous avons implémenté un script Perl qui permet de nettoyer les fichiers html c'est-à-dire d'enlever toutes les balises du fichier html de façon à ne garder que le texte.
- Etape3 : l'objectif ici est de nettoyer le texte obtenu à la deuxième étape. Pour ce faire, nous avons implémenté un script perl en basant sur l'outil de nettoyage de corpus «Autoveille corpus[4]». Cette phase consiste à effacer tout ce que n'est pas utile pour l'analyse syntaxique (les espaces doubles, la répétition, les caractères inconnus…). Notons que nous avons translitéré les textes sous la forme de Buckwalter[5] (Buckwalter,

[3] http://www.elwatannews.com/

[4] http://autoveille.free.fr/constitution-automatique-corpus.html

[5] http://www.qamus.org/transliteration.htm

2002) grâce à un script que nous avons implémenté dans mon travail de mémoire du master (Ghoul, 2013) pour éviter les problèmes de codage arabe et pour faciliter le nettoyage automatique de notre corpus.

Le schéma suivant présente les différentes étapes de réalisation de notre corpus :

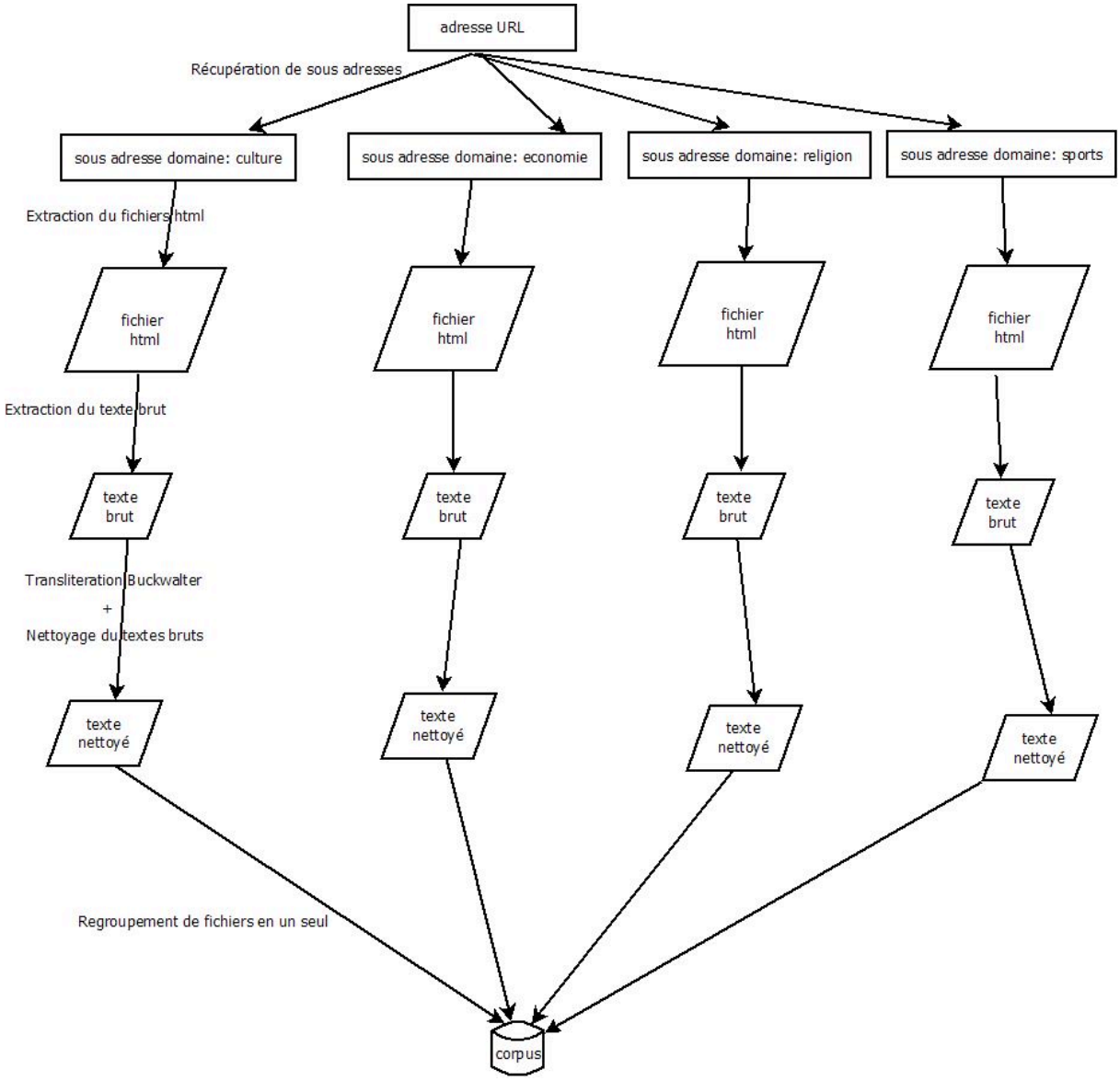

Figure 1. Les étapes de la construction de notre corpus

Notre corpus est constitué de 207 356 phrases et 7 653 881 mots (dont 466 623 mots différents) distribués sur quatre domaines : culture, économie, religion et sports, de la manière suivante :

Domaine	Nb articles	Nb phrases	Nb mots / Nb mots différents	Nb tokens
Culture	12	52 984	1 416 583 / 163 456	326 180
Economie	13	50 715	1 605 236 / 122 270	317 374
Religion	12	55 372	3 159 306 / 105 045	762 314
Sports	12	48 285	1 472 756 / 145 839	301 265

Tableau 1. Statistiques de notre corpus

3. Identification des tokens

L'idée de notre recherche est de traiter la langue arabe en nous basant sur des règles minimales qui peuvent être complexes et non sur un lexique en espérant trouver de bons résultats. Notre objectif, est de réaliser pour chaque *token* une grammaire qui décrit leurs différentes attentes syntaxiques. Dans leurs travaux Claude Audebert et André Jaccarini, ont proposé des grammaires qui définissent quelques opérateurs syntaxiques dans la langue arabe (inna, anna, an, …). En nous basant sur cette bibliographie nous allons essayer d'améliorer la base grammaticale de ces opérateurs. Selon (Dichy et Zamantar, 2009) : « La reconnaissance des mots-outils en particulier, se heurte à un certain nombre d'ambiguïtés dues aux risques de confusion entre mots-outils et verbes d'une part et entre mots-outils et noms d'autre part. ». L'identification et la détection des tokens se fait à partir du logiciel « *kawâkib* » réalisé au sein des équipes de l'IFAO et de la MMSH[6] par (Gaubert, 2010).

Kawâkib est une application Web conçue à partir des éléments applicatifs produits par le logiciel Sarfiyya (Gaubert et al., 2009) dont le but est d'analyser les différentes parties du discours de la langue arabe (verbe, nom et tokens). Les fonctions et les grammaires sont développées puis compilées pour certaines en mode déterministe. Cette application permet aux utilisateurs d'exécuter plusieurs fonctions (racines triées par fréquence, *tokens*, répétitions, citations, expressions régulières…)[7]. Dans notre recherche, nous sommes intéressés uniquement par les mots-outils (*tokens*). La figure suivante présente un extrait de détection des *tokens* à partir de notre corpus grâce à cette application :

[6] http://www.mmsh.univ-aix.fr/Pages/default.aspx

[7] Pour avoir plus de détails sur ces fonctions vous pouvez consulter la version publique : http://www.ifao.egnet.net/kawakib.

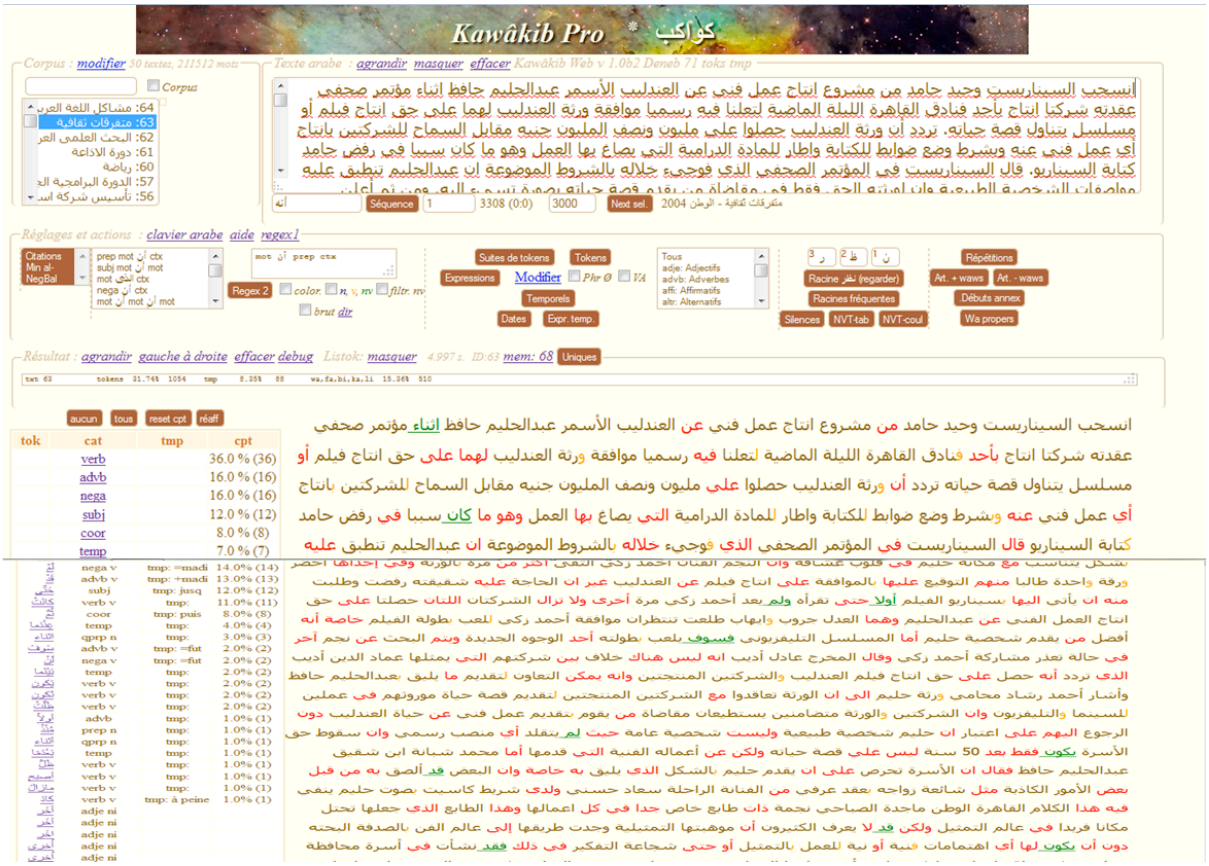

Figure 2. Exemple d'identification de tokens

D'après la figure précédente, on remarque qu'en plus des différentes fonctions intégrées dans l'application *kawâkib*, ce dernier présente une banque de ressources permettant à l'utilisateur d'ajouter son propre corpus. Les *tokens* et mots assimilés repérés sont mis en évidence par un jeu de couleur : rouge pour les *tokens* (vert pour les *tokens* temporels), orange pour les prépositions (bi,li,ka) et coordonnants proclitiques (wa,fa) qui sont repérés seulement par leur position et non par analyse morphologique. Notons que « *kawâkib* » n'accepte en ce moment que des textes écrits en arabe.

4. Conclusion et perspectives

Le but de ce travail est de construire notre propre corpus puis dans un deuxième temps d'analyser les mots outils de la langue arabe afin d'avoir pour chaque mot une fiche signalétique qui décrit d'une façon très détaillée ce dernier. Pour ce faire, nous avons décidé d'utiliser le Web comme source. Dans ce papier nous présentons notre corpus comme un moyen qui nous permet d'analyser les différents *tokens* dans plusieurs contextes différents et non pour l'utiliser dans d'autres applications de traitement automatique de la langue arabe (étiquetage morphosyntaxique, traduction automatique, extraction des entités nommées…). En effet, comme perspectives dans ce travail, il nous reste à étiqueter ce vaste corpus grâce à des linguistes et le rendre public et exploitable pour améliorer le traitement automatique de l'arabe.

Références

Audebert C., Gaubert C. et Jaccarini A. (2009). Stratégies et règles minimales. *Medar. 2nd International Conference on Arabic Language Resources and Tools,* Caire, Egypt, 22-23 Avril 2009.

Buckwalter T. (2002). Arabic Morphological Analyser version 1.0. *Linguistic Data Consortium Catalogue numéro LDC L 49.*

Gaubert C. (2010). Kawâkib, une application pour le traitement automatique de textes arabes. *Vol. Annales islamologiques,* 44, IFAO, Caire,Egypt , http://www.ifao.egnet.net, pp. 53-59.

Ghani R. et Jones D. (2001). Mladenic, mining the web to create minority language corpora. *CIKM* 2001, 279-286.

Ghoul D. (2013). "Développement de ressources pour l'entrainement et l'utilisation de l'étiqueteur morphosyntaxique TreeTagger sur l'arabe". *RECITAL '13. Conférence TALN- Recital,* Sables d'olonnes France, 17-21 Juin 2013.

Dichy J. et Zmantar Y. (2009). L'analyse automatique des mots-outils en arabe. *Actes de la 2ème conférence internationale sur les systèmes d'Informations et l'Intelligence Economique (SIIE'2009)*, 12,13 et 14 février 2009, Hammamet, Tunisie. http://www.siie.fr, pp. 586-597.

Meftouh K., Smaïli K. et Laskri M.T. (2007). Constitution d'un corpus de la langue arabe à partir du Web. *CITALA '07. Colloque international du traitement automatique de la langue arabe*. Iera, Rabat, Morocco, 17-18 juin 2007.

Issac F., Hamon T., Bouchard L., Emirkanian L. et Fouqueré C. (2001). Extraction informatique de données sur le web : une expérience, in Multimédia, Internet et francophonie : à la recherche d'un dialogue, Vancouver, Canada, mars 2001.

Resnik P. (1998). Parallel strands: A preliminary investigation into mining the web for bilingual text. *In conference of the association for machine translation in the* Americas, 1998.