République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Djilali BOUNAÂMA Khemis Miliana

Faculté des Sciences et de la Technologie

Département de Mathématiques et d'Informatique

Mémoire Présenté

Pour l'obtention de diplôme de

Master en Informatique

Spécialité : Ingénierie du Logiciel

Titre :

Identification automatique de mots clés dans les textes arabes

Réalisé par : Boubekeur Yassamina

Soutenu publiquement le : 26/05/2016

devant le jury composé de:

Mr. L.Mohamed Bouziane	.Président
Mr Bahloul Djamel	.Encadreur
Mme. A.Lababou	.Examinateur1
Mme. F.Boudali	.Examinateur2

Année Universitaire 2015/2016

Remerciements

Je rends grâce à dieu de m'avoir donné la force, le courage et la patience d'accomplir ce présent travail.

Je tiens à remercier vivement, notre cher professeur et promoteur Mr BAHLOUL DJAMEL pour ces orientations et sa patience, qu'il trouve ici ma reconnaissance et ma gratitude.

Je remercie également tous les enseignants qui ont contribué à notre formation.

Enfin, je remercîments vont à mes amies et collègues de la promotion 2016.

Dédicaces

Je dédie ce travail de fin d'étude en informatique à toute âme charitable et bienveillance qui n'a pas manqué de m'apporter son aide et son soutien à un moment ou à un autre pour m'assister et me prêter main forte dans les moments difficiles de mon cursus universitaire.

Et en particulier à mes chers parents ainsi qu'à mes frères et sœurs et surtout à mon frère le défunt MOHAMED.

Je dédie aussi ce modeste travail à tous les membres de ma grande famille sans oublier bien sûr mes amies.

Je remercie toute personne ayant participé à travers cette modeste contribution à vos justes commentaires et précieuses observations.

À tous mes collègues et amies de la promotion 2015/2016.

Merci ...

BOUBEKEUR YASSAMINA

Table des matières

Liste des tableaux

Liste des figures

Résumé

A travers ce projet, nous présentons une approche hybride pour la détection automatique des mots clés dans les textes écrits en langue arabe. La méthode que nous proposons combine des critères statistiques tels que la fréquence et la position des termes avec d'autres linguistiques liés à la catégorie grammaticale.

L'expérience menée sur un corpus formé d'une collection d'articles économiques a montré l'intérêt de combiner plusieurs critères pour améliorer le processus d'identification des mots clés dans les textes arabes et en particulier l'intérêt de la catégorie grammaticale dans tel processus.

Mots clés

Traitement automatique de la langue arabe, Analyse morpho-syntaxique, Mots pertinents, Pondération de termes, Catégorie grammaticale.

ملخص

في إطار هذا المشروع قمنا باقتراح طريقة هجينة للتحديد الآلي للكلمات المفتاحية في النصوص المكتوبة باللغة العربية. الطريقة المقترحة تجمع بين المعايير الإحصائية مثل تردد ومواضع الكلمات مع أخرى لغوية تتعلق بالفئة النحوية لهذه الأخيرة.

من خلال التجربة التي قمنا بها على مجموعة من المقالات الاقتصادية تظهر لنا جليا فائدة الجمع بين عدة معايير لتحسين عملية تحديد الكلمات المفتاحية في النصوص العربية وبصفة أخص أهمية اعتبار الفئات النحوية في هذه العملية.

.

الكلمات المفتاحية :

المعالجة الآلية للغة العربية، التحليل الصرفي النحوي، الكلمات ذات الصلة، ترجيح الكلمات، الفئة النحوية.

Abstract

Through this project, we present a hybrid approach for the automatic detection of keywords in texts written in Arabic. The proposed method combines statistical criteria such as frequency and position of words with linguistic criteria related to the grammatical category.

The experiment conducted on a corpus formed of a collection of economic articles showed interest to combine several criteria to improve the process of identifying key words in Arabic texts and in particular the interest of the grammatical category in such process.

Keywords

Arabic language processing, Morpho-syntactic analysis, Relevant words, terms weighting, Grammatical category.

Introduction générale

Notre projet s'inscrit dans le cadre des travaux relatifs au traitement automatique de la langue arabe. L'objet est de réaliser un système d'identification automatique de mots clés dans les textes arabes en se basant sur la mesure d'importance des mots dans les textes. L'importance des mots sera déterminée en appliquant une combinaison de critères statistiques distributionnels (fréquence, position) et d'autres linguistiques grammaticaux (types des termes).

En effet, l'identification de mots clés est une tâche très importante pour les systèmes d'aide à la lecture, le résumé automatique et la traduction automatique…etc.

La mise en place d'un tel système nécessite une série de traitements automatiques comme la segmentation, la normalisation, la lemmatisation et la détection des catégories grammaticales d'où l'intérêt de l'utilisation d'un analyseur morphologique puissant, nous avons choisi pour cette fin l'analyseur AlKhalil.

- **Problématique**

Ces dernières années sont marquées par une augmentation énorme de la quantité d'information électronique arabe et dont l'accès à des informations pertinentes est devenu de plus en plus complexe et le besoin de développer des applications d'aide à la lecture est devenu incontournable.

La réalisation d'un système d'identification de mots clés constitue un domaine à part entière se trouvant à la croisée de T.A.L et de R.I.

Les travaux dans ce domaine existent déjà pour d'autres langues comme l'anglais ou le français, malheureusement pour l'arabe les travaux sont rares et les choses ne font que commencer. Nous essayons donc de contribuer dans ce sens en proposant une méthode de détection de mots clés pour les textes arabes.

- **Plan**

Notre travail est organisé en cinq chapitres : le premier concerne les concepts terminologiques du domaine de T.A.L. Le deuxième discute particulièrement le T.A.L arabe et les difficultés rencontrées. Dans le troisième chapitre, nous explorons les différentes méthodes et techniques déjà utilisées pour extraire les mots clés. Le quatrième chapitre est dédié à la description de l'approche proposée ainsi que l'architecture de notre système. Finalement, un cinquième chapitre expliquant l'implémentation du système, le test et l'évaluation.

Chapitre 1

Traitement Automatique De La Langue (TAL)

CHAPITRE 1
Traitement automatique de la langue

1. Introduction

On regroupe sous le vocable de traitement automatique du langage naturel (TALN) l'ensemble des recherches et développements visant à modéliser et reproduire, à l'aide de machines. Les techniques du traitement automatique des langues permettent d'extraire des textes des informations plus riches que de simples unités lexicales. Ces informations de nature morphologique, syntaxique et sémantique ont été partiellement utilisées en recherche information RI pour améliorer les méthodes d'appariement, les représentations des contenus des documents et requêtes et le processus de recherche. Ce chapitre établit un tour d'horizon sur l'impact de ces différentes informations linguistiques issues par des techniques du traitement automatique de la langue (TAL).
Une des principales questions à considérer lors de l'élaboration de tout traitement du langage naturel est le choix de l'outil le plus approprié. De nombreux outils de développements intéressants existent déjà : analyse morphologique pour définir la structure des mots, stemmer pour regrouper les mots qui ont la même racine ou stem, analyse sémantique qui permet de déterminer le sens des mots, etc.

2. Traitement automatique de la langue (TAL)

Le traitement automatiques des langues naturelle ou de la langue est un domaine de recherche qui se positionne à l'intersection de plusieurs disciplines : Intelligence artificielle, Informatique théorique, calcul statistique, linguistique, …etc. Dont le principal objectif est la conception et le développement de programmes capables de traiter de manière automatique des données linguistiques c'est-à-dire des données exprimées dans une langue dite naturelle.

Ces dernières décennies le traitement automatique des langues a connu une véritable ascension que ce soit sur le plan scientifique mais aussi socio-économique est-ce là par l'émergence de plusieurs firmes et de produits spécialisés, on parle aujourd'hui : de Traduction automatique, de correction automatique d'orthographe, de résumé automatique, d'interrogation de base de données en langues naturelle, ….etc. [1], [2].

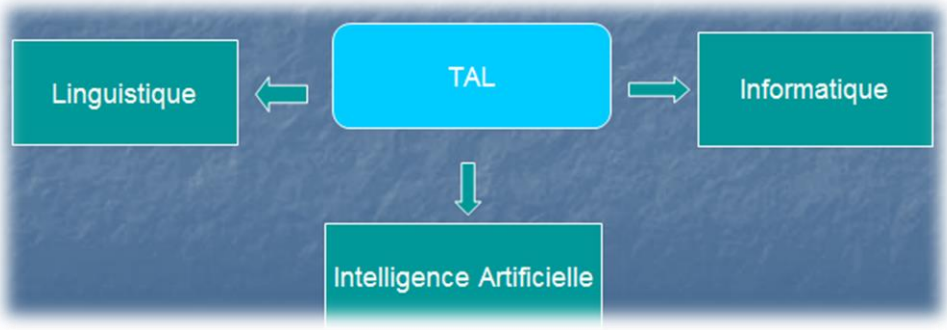


Figure 01 : Présente La Pluridisciplinaire De TAL

La réalisation de n'importe quelles applications parmi celles citées précédemment passe principalement par différent niveaux (lexicale, morphologique, syntaxique, sémantique et pragmatique) mais aussi par le développement de plusieurs modules important, où la réussite de l'application dépend pleinement de la performance de ces modules.

3. Applications TALN et des travaux existes

Concernant les applications, la demande de TALN provient, pour dire vite, de deux tendances « lourdes » : d'une part la nécessité de concevoir des interfaces de plus en plus ergonomiques, d'autre part la nécessité de pouvoir traiter (produire, lire, rechercher, classer, analyser, traduire) de manière de plus en plus « intelligente » les informations disponibles sous forme textuelle, de manière à pouvoir résister à leur prolifération exponentielle. Les applications des techniques de TAL sont donc nombreuses et variées. On a regroupé ces applications en grandes familles, qui correspondent aux aides à la lecture de documents, aux aides à la production de documents, et enfin aux interfaces homme-machines. Comme ce suit :

- Traduction automatique : ce problème longtemps sous-estimé s'est en fait avéré l'un des plus délicats à effectuer pour un ordinateur. Aux phases lexicales et syntaxiques, à peu près maîtrisées, s'ajoutent une analyse sémantique, puis pragmatique, qui tentent de déterminer le sens particulier d'un mot, dans le contexte où il apparaît. Le contexte lui-même pouvant s'étendre à l'ensemble du texte traduit[1].

Figure 02 : un exemple sur les traducteurs automatique

- Le classement ou l'indexation automatique de documents ; (eg : le SpamBuster).
- Correction automatique ;
- Recherche de l'information et Fouille de texte ;
- Résumé automatique de texte : résumer un texte signifie identifier le contexte et pondérer les parties significatives des autres.
- Synthèse de la parole ;
- Reconnaissance vocale ;
- Le traitement documentaire ;
- La production de documents : Le TALN trouve également des applications directes dans le domaine de l'aide à la production de documents, telles que :
  - les claviers « auto-correcteurs » (par exemple pour les handicapés) ;
  - les correcteurs d'orthographe ou de syntaxe (voir le "Réaccentueur")

1 : http://www.systransoft.com .

- les correcteurs « stylistiques », ou les aides intelligentes à la rédaction (thésaurus, etc.) ;
- la génération automatique de documents à partir de spécifications formelles (par exemple les documentations techniques) ;
- la reconnaissance optique de caractères ;
- l'apprentissage assisté par ordinateur des langues naturelles ;

❖ …etc.

4. Histoire du traitement automatique de la langue naturel

Historiquement, les premiers travaux importants dans le domaine du TALN ont porté sur la traduction automatique, avec, dès 1954, la mise au point du premier traducteur automatique (très rudimentaire). Quelques phrases russes, sélectionnées à l'avance, furent traduites automatiquement en anglais. Bien que le vocabulaire ne comptât que 250 mots et la grammaire 6 règles, cette expérience a déclenché de nombreux travaux dans ce domaine. Depuis 1954, de lourds financements ont été investis et nombre de recherches ont été lancées, avec un optimisme que l'on peut considérer aujourd'hui comme exagéré. Les principaux travaux présentés concernent alors la fabrication et la manipulation de dictionnaires électroniques, car les techniques de traduction consistent essentiellement à traduire mot à mot, avec ensuite un éventuel réarrangement de l'ordre des mots.

Le problème fondamental de la représentation des connaissances et de leur utilisation est donc posé, après moins de dix ans de recherches sur la traduction automatique. Ce problème est alors considéré comme insoluble. Un groupe d'experts (l'Automatic Language Processing Advisory Council : ALPAC[2]) rédige alors un rapport dans lequel il apparaît que la traduction automatique, en l'état des connaissances de l'époque, coûte environ deux fois plus cher que la traduction humaine et donne des résultats nettement moins bons. Cette considération purement économique amène un arrêt de la plus grande part des financements publics aux Etats-Unis puis en Europe.

Zellig Harris publie ses travaux les plus importants de linguistique (linguistique distributionnaliste) entre 1951 et 1954.

2 : ALPAC : un comité de sept scientifiques dirigée par John R. Pierce, créé en 1964 par le gouvernement U.S afin d'évaluer les progrès en linguistique informatique en traduction générale et de la machine, en particulier.

Il est suivi par N. Chomsky, qui publie en 1957 ses premiers travaux importants sur la syntaxe des langues naturelles, et sur les relations entre grammaires formelles et grammaires naturelles. Très schématiquement, la démarche de Chomsky est axée sur la volonté de formuler, à travers l'étude du langage, des hypothèses intéressantes sur la cognition. Le langage est une faculté à la fois universelle (tous les humains développent spontanément, pour peu que l'environnement s'y prête, un langage), et spécifique à l'espèce humaine (aucune espèce animal ne possède de système de communication comparable, dans sa richesse et dans sa complexité, au langage humain). En conséquence, la mise à jour des propriétés que possèdent tous les langages humains est aussi un moyen de mettre en évidence certaines propriétés de l'appareillage cognitif universellement utilisé pour traiter le langage (la grammaire universelle).On peut également situer en 1956, à l'école d'été de Dartmouth, la naissance de l'intelligence artificielle.

Posant comme conjecture que tout aspect de l'intelligence humaine peut être décrit de façon suffisamment précise pour qu'une machine le simule, les figures les plus marquantes de l'époque (John Mc Carthy, Marvin Minsky, Allan Newell, Herbert Simon) y discutent des possibilités de créer des programmes d'ordinateurs qui se comportent intelligemment, et en particulier qui soient capables d'utiliser le langage.

Les élèves de Marvin Minsky, au MIT, développent divers systèmes (BASEBALL (1961), SIR (1964), STUDENT (1964), ELIZA (1966) ...) mettant en œuvre des mécanismes de traitement simples, à base de mots-clés. Leurs résultats, en particulier le comportement assez spectaculaire d'ELIZA, qui simule un dialogue entre un psychiatre et son patient, relancent les recherches sur la compréhension automatique du langage.

Des réflexions importantes sur la représentation des connaissances voient aussi le jour, principalement à l'initiative de Ross Quillian, qui préconise l'utilisation de réseaux sémantiques pour représenter le sens des mots et des phrases en explicitant les relations des divers concepts entre eux grâce à des liens qui indiquent le sens des relations.
Terry Winograd, en réalisant en 1972 SHRDLU, le premier logiciel capable de dialoguer en anglais avec un robot, dans le cadre d'un micro-monde (quelques blocs de couleurs et de formes variées, posés sur une table).

Les années 70 voient ensuite le développement d'approches surtout sémantiques (Roger Schank, Yorick Wilks, ...), le rôle de la syntaxe étant pratiquement omis ou, tout du moins considéré comme secondaire. L'importance du contexte et le rôle essentiel d'une bonne connaissance du domaine traité pour comprendre un texte est ainsi mis en avant. M. Minsky tente alors d'élaborer un cadre général de représentation des connaissances, les frames, alors que R. Schank s'efforce d'identifier clairement les diverses connaissances nécessaires dans un système interprétant le langage naturel. Les recherches ont alors cessé de se limiter à l'interprétation de phrases seules pour aborder le traitement d'unités plus importantes comme les récits et les dialogues.

Parallèlement, les modèles syntaxiques connaissent en informatique des développements et des raffinements continus, et des algorithmes de plus en plus performants sont proposés pour analyser les grammaires les plus simples (grammaires régulières et algébriques). Depuis Chomsky, ces formalismes grammaticaux sont toutefois considérés comme trop simples pour modéliser correctement les phénomènes observés dans les langues naturelles. Ces développements des grammaires formelles sont donc largement sous-estimés, jusqu'à ce qu'au milieu des années 70, divers travaux théoriques, en particulier ceux de Ronald Kaplan et de Martin Kay, réhabilitent ces formalismes dans le cadre du traitement de la morphologie et de la phonologie des langues naturelles. Ces années voient également une recrudescence des travaux en syntaxe des langues naturelles, et à l'émergence de nouveaux formalismes de description grammaticale, qui étendent de manière informatiquement gérable les grammaires algébriques. Ce sont tout d'abord les réseaux de transition augmentés (abréviation anglaise ATN), puis les grammaires d'unification.

Bien évidemment, et quelle que soit leur élégance, les propositions issues de l'intelligence artificielle jusqu'au début des années 80 ne permettent pas d'échapper à l'obligation d'affronter la complexité de la tâche de description préalable des connaissances sur la langue et sur le monde. C'est pourquoi une partie importante des travaux actuels vise à analyser et à formaliser des mécanismes d'acquisition automatique des connaissances, qui permettent d'extraire directement de lexiques ou de corpus de documents, des règles de grammaire, ou encore des connaissances sémantiques.

Aujourd'hui, le champ du traitement du langage naturel est un champ de recherche très actif. De nombreuses applications industrielles (traduction automatique, recherche documentaire, interfaces en langage naturel), qui commencent à atteindre le grand public, sont là pour témoigner de l'importance des avancées accomplies mais également des progrès qu'il reste encore à accomplir.[14]

5. Les difficultés du TALN : ambiguïté et implicite

Les difficultés que l'on rencontre en TALN sont principalement de deux ordres, et ressortent soit de l'ambiguïté du langage, soit de la quantité d'implicite contenue dans les communications naturelles.

5.1 Ambiguïté

Le langage naturel est ambigu, et ce à quelque niveau qu'on l'appréhende. Cette ambiguïté, loin d'être marginale, est un de ses traits caractéristiques. On peut d'ailleurs voir là le résultat d'un compromis inévitable entre d'un côté une capacité d'expression quasi illimitée, et de l'autre des contraintes liées à la limitation des ressources physiologiques mises en œuvre (taille de la mémoire à long et court-terme, densité de l'espace phonétique, contraintes articulatoires, etc.). Cette ambiguïté se manifeste par la multitude d'interprétations possibles pour chacune des entités linguistiques pertinentes pour un niveau de traitement, comme en témoignent les exemples suivants :

- ✓ ambiguïté des graphèmes (lettres) dans le processus d'encodage orthographique : comparez la prononciation du i dans lit, poire, maison ;
- ✓ ambiguïté des terminaisons dans les processus de conjugaison et d'inflexion ;
- ✓ ambiguïté dans les propriétés grammaticales et sémantiques (i.e. associées à son sens) d'une forme graphique donnée : ainsi manges est ambigu à la fois morpho-syntaxiquement, puisqu'il correspond aux formes indicative et subjonctive du verbe manger), mais aussi sémantiquement. Exemple suivant :

  (a) Demain, Paul mange avec ma sœur.

  (b) Paul mange son pain au chocolat.

Ainsi que les déductions que l'on peut faire à partir de ces deux énoncés : de (a), on peut raisonnablement conclure que Paul sera assis à une table, disposera de couverts, tout ceci n'est pas nécessairement vrai dans le cas de l'énoncé (b).

- ✓ ambiguïté de la fonction grammaticale des groupes de mots, illustrée par la phrase.
- ✓ ambiguïté de la portée des quantificateurs, des conjonctions, des prépositions.

- ✓ Ambiguïté sur l'interprétation à donner en contexte à un énoncé. Comparez ainsi la « signification » de non, dans les deux échanges suivants :
  (a) Si je vais en cours demain ? Non (négation).
  أنت ذاهب الى الصف غدا! لا! (لا أعتقد ذلك (b)
  (b) Tu vas en cours demain ! Non ! (j'y crois pas).

5.2 Implicite

L'activité langagière s'inscrit toujours dans un contexte d'interaction entre deux humains, sensément dotés d'une connaissance du monde et de son fonctionnement telle que l'immense majorité des éléments de contexte nécessaires à la désambiguïsation mais aussi à la compréhension d'un énoncé naturel peuvent rester implicites. La situation change du tout au tout dès qu'une machine tente de s'insérer dans un processus de communication naturel avec un humain : la machine ne dispose pas de cette connaissance d'arrière-plan, ce qui rend la compréhension complète de la majorité des énoncés difficile, voire impossible, si l'on ne dispose pas de bases de connaissance additionnelles, donnant accès à la fois à un savoir sur le monde (ou le domaine) en général (connaissance statique) et sur le contexte de l'énonciation (connaissance dynamique).

Un exemple typique est la désambiguïsation du référent du pronom personnel-il dans les trois énoncés suivants : Le professeur envoya l'élève chez le censeur parce qu' ... :
(a). ... il lançait des boulettes (il réfère probablement à l'élève) ;
(b). ... il voulait avoir la paix (il réfère probablement au professeur) ;
(c). ... il voulait le voir (il réfère probablement au censeur).

En l'absence de telles connaissances, bien d'autres problèmes de compréhension deviennent pratiquement insurmontables : pensez par exemple aux ellipses, aux métaphores, et, plus généralement, aux figures de style.

Fort heureusement, il existe de nombreuses applications pour lesquelles ces difficultés peuvent être, dans une large mesure, circonscrites. Dès lors, en effet, que l'on restreint le cadre des textes analysés à un sous domaine particulier (textes juridiques, textes scientifiques, serveur d'information spécialisé dans les informations sportives...), il devient possible d'une part d'ignorer un grand nombre d'ambiguïtés, en particulier sémantiques (par exemple dans le

contexte de textes juridiques, on pourra probablement négliger la possibilité qu'un avocat marron désigne un fruit un peu trop mûr) ; et d'autre part de représenter formellement un grand nombre des connaissances nécessaires à la compréhension des énoncés du domaine considéré. En fait, certains domaines d'activité ou contextes d'interactions spécifiques semblent restreindre de manière drastique l'ensemble des énoncés possibles (ou acceptables), simplifiant de manière considérable le traitement de ces véritables sous-langages par une machine [14].

6. Les niveaux de traitement

J'introduis dans cette section les différents niveaux de traitement nécessaires pour parvenir à une compréhension complète d'un énoncé en langage naturel.
Vous le verrez, au fur et à mesure que le progresse dans cette hiérarchie des niveaux, les difficultés s'accumulent, et les outils aujourd'hui disponibles se font moins performants, ou ne sont opérants que pour des sous-domaines particuliers. Toutefois, bien des applications ne nécessitent pas une compréhension complète des énoncés, et ne mettent en œuvre que des traitements correspondant aux niveaux les mieux compris et automatisés.
Je brosse ici les grands domaines du TAL, en j'appuyant sur un découpage méthodologique classique dans le domaine et en linguistique :

- ***La phonétique*** concerne l'étude de Phonèmes (sons) et prosodies (variations) ;
- ***La morphologie*** concerne l'étude de la formation des mots et de leurs variations de forme ;

- ***La syntaxe*** consistant à extraire les relations grammaticales que les mots et groupes de mots entretiennent entre eux ;
- ***La sémantique*** se consacre au sens des énoncés ;
- ***La pragmatique*** prend en compte le contexte d'énonciation.

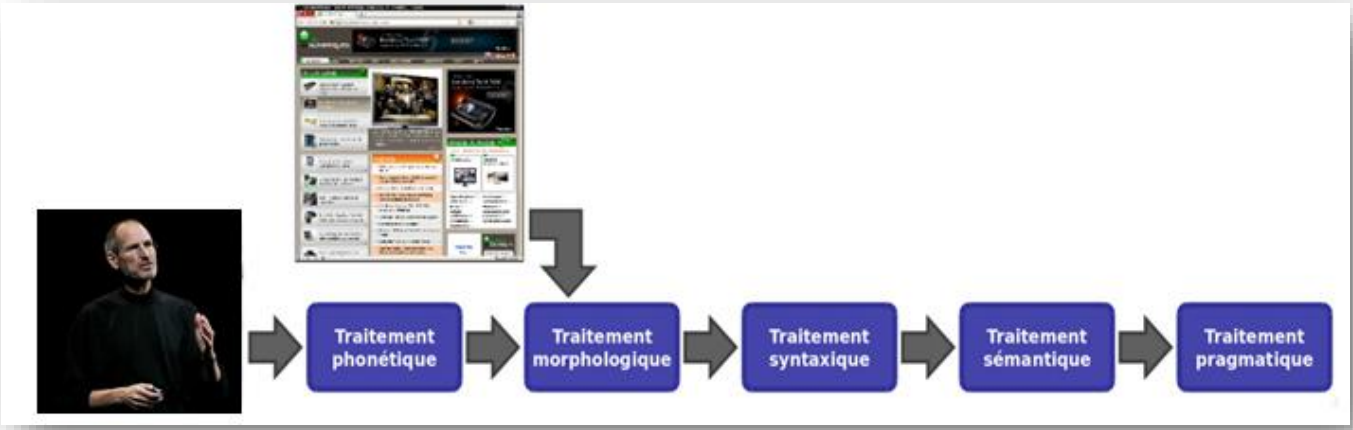


Figure 03: présente les niveaux de traitement

6.1 Phonétique

Le traitement automatique de la parole peut concrètement contribuer à éclairer de nombreuses questions concernant la variabilité phonémique à l'oral. L'exploitation de grandes masses de données permet ainsi de dégager de grandes tendances, dont une interprétation plus fine repose ensuite à la fois sur un éclairage linguistique et sur un certain nombre de précautions méthodologiques.

6.2 Morphologie

D'un point de vue informatique, un texte est une chaîne de caractères. La première étape de l'analyse d'un texte est la reconnaissance, dans cette chaîne de caractères, d'unités linguistiques de base, les mots, et la mobilisation des informations associées, puisées dans un lexique. Pour commencer, la chaîne de caractères d'entrée doit utiliser un encodage déterminé, les caractères de contrôle (fin de ligne, etc.) étant eux aussi normalisés. Élimine généralement les caractères non répertoriés. Il s'agit ensuite de segmenter la chaîne d'entrée en unités élémentaires (en anglais, *tokens*). Différents choix peuvent être effectués à ce stade, selon les séparateurs choisis : tous les caractères non alphabétiques (espaces, apostrophes, tirets...) ou les espaces seulement ; et selon que l'on prend en considération les « mots

composés » (« *pomme de terre* » = une unité,الجودة الشاملة) ou pas. En tout état de cause, on a généralement amené à distinguer la notion d'unité minimale (« token ») et celle de mot (associé à une information lexicale).

Le lexique, en première approximation, est la liste des mots de la langue, et associe à chaque mot les informations linguistiques correspondantes : catégorie syntaxique, traits morphosyntaxiques (genre, nombre, etc.), etc. Plusieurs phénomènes amènent à préciser cette définition du lexique.

- Un mot peut exister sous plusieurs formes : formes fléchies des noms, adjectifs, etc., conjugaison des verbes. On peut alors considérer une *forme canonique*, ou lemme, pour chaque mot, qui sert d'entrée dans le lexique pour l'ensemble de ses formes fléchies (singulier pour le nom, masculin singulier pour l'adjectif, infinitif pour le verbe).
- Un mot peut avoir plusieurs sens (*polysème*) : *« avocat », « coup », « livre »* en sont des exemples ; selon le cas, plusieurs entrées ou sous entrées sont alors distinguées.
- Plusieurs mots peuvent se trouver partager une forme commune (*homographes*).
- Un mot peut être construit à partir d'un autre : par dérivation ou par composition.

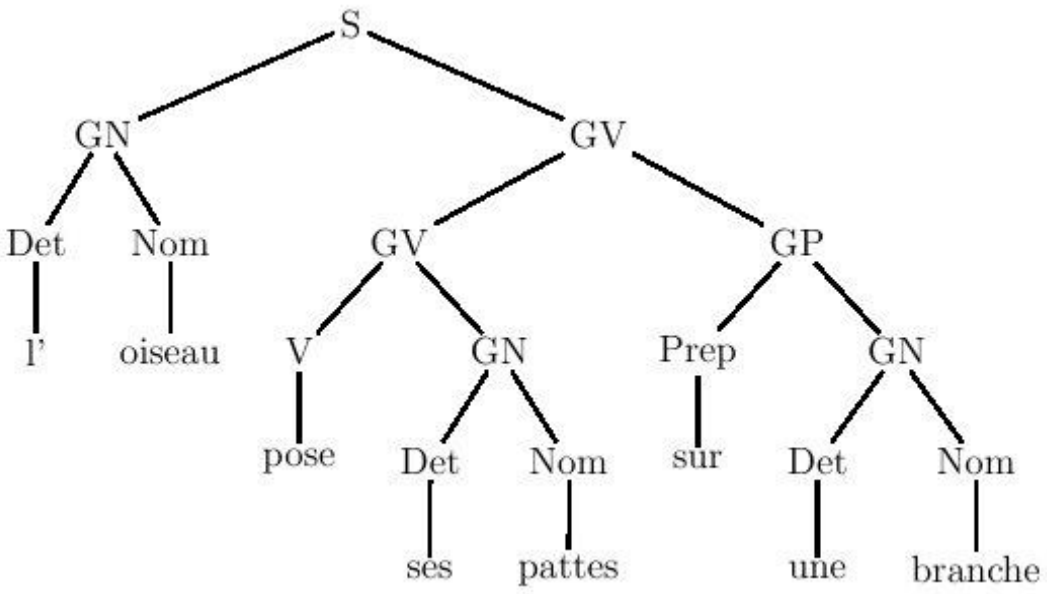
Enfin, pour de multiples raisons, tous les mots possibles d'une langue ne sont ou ne peuvent être répertoriés a priori dans un lexique. D'une part, les noms propres constituent un inventaire ouvert. D'autre part, de nouveaux mots sont créés régulièrement (néologie) par dérivation et composition, mais aussi par siglaison, abréviation, emprunt, etc. [15]

6.3 Syntaxe

Pour repérer quels mots fonctionnent ensemble dans une phrase, un premier niveau de modélisation consiste à constituer des classes de mots (catégories syntaxiques, parties du discours) possédant un fonctionnement similaire : Nom (N), Verbe (V), Adjectif (A), etc. Certaines unités, par accident (homographes: *« la », « est »*) ou de façon plus systématique, peuvent être *ambiguës* entre plusieurs catégories (ambiguïté catégorielle ou lexicale). Par exemple, chacune des unités de la phrase *« La coronarographie est normale. »* est ambiguë, ce que l'on peut noter : *« La/DET, N, PRO coronarographie/N, V est/A, N, V normale/A, N. »* On remarquera que dans le contexte de la phrase entière, aucune de ces unités n'est ambiguë.

Les relations syntaxiques entre les mots d'une phrase peuvent se représenter de plusieurs façons. Le modèle en constituants considère des groupes de mots, ou syntagmes,

généralement centrés sur un mot de tête (N, V, etc.), et les modélise par des catégories spécifiques (syntagme nominal ou SN, syntagme verbal ou SV,etc.). Ces syntagmes peuvent eux-mêmes être éléments d'autres syntagmes, et la structure d'une phrase est alors un arbre de constituants. Exemple sur un group nominal et verbal.



6.4 Sémantique

De même que pour la syntaxe, un premier niveau de modélisation consiste à constituer des classes de mots (*catégories sémantiques*). Ces classes regroupent des mots dont le sens est proche, ou au minimum (pour des classes générales) des mots qui possèdent certaines propriétés sémantiques communes. Cependant, si en syntaxe on arrive à s'accorder sur des jeux de catégories relativement consensuels (il s'agit d'une vue d'ensemble ; de près, le tableau est beaucoup plus polychrome, en sémantique aucune classification universelle n'existe (la constitution d'une classification universelle risque même d'être théoriquement impossible). Les classifications que l'on pourra utiliser (par exemple, les catégories générales de WordNet[3] reflètent nécessairement un point de vue, une prise de position culturelle ou ontologique spécifique.

3 : WordNet : est une base de données lexicale développée par des linguistes du laboratoire des sciences cognitives de l'université de Princeton depuis une vingtaine d'année.

Un mot, même syntaxiquement non ambigu, pourra posséder plusieurs sens. Le contexte permet en général de déterminer quel sens est à l'œuvre dans un énoncé.

Les mots d'une langue entretiennent un réseau riche de *relations sémantiques paradigmatiques* [15]. Dans un énoncé, les relations grammaticales sont le support de *relations sémantiques syntagmatiques*. Par exemple, les différents actants d'un événement jouent différents *rôles thématiques* : agent, thème, source, destination, etc. Ainsi, dans *« Jean donne un livre à Marie. »*, les rôles par rapport à l'événement *« donne »* pourront être : *« Jean/agent, source donne un livre/thème à Marie/destination. »* Un mot qui désigne un événement possède des propriétés combinatoires restreintes : il sélectionne comme actants certains types de mots (*restrictions de sélection*). Ces types restreints peuvent être exprimés en termes de classes sémantiques. On pourra par exemple poser pour le verbe *« donner quelque chose à quelqu'un » donner (animé, objet, animé)*, ou encore pour *« interdire » interdire (animé, animé, événement).*

La représentation sémantique finale que l'on vise à associer à un énoncé dans un système de TAL dépend de l'objectif de ce système. Cet objectif peut être l'extraction d'informations spécifiques. Un éventail d'informations plus large peut aussi être recherché. La représentation doit alors être plus complète, comme dans le système MENELAS4[15]. Les formalismes de représentation employés sont en général issus de l'Intelligence artificielle, comme les logiques de description et les Graphes Conceptuels.

6.5 Pragmatique

L'interprétation d'un énoncé dépend de son contexte. Dès que l'on veut traiter plus d'une phrase (et même pour une seule phrase), cette dimension intervient.
Le *cotexte* désigne le texte qui précède (et suit) la phrase courante. Deux facteurs concourent à faire qu'une phrase s'insère bien dans un texte.

- La *cohésion* régit la continuité du texte. Elle est assurée par l'emploi d'anaphores, l'homogénéité du thème, un emploi judicieux d'ellipses, etc.
- La *cohérence* détermine l'intelligibilité du texte. Elle s'appuie sur des structures de discours ainsi que sur les relations causales, temporelles, etc., entre les événements décrits.

4 : MENELAS : un système d'accès pour les dossiers médicaux en langage naturel.

Au-delà du texte lui-même, les conditions d'énonciation et les connaissances partagées complètent le contexte d'un énoncé. L'interprétation devra donc faire appel à des connaissances sur le monde (scénarios, plans, etc.). L'identification de structures de discours (structure de dialogue, structure argumentative, etc.) est également nécessaire selon le type de texte. De façon générale, une représentation de la situation décrite par un énoncé demande d'effectuer des inférences à partir de représentations initiales (par exemple, « littérales ») de cet énoncé et de représentations du contexte [15].

7. Conclusion

L'étude du langage naturel et des mécanismes nécessaires à la mise en œuvre à son traitement automatique par des machines est un domaine d'études foisonnant, et riche en applications potentielles ou émergentes. De nombreux progrès restent à accomplir pour mieux comprendre cette faculté et pour bâtir des systèmes capables de soutenir la comparaison avec l'humain, mais l'état des connaissances en permet aujourd'hui de proposer de nombreuses solutions efficaces à des problèmes et des demandes réels.

L'utilisation de techniques d'apprentissage automatique et d'acquisition de connaissance est donc aujourd'hui une tendance importante en TALN et spécifiquement dans le traitement automatique de la langue arabe, qui concentre les efforts de nombreuses équipes de recherche.

Chapitre 2

Traitement Automatique de la Langue Arabe (TALA)

CHAPITRE 2
Traitement automatique de la langue Arabe

1. Introduction

La langue arabe est d'une origine très différente des langues européennes. Elle fait partie du groupe des langues sémitiques. Ce groupe se divise en langues sémitiques orientales, sémitiques occidentales et sémitiques méridionales. À la différence d'autres nations; telles que les anciens égyptiens, les babyloniens et les chinois dont les systèmes d'écriture remontent à des milliers d'années, l'écriture arabe n'est apparue qu'au VIe siècle.

Certains historiens et chercheurs pensent que l'origine de l'écriture arabe est le syriaque en se basant sur:

- l'ordre primitif des lettres arabes;
- les formes de l'ancien alphabet arabe dit « coufique », qui sont comparables à celles de « l'estranghelo », une forme de l'écriture syriaque.

L'arrivée de l'Islam a profondément marqué l'histoire de la langue et de l'écriture arabe. Le Coran, livre sacré recueillant la parole de Dieu, mais aussi code juridique et moral, occupe d'emblée une place centrale dans la vie du croyant et de la communauté musulmane.

Les recherches sur le traitement automatique de l'arabe ont débuté vers les années 1970, et les premiers travaux concernaient notamment les lexiques et la morphologie. Ainsi, la croissance rapide du World Wide Web (WWW) et l'explosion primordiale des ressources de l'indexation et des outils de recherche, influent effectivement sur le développement dans le domaine de la recherche d'information multilingue [16].

2. La langue arabe

L'**arabe** (العربية, *al ʿarabīya*) est une langue sémitique aujourd'hui parlée en première ou seconde langue par plus de 220 millions de personnes au sein du monde arabe et de la diaspora arabe. La langue arabe est marquée par une importante diglossie entre l'arabe littéral, langue véhiculaire surtout écrite, et l'arabe dialectal, langue vernaculaire surtout orale. L'arabe littéral comprend l'arabe classique (pré-coranique, coranique, et post-coranique) et l'arabe standard moderne. L'arabe dialectal comprend de nombreuses variétés régionales, pas toutes intelligibles entre elles.

L'arabe est parlé à des degrés divers dans les pays arabes du Moyen-Orient, en Iran (province du Khouzistan), dans les pays d'Afrique du Nord, du Sahara, du Sahel et sur les côtes de la Corne de l'Afrique. Il est également pratiqué dans la diaspora arabe.

Figure 4 : présente les pays du monde arabe

3. Traitement automatique de la langue Arabe (TALA)

Le traitement automatique de la langue arabe est une discipline en pleine expansion, dans laquelle on voit de plus en plus de recherches et de technologies se soucier des spécificités de cette langue [9] et proposer des outils nécessaires au développement de son traitement automatique. Par ses propriétés morphologiques et syntaxiques la langue arabe est considérée Comme une langue difficile à maîtriser dans le domaine du traitement automatique de la langue [3], [4]. L'arabe doit sa formidable expansion à partir du 7eme siècle grâce à la propagation de l'islam et la diffusion du Coran [5]. Les recherches pour le traitement automatique de l'arabe ont débuté vers les années 1970. Les premiers travaux concernaient notamment les lexiques et la morphologie.

Avec la diffusion de la langue arabe sur le Web et la disponibilité des moyens de manipulation de textes arabes, les travaux de recherche ont abordé des problématiques plus variées comme la syntaxe, la traduction automatique, l'indexation automatique des documents, la recherche d'information, etc.

Par exemple le mot français *ferme*, est hors contexte, un substantif, un adjectif ou un verbe. Alors que le mot arabe RaLaKa غَلَقَ est un verbe à la 3è personne masculin singulier de l'accompli actif, par contre sa forme non voyellée غلق (dans l'exemple donné ne sont représentées que les consonnes RLK) admet quatre catégories grammaticales :

- Substantif masculin singulier (RaLKun : une fermeture),
- Verbe à la 3è personne masculin singulier de l'accompli actif (RaLaKa : il a fermé ou RaLLaKa il a fait fermé),
- Verbe à la 3è personne masculin singulier de l'accompli passif (RuLiKa : il a été fermé),
- Verbe à l'impératif 2è personne masculin singulier (RaLLiK: fais fermer).

Les voyelles jouent un rôle proche des accents en français pour un mot comme *peche* qui peut être interprété comme *pêche, pèche et péché*. Par contre, en arabe chaque lettre de chaque mot devrait posséder sa voyelle ce qui n'est en général pas le cas, on constate donc l'étendue du rôle que jouent les voyelles dans les mots arabes, non seulement parce qu'elles enlèvent l'ambiguïté, mais aussi parce qu'elles donnent l'étiquette grammaticale d'un mot indépendamment de sa position dans la phrase.

3.1 Particularité de la langue arabe

L'alphabet de la langue arabe compte 28 consonnes (tableau01). L'arabe s'écrit et se lit de droite à gauche les lettres changent de forme de présentation selon leur position (au début, au milieu ou à la fin du mot). (Le Tableau 2) montre les variations de la lettre ع (Ayn). Toutes les lettres se lient entre elles sauf (ا, و, ر, ز, د ذ) qui ne se joignent pas à gauche.

Lettre arabe	Correspondant français	Prononciation
أ	a	Alef
ب	b	Ba'
ت	t	Ta
ث	th	Tha
ج	j	Jim
ح	h	Hha
خ	kh	Kha
د	d	Dal
ذ	d	Thal
ر	r	Ra
ز	z	Zayn
س	s	Sin
ش	sh	Shin
ص	s	Sad

Lettre arabe	Correspondant français	Prononciation
ض	d	Dad
ط	t	Tah
ظ	z	Zah
ع	''	Ayn
غ	gh	Ghayn
ف	f	Fa
ق	q	Qaf
ك	k	Kaf
ل	l	Lam
م	m	Mim
ن	n	Nun
ه	h	Ha
و	w	Waw
ي	y	Ya

Tableau 01 : presente les 28 lettres de la langue arabe

La représentation morphologique de l'arabe est assez complexe en raison de la variation morphologique et du phénomène d'agglutinement; les lettres changent de formes selon leur position dans le mot (isolée, initiale, médiane et finale). (tableau02) montre un exemple des différentes formes de la lettre «qaf » dans différentes positions. Nous pouvons observer ainsi plusieurs caractéristiques générales de cette langue suivant le détail ci-après :

Isolée	Initiale	Médiane	Finale
ق	قـ	ـقـ	ـق
	قرآن	القرآن	غسق

Tableau 02 : représente différentes écritures de la lettre **« qaf ».**

Un mot arabe s'écrit avec des consonnes et des voyelles. Les voyelles sont ajoutées au-dessus ou au-dessous des lettres (ْ ,ُ ,ِ ,َ). Elles sont nécessaires à la lecture et à la compréhension correcte d'un texte, elles permettent de différencier des mots ayant la même représentation. Pour une meilleure précision de la prononciation, des signes ont été inventés. Il s'agit de trois voyelles brèves et de sept signes orthographiques qui s'ajoutent aux consonnes. Ces trois voyelles brèves sont :

- Fatha « َ », elle surmonte la consonne et se prononce comme un «a» français ;
- Damma « ُ », elle surmonte la consonne et se prononce comme un «ou» français ;
- Kasra « ِ », elle se note au-dessous de la consonne et se prononce comme un « i » français).

Les sept signes orthographiques sont :

- Sukun « ْ » : ce signe indique qu'une consonne n'est pas suivie (ou muet) par une voyelle. Il est noté toujours au-dessus de la consonne;
- Les trois signes de tanwin : lorsque (la Fatha, la Kasra et la Damma) sont doublées, elles prennent un son nasal, comme si elles étaient suivies de «n» et on les prononce respectivement :
  - an « ً » pour les Fathatan ;
  - in « ٍ » pour les Kasratan ;
  - un « ٌ » pour les Dammatan.
- Chadda « ّ » comme dans le français, l'arabe peut renforcer une consonne quelconque;
- Wasla « ٱ» : quand la voyelle d'un Alif au commencement d'un mot doit être absorbée par la dernière voyelle du mot qui précède;
- Madda « آ» : la madda (prolongation) se place sur l'Alif pour indiquer que cette lettre tient lieu de deux alifs consécutifs ou qu'elle ne doit pas porter le Hamza. [6]

	Interprétation I		Interprétation II		Interprétation III	
كتب	كَتَبَ	Il a écrit	كُتِبَ	Il a été écrit	كُتُبٌ	Des livres
شعر	شَعَرَ	Il a senti	شِعْرٌ	Poème	شَعْرٌ	Chevelure

Tableau 03 : montre un exemple pour les mots « كتب » et « شعر »

Les lettres lunaires initiales d'un nom n'assimilent pas l'article qui les précède et par « ا ب ج ح خ ع غ ف ق ك م و ي » : conséquent ne reçoivent pas le chadda. La lettre ال est prononcée.
Les lettres solaires initiales d'un nom assimilent l'article qui les précède et reçoivent « ت ث د ذ ر ز س ش ص ض ط ظ ل ن » : ainsi le chadda. La lettre ال est muette. Les caractères de la langue arabe n'appartiennent pas au code ASCII, d'où la nécessité d'utiliser un autre code qui prend en charge la langue arabe, ce code est l'Unicode, ce dernier permet de coder tous les caractères utilisés par la langue arabe en mode 16 bits.

3.2 Morphologie arabe

Le lexique arabe comprend trois catégories de mots : verbes, noms et particules. Les verbes et noms sont le plus souvent dérivés d'une racine à trois consonnes radicales [7]. Une famille de mots peut ainsi être générée d'un même concept sémantique à partir d'une seule racine à l'aide de différents schèmes. Ce phénomène est caractéristique à la morphologie arabe. On dit donc que l'arabe est une langue à racines réelles à partir desquelles on déduit le lexique arabe selon des schèmes qui sont des adjonctions et des manipulations de la racine. Le Tableau 4 donne quelques exemples de schèmes appliqués aux mots كتب *écrire* et حمل *porter*. On peut ainsi dériver un grand nombre de noms, de formes et de temps verbaux.

schèmes	KTB	كتب	Notion d'écrire
فَاعِل	KâTiB	كَاتِب	écrivain
فَعَلَ	KaTaBa	كَتَبَ	a écrit
مَفْعَل	maKTaB	مَكْتَب	bureau
فُعِلَ	KuTiBa	كُتِبَ	a été écrit

HML	حمل	Notion de porter
HâMiL	حَامِل	porteur
HaMaLa	حَمَلَ	a porté
maHMaL	مَحْمَل	brancard
HuMiLa	حُمِلَ	a été porté

Tableau 04 : La dérivation de mots كتب écrire et حمل porter

La majorité des verbes arabes ont une racine composée de 3 consonnes. L'arabe comprend environ 150 schèmes ou patrons dont certains plus complexes, tel le redoublement d'une consonne ou l'allongement d'une voyelle de la racine, l'adjonction d'un ou de plusieurs éléments ou la combinaison des deux. Une autre caractéristique est le caractère flexionnel des mots : les terminaisons permettent de distinguer le mode des verbes et la fonction des noms [7].

3.3 Structure d'un mot

La définition du mot du point de vue du traitement automatique se heurte à des considérations syntaxiques et sémantiques. Dans le domaine des langages formels, la transformation du flux de caractères représentant un texte en une suite d'unités mieux adaptées aux traitements ultérieurs, est habituellement appelée segmentation (tokenization), et les unités produites les segments (tokens) sont construites sur la base de définitions purement orthographiques. Le problème posé par de telles techniques pour des applications de traitements de langue est malheureusement l'absence de correspondance biunivoque entre les segments ainsi identifiés et les unités textuelles élémentaires (les mots) manipulées dans le traitement linguistique. En arabe cette séquence de lettres est appelée le mot graphique.
Les mots sont séparés par des espaces et d'autres signes de ponctuation. Néanmoins, des prépositions sont agglutinées au mot (apparaissant après eux), faisant des limites invisibles entre le mot et la préposition.

Plusieurs types d'affixes sont agglutinés au début et à la fin des mots : antéfixes, préfixes, suffixes et post fixes. La représentation suivante schématise une structure possible d'un mot. Note que la lecture et l'écriture d'un mot se fait de droite vers la gauche.

Post fixe	**Suffixe**	**Corps Schématique**	**Préfixe**	**Antéfixe**

Tableau 05 : Montre une structure possible d'un mot.

- *Antéfixes* sont des prépositions ou des conjonctions.
- *Préfixes* et *Suffixes* expriment les traits grammaticaux et indiquent les fonctions : cas du nom, mode du verbe et les modalités (nombre, genre, personne,…)
- *Post fixes* sont des pronoms personnels.

Exemple :

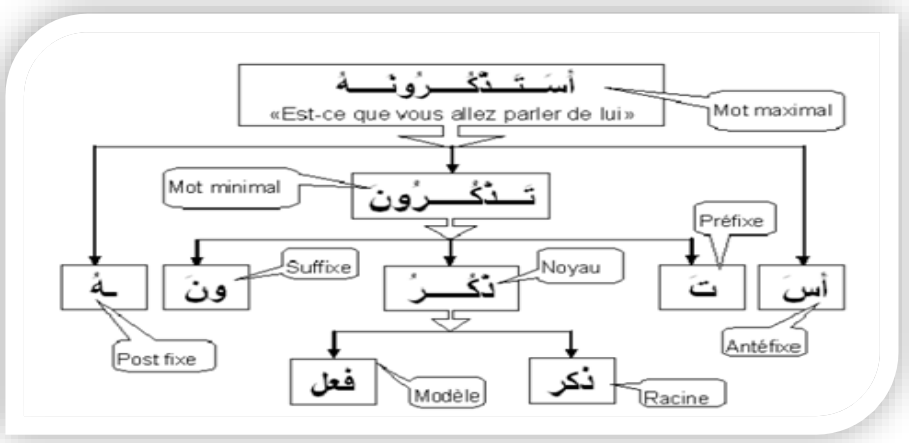


Figure 05 : La segmentation de mot « أَسَتَذْكُرُونَهُ ».

3.3.1 Catégories des mots

L'arabe considère 3 catégories de mots :

- Le verbe : entité exprimant un sens dépendant du temps, c'est un élément fondamental auquel se rattachent directement ou indirectement les divers mots qui constituent l'ensemble.
- Le nom : l'élément désignant un être ou un objet qui exprime un sens indépendant du temps.
- Les particules : entités qui servent à situer les événements et les objets par rapport au temps et l'espace, et permettent un enchaînement cohérent du texte.

3.3.1.1 Le verbe

La plupart des mots en arabe, dérivent d'un verbe de trois lettres. Chaque verbe est donc la racine d'une famille de mots. Comme en français, le mot en arabe se déduit de la racine en rajoutant des suffixes ou des préfixes.

La conjugaison des verbes dépend de plusieurs facteurs :

- Le temps (accompli, inaccompli).
- Le nombre du sujet (singulier, duel, pluriel).
- Le genre du sujet (masculin, féminin).
- La personne (première, deuxième et troisième)

- Le mode (actif, passif).

Par exemple : ب + ت + ك, *K+T+B* donne le verbe كتب *KaTaBa* (écrire).

Dans tous les mots qui dérivent de cette racine, on trouvera ces trois lettres K, T, B. La conjugaison des verbes se fait en ajoutant des préfixes et des suffixes, un peu comme en français. La langue arabe dispose de trois temps :

- L'accompli : correspond au passé et se distingue par des suffixes (par exemple pour le pluriel féminin on a كتبن KaTaBna, *elles ont écrit* et pour le pluriel masculin on كتبوا KaTaBuu, *ils ont écrit*).
- L'inaccompli présent : présente l'action en cours d'accomplissement, ses éléments sont préfixés (يكتب yaKTuBu *il écrit*; تكتب taKTuBu, *elle écrit*).
- L'inaccompli futur : correspond à une action qui se déroulera au futur et est marqué par l'antéposition de « س » *sa* ou « سوف » *sawfa* au verbe (سيكتب sayaKTuBu *il écrira*, سوف يكتب sawfa yaKTuBu *il va écrire*).

3.3.1.2 Les noms

Les substantifs arabes sont de deux catégories, ceux qui sont dérivés de la racine verbale et ceux qui ne le sont pas comme les noms propres et les noms communs. Dans le premier cas, le fait que le nom soit dérivé d'un verbe, il exprime donc une certaine sémantique qui pourrait avoir une influence dans la sélection des phrases saillantes d'un texte pour le résumé.

La déclinaison des noms se fait selon les règles suivantes:

- Le féminin singulier: On ajoute le ‘ة’, exemple صغير *petit* devient صغيرة *petite.*
- Le féminin pluriel : De la même manière, on rajoute pour le pluriel les deux lettres ‘ا’ ,’ت‘ , exemple صغير *petit* devient صغيرا ت *petites.*
- Le masculin pluriel : Pour le pluriel masculin on rajoute les deux lettres ين ou ون dépendamment de la position du mot dans la phrase (sujet ou complément d’objet), exemple : الراجع *revenant* devient الراجعين ou الراجعون *revenants.*

- Le Pluriel irrégulier: Il suit une diversité de règles complexes et dépend du nom. exemple : طفل *un enfant* devient أطفال *des enfants.*

Le phénomène du pluriel irrégulier dans l'arabe pose un défi à la morphologie, non seulement à cause de sa nature non concaténâtes, mais aussi parce que son analyse dépend fortement de la structure [8] comme pour les verbes irréguliers.

Certain dérivés nominaux associent une fonction au nom :

- Agent (celui qui fait l'action),
- Objet (celui qui a subi l'action),
- Instrument (désignant l'instrument de l'action),
- Lieu.

Pour les pronoms personnels, le sujet est inclus dans le verbe conjugué. Il n'est donc pas nécessaire (comme c'est le cas en français) de précéder le verbe conjugué par son pronom. On distinguera entre singulier, duel (deux) et pluriel (plus de deux) ainsi qu'entre le masculin et féminin.

3.3.1.3 Les particules

Ce sont principalement les mots outils comme les conjonctions de coordination et de subordination. Les particules sont classées selon leur sémantique et leur fonction dans la phrase, on en distingue plusieurs types (introduction, explication, conséquence, …). Elles jouent un rôle important dans l'interprétation de la phrase [8]. Elles servent à situer des faits ou des objets par rapport au temps ou au lieu, elles jouent également un rôle clé dans la cohérence et l'enchaînement d'un texte. Comme exemple de particules qui désignent un temps بعد ، قبل ، منذ *pendant, avant, après*, un lieu حيث *où*, ou de référence الذين١ *ceux*,….

Ces particules seront très utiles pour notre traitement à deux niveaux :

- Elles font partie de l'anti dictionnaire qui regroupe les termes à ne pas prendre en considération lors de calcul de fréquence de distribution des mots,
- Elles identifient des propositions composant une phrase.

Les particules peuvent avoir des préfixes et suffixes ce qui rajoute une complexité quant à leur identification.

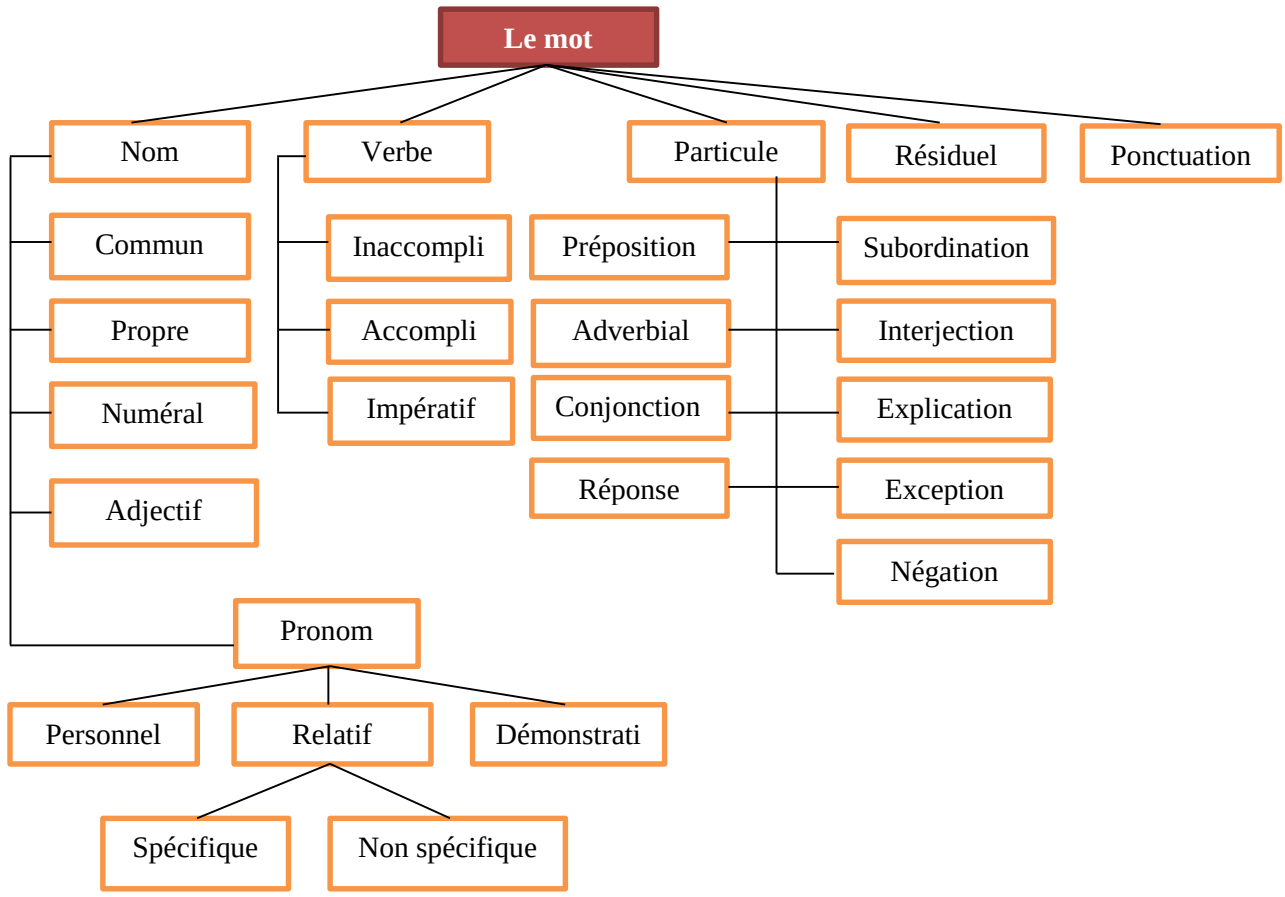


Figure 06 : Classification des unités lexicales proposée par [8].

4. Les particularités de la langue arabe

Dans la section suivante, j'essayé de donner une brève présentation de ces problèmes, qui rendent le traitement automatique de la langue arabe une tâche difficile à maîtriser.

4.1 Les voyelles

En Arabe écrit, les voyelles (signes diacritiques) sont omises et le résultat de cette omission est que les mots tendent à avoir un haut niveau d'ambiguïté. qui risque de générer une certaine ambiguïté à deux niveaux :

- Sens du mot.
- Difficulté à identifier sa fonction dans la phrase, (différencier entre le sujet et le complément,…). Ceci peut influencer les fréquences des mots étant donné qu'elles sont calculées après la détection de la racine ou la lemmatisation des mots qui est basée sur la suppression de préfixes et suffixes. Lors du calcul des scores à partir des titres, il peut arriver que des mots soient considérés comme dérivants d'un même concept alors qu'ils ne le sont pas. Dans l'exemple, en utilisant la distribution des mots ou le titre avec ou sans lemmatisation, la phrase 3 aura un score le plus important alors que les phrases 1 et 2 semblent plus intéressantes, ce qui n'aurait pas été le cas avec un texte voyellé.

Figure 07 : Exemple sur l'effet du mot non voyelle « العلم» sur les extraits.

L'ambiguïté vient du mot العلم *la science* ou *drapeau* alors que voyellé on aura العِلْمُ pour *la science* et العَلَمُ pour *le drapeau*. Cette ambiguïté pourrait, dans certains cas, être levée soit par une analyse plus profonde de la phrase ou des statistiques (par exemple il est plus probable d'avoir « العلم الوطني » *le drapeau national* que la *science nationale*). De plus la capitalisation n'est pas employée dans l'arabe ce qui rend l'identification des noms propres, des acronymes, et des abréviations encore plus difficile [10].

Comme la ponctuation est rarement utilisée, on doit ajouter une phase de segmentation de phrase pour l'analyse d'un texte [11].

4.2 Agglutination

Contrairement aux langues latines, en arabe, les articles, les prépositions, les pronoms, etc. collent aux adjectifs, noms, verbes et particules auxquels ils se rapportent. Comparé au français, un mot arabe peut parfois correspondre à une phrase française (Souissi, 1997).

Exemple : le mot arabe « أتتذكروننا » correspond en Français à la phrase "Est-ce que vous vous souvenez de nous ?".

Cette caractéristique peut engendrer une ambiguïté au niveau morphologique. En effet, il est parfois difficile de distinguer entre une proclitique ou enclitique et un caractère original du mot. Par exemple, le caractère "و" dans le mot "وصل" (il est arrivé) est un caractère original alors que dans le mot « وفتح » (il a ouvert), il s'agit d'une proclitique [11].

4.3 Irrégularité de l'ordre des mots dans la phrase

L'ordre des mots en arabe est relativement libre. D'une manière générale, on met au début de la phrase le mot sur lequel on veut attirer l'attention et l'on termine sur le terme le plus long ou le plus riche en sens ou en sonorité. Cet ordre provoque des ambiguïtés syntaxiques artificielles, dans la mesure où il faut prévoir dans la grammaire toutes les règles de combinaisons possibles d'inversion de l'ordre des mots dans la phrase .

Ainsi par exemple, on peut changer l'ordre des mots dans la phrase (tableau06) pour obtenir deux phrases ayant le même sens.

Verbe + sujet + complément	فعل + فَاعل + متمم	Est allé le garçon à l'école	ذهب الولدُ إِلَى المدرسة
sujet + verbe + complément	فَاعل + فعل + متمم	Le garçon est allé à l'école	الولدُ ذهب إِلَى المدرسة
complément + verbe + sujet	متمم + فعل + فَاعل	A l'école est allé le garçon	إِلَى المدرسة ذهب الولدُ

Tableau 06 : Exemple de combinaisons possibles d'inversion de l'ordre des mots dans la phrase.

4.4 Absence de ponctuation régulière

La langue arabe n'est pas appuyée principalement sur les signes de ponctuations et les marqueurs typographiques ; il est à noter que ces derniers ne sont pas utilisés de façon régulière dans les textes arabes actuels, et même dans le cas où ils y figurent, ils ne sont pas gérés par des règles précises d'utilisation.

Par ailleurs, nous pouvons trouver tout un paragraphe arabe ne contenant aucun signe de ponctuation à part un point à la fin de ce paragraphe. Ainsi, il convient de noter que la présence des signes de ponctuation ne peut pas guider la segmentation comme c'est le cas pour d'autres langues latines, telles que le français ou l'anglais. Ainsi, la segmentation de textes arabes doit être guidée non seulement par les signes de ponctuations et les marqueurs typographiques mais aussi par des particules et certains mots tels que les conjonctions de coordination, etc. [17].

4.5 Détection de racine

Pour détecter la racine d'un mot, il faut connaître le schème par lequel il a été dérivé et supprimer les éléments flexionnels (antéfixes, préfixes, suffixes, post fixes) qui ont été ajoutés. J'utilise la liste de préfixes et de suffixes proposé par [17] voir (Tableau 07). Plusieurs d'entre eux ont été utilisés pour la lemmatisation de mots arabes; ils ont été déterminés par un calcul de fréquence sur une collection d'articles arabes de l'Agence France Press (AFP).

Préfixes							
لا	فبـ	للـ	كمـ	بمـ	وتـ	بتـ	والـ
با	وا	ليـ	فمـ	لهـ	ستـ	يتـ	فالـ
	فا	ويـ	الـ	ومـ	نتـ	متـ	بالـ
suffixes							
ا	ة	ين	ية	هم	ته	وه	ات
	ه	يه	تك	هن	تم	ان	وا
	ي	ية	نا	ها	كم	تي	ون

Tableau 07 : La liste de Préfixes et suffixes les plus fréquents.

L'analyse morphologique devra donc séparer et identifier des morphèmes semblables aux mots préfixés comme les conjonctions wa-‘و’ et fa- ‘ف’ , des prépositions préfixées comme bi-‘ بـ’ et li- ‘لِ’, l'article défini ‘ال’ , des suffixes de pronom possessif.

La phase d'analyse morphologique détermine un schème possible. Les préfixes et suffixes sont trouvés en enlevant progressivement des préfixes et des suffixes et en essayant de faire correspondre toutes les racines produites par un schème afin de retrouver la racine [16].

Lorsqu'un mot peut être dérivé de plusieurs racines différentes, la détection de la racine est encore plus difficile, en particulier en absence de voyelles [16].

4.6 Le caractère '_'

Les typographes font un usage fréquent du caractère '-' (appelé Kashida), qui permet l'allongement du trait au milieu des mots, pour une meilleure lisibilité, pour limiter les espaces blancs sur une ligne justifiée, voire pour des raisons purement esthétiques. Or cet usage peut nuire aux analyses automatiques : ce caractère ne fait pas partie de l'alphabet arabe, il est considéré comme un intrus par le système d'analyse automatique. Il faut donc recourir à un sous-programme particulier afin de l'éliminer. Exemple : le mot الكتاب : peut-être écrit de plusieurs façons : الكتاب, الكتاب, الكتاب, …etc.

4.7 Mots étrangers translittérés en arabe

Les translittérations en arabe de mots étrangers posent un problème, puisqu'ils n'ont pas de racine en arabe. Les mots translittérés sont considérés comme inconnus par l'analyseur. Quelques items étrangers méritent une attention particulière en raison de leurs fréquences élevées. Exemple: أورو, دولار …etc.

4.8 Segmentation de phrase

La reconnaissance de la fin de phrase est délicate car la ponctuation n'est pas systématique et parfois les particules délimitent les phrases. Pour la segmentation de texte [16] utilise :

• Une segmentation morphologique basée sur la ponctuation,

• Une segmentation basée sur la reconnaissance de marqueurs morphosyntaxiques ou des mots fonctionnels comme : حتى ، لكن ، أي ، و ، أو *ou, et, c.à.d, mais, quand.*

Cependant, ces particules peuvent jouer un autre rôle que celui de séparer les phrases.

5. Difficultés de traitement automatique de la langue arabe

5.1 La segmentation de textes

La segmentation d'un texte est une étape fondamentale pour son traitement automatique ; son rôle est de découper le texte en unités d'un certain type qu'on aura défini et repéré préalablement. En effet, l'opération de segmentation d'un texte consiste à délimiter les segments de ses éléments de base qui sont les caractères, en éléments constituants différents niveaux structurels tels que : paragraphe, phrase, syntagme, mot graphique, mot-forme, morphème, etc.

Toutefois, les particularités de la langue arabe, rend la segmentation arabe toujours différente, il n'y a pas de majuscules qui marquent le début d'une nouvelle phrase. De plus, les signes de ponctuation, ne sont pas utilisés de façon régulière. D'après l'étude réalisée par Belguith [17], certaines particules comme "et | و ", "donc | ف", etc. jouent un rôle principal dans la séparation de phrases et peuvent être déterminantes pour guider la segmentation.

5.2 L'analyse morphologique

La morphologie est un niveau essentiel dans les systèmes de traitement automatiques de la langue. L'opération de l'analyse morphologique tient à étudier la forme d'un mot (unités lexicales) en faisant une analyse interne de la structure de ce dernier. Le but étant de décomposer un mot à des éléments plus petits (préfixes, suffixes, etc.) selon des règles de combinaison relatives à ces derniers. À proprement parler, l'analyse morphologique ne fait que la séparation et l'identification des morphèmes semblables aux mots préfixés (comme les conjonctions "wa | وَ" et "fa | فَ", etc.), des prépositions préfixées (comme "bi |بِ. " et "li | لِ", l'article défini "ال", etc.), des suffixes de pronom possessif.

La phase d'analyse morphologique détermine un schéma possible. Les préfixes et suffixes sont trouvés en enlevant progressivement des préfixes et des suffixes et en essayant de faire correspondre toutes les racines produites par un schème afin de retrouver la racine. Le problème principal de cette analyse réside dans l'agglutination et l'absence de voyellation. Pour l'agglutination et contrairement aux langues latines, en arabe, les pronoms, les

prépositions, les articles, les conjonctions, et autres particules collent aux noms, verbes, adjectifs et particules auxquels ils se rapportent. Comparé au français, un mot arabe peut parfois correspondre à une phrase française.

Cette caractéristique engendre une ambiguïté morphologique au cours de l'analyse. Ainsi, la reconnaissance des unités lexicales qui composent une unité morphologique n'est pas toujours facile à détecter. Le problème est de reconnaître que la bonne segmentation réside ainsi, dans la difficulté de distinction entre une proclitique ou enclitique et un caractère original du mot. Par exemple, le caractère "و" dans le mot "il est arrivé | وصل" est un caractère original alors que dans le mot "et il a ouvert | وفتح", il s'agit plutôt d'une proclitique [17].

L'absence de voyellation pose un autre problème important. En effet, les mots non voyellés engendrent beaucoup de cas ambigus au cours de l'analyse (e.g. le mot non voyellé "فصل " pris hors contexte peut être un verbe au passé conjugué à la troisième personne du singulier "il a licencié | فَصَلَ", ou un nom masculin singulier "chapitre/ saison | فَصْلٌ", ou encore une concaténation de la conjonction de coordination "puis | فَ " avec le verbe "صل" : impératif du verbe lier conjugué à la deuxième personne du singulier masculin).

5.3 L'étiquetage grammatical

L'étiquetage grammatical est l'opération qui consiste à attribuer à chacun des mots d'un texte la catégorie (non, verbe, adjectif, article défini, etc.) qui est la sienne dans le contexte où il apparaît.

La difficulté de l'étiquetage grammatical s'amplifie lorsque les textes visés se présentent sous leur forme non pas voyellée, mais partiellement voyellée ou encore totalement non voyellée, ce qui correspond au cas le plus courant.

Dans ces conditions, le but général de l'étiquetage grammatical consiste à répondre à la question suivante : Comment associer aux différents mots qui composent un texte l'étiquette qui leur convient, compte tenu du contexte où ils occurrent ? Ainsi, le problème des étiquettes grammaticales est souvent posé lorsque les textes désirés sont sous leurs formes non ou partiellement voyellées, plutôt qu'à leurs formes voyellées .

Le problème de la voyellation d'un mot est ainsi posé du fait que le choix de l'accentuation qui convient au mot est difficile et dépend essentiellement du contexte.

Le tableau 08 présente le problème d'ambiguïté grammaticale rencontrée lors de l'attribution catégorique d'un mot non voyéllé "ktb | كتب", qui admet au moins cinq étiquettes grammaticales qui sont les suivantes :

Exemple de voyellation	Étiquettes grammaticales
كُتُبٌ | kutubun : des livres	substantif, masculin, pluriel
كَتَبٌ | katbun : un écrit	substantif, masculin, singulier
كَتَبَ | kataba : il à écrit	verbe, 3ème personne masculin, singulier de l'accompli actif
كُتِبَ | kutiba : il a été écrit	verbe, 3ème personne masculin, singulier de l'accompli passif
كَتِّبْ | kattib : fais écrire	verbe à l'impératif, 2ème personne masculin, singulier

Tableau 08 : Exemple d'étiquettes grammaticales attribuées selon la voyellation [17].

5.4 L'analyse syntaxique

L'analyse syntaxique permet d'associer à un énoncé sa ou ses structures syntaxiques possibles, en identifiant ses différents constituants et les rôles que ces derniers entretiennent entre eux. Toutefois, l'analyse syntaxique prend en entrée le résultat de l'analyse lexicale (éventuellement de l'étiquetage morpho-syntaxique) et fournit en sortie une structure hiérarchisée des groupements structurels et des relations fonctionnelles qui unissent les groupements.

Enfin, il est à signaler que les ambiguïtés vocaliques et grammaticales, relatives à la non voyellation des mots, pose des difficultés au niveau de l'analyse syntaxique. Ainsi, une phrase, en absence de la voyellation, peut être interprétée et traduite selon plusieurs interprétations qui sont toutes syntaxiquement correctes.

6. Outils de traitement automatique de la langue arabe

On trouve dans l'arabe 3 outils principale pour le traitement automatique de la langue arabe, Les analyseur morphologiques, les concordances, les racineurs.

6.1 Analyseurs morphologiques

L'analyseur morphologique segmente les unités lexicales, repère les différents composants et atteste leur appartenance à la langue.

6.1.1 PROLOG arabe analyseur de Shaalan

(Shaalan, 1989) signale un projet de maîtrise au Caire Université. est un moteur de règles écrites dans SICStus Prolog et a besoin de fond en Prolog qui est difficile à atteindre par un linguiste typique. Elle est antérieure normes de codage modernes, en utilisant un système de translittération.

6.1.2Buckwalter

L'analyseur de Buck Walter développé par LDC (Linguistic Data Consortium) permet de segmenter chaque unité lexicale en une séquence du type préfixe-stem-suffixe. Le préfixe est une combinaison de 0- 4 caractères, le suffixe est composé de 0 à 6 caractères et le stem comprend un à plusieurs caractères. Il est constitué principalement de trois lexiques : préfixes (548 entrées), suffixes (906 entrées), et stem (78839 entrées). Les lexiques sont complétés par trois tables de compatibilité utilisés pour couvrir toutes les possibilités de combinaisons préfixe-stem (2435 entrées), suffixe-stem (1612 entrées) et préfixe-suffixe (1138 entrées). Ainsi, l'analyseur donne en sortie l'unité lexicale, sa catégorie morphosyntaxique et sa traduction anglaise.

6.1.3 Ahmed Processeur de calcul de l'arabe morphologie

Ahmed 2000 est une thèse de maîtrisées sciences de l'Université du Caire, de rapports sur la théorie et l'application d'un modèle hybride de l'analyse morphologique arabe: "MorphOS" peuvent être considérés comme une démonstration de la façon dont une base de connaissances axées sur les règles et d'une base de connaissances statistiques peut être résoudre les problèmes en linguistique computationnelle.

6.1.4 Khoja tagger APT

APT, arabe partie du discours Tagger, (Khoja 2001 et al Khoja 2005), utilise une combinaison de techniques statistiques et à base de règles, comme elle croit que cette

technique permet d'obtenir des taux de précision plus élevés. Les balises du tag set APT sont essentiellement issues du tag set (BNC anglais), modifié avec certains concepts de la grammaire traditionnelle arabe. La raison en est que l'arabe a ses propres systèmes syntaxiques, sémantiques et morphologiques uniques, qui Mahé difficile à adapter à la tag set utilisé pour les langues indo-européennes. Le jeu d'étiquettes contient des balises et thés sont affectés à des mots. Un corpus de 50,000 mots du journal saoudien Al-Jaziira a été utilisé pour former le tagger.

6.1.5 Sakhr Analyseur morphologique

Le Sakhr Société a également produit un analyseur morphologique, qui est désigné par Multi-Mode processeur morphologique (MMMP). Le site Sahhr http://www.zahhr.com/ prétend que leur programme couvre moderne et l'arabe classique, et il identifie la forme de base en supprimant tous les affixes et lui donne le modèle morphologique.

6.1.6 Darwish Sebawai morphologique Analyseur

Sebawai est un analyseur morphologique arabe (développé par Darwish 2005) . L'analyseur morphologique utilise des modèles orthographiques arabes pour trouver les racines. La couverture est pas parfait .cet analyseur morphologique trouve la racine avec succès 84% du temps".

6.1.7 Analyseur Morphologique Système Alkhalil

Alkhalil Morpho Sys qui est un analyseur morphologique on line pour le texte arabe standard. Alkhalil peut traiter les textes non diacrités, ainsi, il peut traiter les textes diacrités partiellement ou totalement. En fait, Alkhalil a remporté la première position, parmi les 13 systèmes morphologiques arabes à travers le monde, à un concours organisé par la Ligue arabe pour l'éducation la culture et la science (ALECSO) (2010 برنامج الخليل الصرفي).

6.1.8 Aramorph

L'analyseur morphologique Aramorph [18] segmente les unités lexicales, repère les différents composants et atteste son appartenance à la langue. Pour cela, le système est assisté par le lexique DINAAR.1 pour éviter les analyses théoriquement possibles et inexistantes dans la langue. Par la suite, l'analyseur donne une liste des traits associés à l'unité lexicale en entrée. Il offre deux types d'options. Le premier vise les traits morphosyntaxiques, le second concerne l'analyse des préfixes et suffixes.

En plus des étiquettes morphosyntaxiques, il donne en sortie d'autres informations comme la base, l'unité lexicale minimale vocalisé ou non ainsi que la forme complété supposée vocalisée ou non.

Analyser les préfixes revient à décrire ses découpages possibles et d'examiner les compositions des clitiques. Ceci amène le système à faire la distinction entre les clitiques ayant la même forme mais appartenant à des catégories syntaxiques différentes.

Exemple : "وَالتلوث" et la pollution" l'analyseur découpera le proclitique و et dira que و est celui de la liaison.

6.2 Les concordanciers

Le concordancier a pour objectif de permettre l'exploration du corpus selon les traits proposés par l'analyse morphologique et selon les informations graphiques qui se trouvent dans le texte. Il prend en entrée un texte ou un ensemble de textes et il permet :

- La construction de listes de fréquences d'items, de racines ou tout autre trait de l'analyse morphosyntaxique, par ordre alphabétique ou par ordre fréquentiel.
- La construction d'une concordance.

6.3 Racineurs

Le racineur consiste à détecter la racine d'une unité lexicale, Les algorithmes de racinisation en arabe les plus connus sont ceux de larkey [16] et Khoja [16].

7. Prétraitements nécessaires pour le tala

Afin de tenir compte de toutes les spécificités de la langue arabe et de pallier au problème de variation de représentation des caractères arabes dans les textes comme dans les requêtes, il est nécessaire de définir et d'appliquer quelques prétraitements sur le corpus de texte avant le traitement.

7.1 Encodage

La langue arabe est encodée suivant plusieurs formats d'encodage comme Unicode, ISO-8859-6, ou autres. Les textes recherchés et les requêtes peuvent être encodés différemment, afin de rendre ceux-ci incomparables. Par exemple, les documents sont représentés en Unicode (UTF-8) et les requêtes en ISO-8859-6 ou un autre encodage. Afin d'apparier les documents avec les requêtes, nous devons réutiliser des outils de conversion entre différents encodages. Ainsi, tout a été transformé en format Unicode dans ce cas.

7.1.1 L'Unicode:

Le standard Unicode est un mécanisme universel de codage de caractères. Il définit une manière cohérente de coder des textes multilingues et facilite l'échange de données textuelles. Il est obligatoire pour la plupart des protocoles de l'Internet, et mis en œuvre dans tous les systèmes d'exploitation et langages informatiques modernes. Unicode est la base de tout logiciel voulant fonctionner aux quatre coins du monde.

À l'heure actuelle, les données Unicode peuvent être codées sous trois formes principales : une forme codée sur 32 bits (UTF-32), une forme sur 16 bits (UTF-16) et une forme de 8 bits (UTF-8) conçue pour faciliter son utilisation sur les systèmes ASCII préexistants.

Unicode (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
062		ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
063	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ					
064		ف	ق	ك	ل	م	ن	ه	و	ى	ي	ً	ٌ	ٍ	َ	ُ
065	ِ	ّ	ْ													

Tableau 9 : Standard Unicode pour les caractères arabes.

7.1.2 UTF-8

Afin de satisfaire les besoins des systèmes architecturés autour de l'ASCII ou d'autres jeux de caractères à un octet, le standard Unicode définit une forme en mémoire supplémentaire : l'UTF-8. C'est une forme de mémorisation très fréquemment adoptée pour effectuer la transition des systèmes existants vers Unicode, et elle a notamment été choisie comme forme préférée pour l'internationalisation des protocoles d'Internet. UTF-8 est un codage constitué de suites d'octets; les bits de poids le plus fort d'un octet indiquent la position de celui-ci dans la suite d'octets. Autres caractéristiques importantes de l'UTF-8 :

- Conversion efficace à partir de ou vers un texte codé en UTF-16 ou en UTF-32.
- Le premier octet indique le nombre d'octets, ceci permet une analyse rapide du texte vers l'avant.
- Recherche rapide du début de tout caractère, quel que soit l'octet où l'on commence la recherche dans un flux de données, il suffit de consulter au plus quatre octets en amont pour reconnaître aisément le premier octet qui code le caractère.
- UTF-8 est un mécanisme de stockage relativement compact en termes d'octets.

7.1.3. Produits Unicode supportant l'écriture arabe

Comme nous l'avons vu, l'écriture arabe ainsi que les écritures qui sont dérivées de l'arabe de base se caractérisent notamment par leur directionnalité de droite à gauche, par leur nature cursive et par leurs signes de vocalisation qui s'ajoutent au-dessous et au-dessus des caractères.

Ces trois caractéristiques constituent en fait les problèmes majeurs que rencontrent-les technologies informatiques. Pour remédier à ces problèmes, le standard Unicode offre toute une panoplie de codes de formatage et des algorithmes permettant, par conséquent, un traitement informatique fiable de l'écriture arabe et des écritures qui en dérivent.

Tirant profit de son importance économique et technologique, le standard Unicode a été vite adopté par un grand nombre de développeurs d'applications et de constructeurs de matériel informatique. Ils exigent que tout nouveau protocole doit être en mesure d'utiliser le codage UTF-8 et que les protocoles existants qui utilisent d'autres jeux de caractères ou

même qui utilisent un jeu de caractères par défaut que l'UTF-8, doivent supporter le codage UTF-8.

Le codage Unicode est actuellement supporté par plusieurs systèmes d'exploitation, ainsi que par plusieurs langages de programmation, et par des logiciels.

7.1.4 L'encodage de corpus et requêtes

L'arabe est encodé suivant plusieurs formats d'encodage comme Unicode, ISO-8859-6, CP1256, etc. Les textes et les requêtes indexés peuvent être encodés différemment; les rendant incomparables. Par exemple, si les documents de notre corpus de test sont représentés en ISO-8859-6 et les requêtes, en Unicode (UTF-8), ou en un autre encodage comme Windows CP1256 qui est utilisé sur le Web pour représenter les textes arabes; alors nous devons réutiliser des outils de conversion entre différents encodages en utilisant des tables de l'alphabet arabe, afin d'apparier les documents avec les requêtes.

8. Conclusion

Dans ce chapitre, j'ai essayé de cerner les principales informations sur la langue arabe et leur problèmes d'analyse automatique inhérents à certains phénomènes tels que la non voyellation, l'agglutination, l'irrégularité de l'ordre des mots et l'absence de ponctuation régulière et leur morphologique et les outils que sont des travaux dirigé pour le traitement automatique de la langue arabe.

Dans le chapitre suivant je viens parler sur l'objet de m'étude l'identification automatique de mot clés et les approche qui existes pour l'analyse automatique de la langue arabe et les différentes méthodes de pondération d'un mot clés.

Chapitre 03

Identification Automatique

De Mots Clés

CHAPITRE 3
Identification automatique de mots clés

1. Introduction

L'identification automatique de mot-clé peut être officieusement décrite comme un processus par lequel une courte liste de mots-clés est extraite d'un texte beaucoup plus grand avec peu de perte d'informations. Il est utilisé pour fournir un procédé efficace pour les humains et les machines afin d'identifier rapidement le contenu et le type de textes et de documents. Un tel processus se trouve au cœur de nombreuses applications importantes telles que l'indexation dans les bases de données bibliographiques et les moteurs de recherche sur Internet, ainsi que la classification des services d'annuaire. Toutefois, en raison d'une mesure de la perte d'information diffère clairement d'une application à une autre.

2. Histoire

Le sujet de l'identification des mots clés a reçu un intérêt équitable dans le milieu de la recherche depuis l'aube précoce des ordinateurs dans les années 1960. Probablement, l'une des premières tentatives pour faire face à un sujet connexe était une enquête expérimentale menée par M. E. Maron en 1961, qui a démontré une utilisation efficace de l'analyse bayésienne pour classer les documents en fonction de leurs "mots indices" . Dans une autre étude, les réseaux bayésiens ont été utilisés pour aider à l'indexation des articles grâce à l'utilisation d'un thésaurus. Parce que la méthode d'identification a été difficile à codé, aucune formation n'a été nécessaire dans cette étude. En outre, de nombreuses techniques sophistiquées ont été développées sur la base de l'analyse de fréquence qui va de méthodes simples, telles que l'utilisation de mots de fréquence intermédiaire, à des méthodes avancées telles que tf - idf, qui prend en compte la fréquence d'un mot à travers de multiples documents [19]. En outre,

l'analyse de fréquence a également été complétée par l'utilisation d'outils linguistiques tels que les analyseurs morphologiques et syntaxiques. L'utilisation de l'analyse de la fréquence et des outils linguistiques est probablement l'approche la plus répandue de nos jours [19].

3. Identification automatique des mots-clés

L'identification automatique d'un mot clé est l'opération d'extraction de l'information sous forme d'expressions (mots, termes) clés à partir de l'analyse des documents d'un domaine de recherche définie qui consiste à extraire les aspects importants, d'exploiter des outils dédiés à la constitution de terminologies structurées. Les termes obtenus sont aussi été triés et filtrés à l'aide de leur position, de méthodes de pondération statistiques et de critères linguistiques.

3.1 Définition d'un Mot-clé

Le terme mot-clef (les orthographes suivantes sont également correctes : mot-clé, mot clé, mot clef, en anglais, *keywords*) désigne de manière générale un mot qui a une importance particulière. Mot essentiel d'une phrase, d'un vers ou d'un texte, d'un sujet de rédaction ou de dissertation, les mots peuvent être des simples mots ou bien des termes (multi-mots).

Dans le domaine des sciences de l'information et des bibliothèques, les mots-clefs sont des mots utilisés lors des recherches d'informations.

Dans le domaine du référencement dans les moteurs de recherche, le terme "mot clé" désigne une expression, composée d'un ou de plusieurs mots, qui est tapée par l'internaute dans la zone de recherche du moteur de recherche. Ce dernier affiche alors une liste de liens hypertextes qui pointent vers les sites web en rapport avec l'expression saisie. Le travail du référenceur est de trouver les mots clés relatifs à un sujet donné afin d'optimiser les pages web des sites dont il a la charge. Son objectif est d'augmenter la visibilité des pages dans les moteurs.

En programmation informatique, un mot-clef est un mot qui appartient à un vocabulaire restreint de mots qui ont un sens particulier dans un langage informatique. On distingue les mots-clefs des identificateurs [20].

Dans les articles scientifiques, une liste de mots-clés est définit par l'auteur lui-même, cette liste représente les idées générales abordées dans le sujet de l'article. L'exemple suivant, montre une liste de mots-clés pour un article d'une revue économique :

الكلمات المفتاحية: مقياس التدريب، الموارد البشرية، التنوع، الشمولية، الاحتياجات التدريبية

Mots clés: formation à l'échelle, les ressources humaines, la diversité, l'inclusion, les besoins de formation.

Key words: scale training, human resources, diversity, inclusiveness, training needs.

Figure 08 : Un exemple des mots clés identifié par un auteur [21].

3.2 Les méthodes d'extraction automatique de mots-clés

L'extraction de mots-clés est une tâche qui consiste à analyser un document et à en extraire les aspects importants. Alors que les méthodes de résumé automatique utilisent des phrases pour construire une vision synthétique du document, l'extraction de termes-clés se focalise sur les unités textuelles qui composent ces phrases. Un ensemble de termes-clés peut donc être perçu comme un résumé dont les points clés sont exprimés sans liaisons entre eux. Les unités textuelles sur lesquelles travaillent les systèmes d'extraction automatique de termes-clés sont appelées termes candidats. Ces derniers sont des mots ou des multi-mots (phrasèmes) pouvant être promus au statut de terme-clé.

L'extraction de mots candidats est une étape préliminaire de l'extraction de mots-clés, que ce soit pour les méthodes non-supervisées ou supervisées. Cette étape est importante, car si certains mots-clés du document analysé ne sont pas présents dans l'ensemble des termes candidats, alors ceux-ci ne pourront pas être extrait. Hulth [23] étudie trois méthodes d'extraction des termes candidats. L'une consiste à extraire les chunks nominaux, tandis que les deux autres extraient tous les n-grammes et les filtrent, soit pour retirer les termes contenant des mots outils dans le premier cas, soit pour ne retenir que les termes respectant certains patrons syntaxiques dans le second cas (usage des parties du discours). Dans ses

expériences Hulth [23] montre que l'extraction de mots-clés à partir de n-grammes filtrés avec les mots outils donne les meilleurs résultats parmi les trois méthodes qu'elle propose.

Les travaux de Hulth [23] sont évalués avec un corpus dont les documents sont des résumés d'articles scientifiques. Cependant, dans d'autres domaines tels que la biomédecine, la nature des termes à extraire n'est pas la même. En effet, ce sont les acronymes et les entités nommées (noms de protéines par exemple) qu'il est nécessaire d'extraire en tant que termes-clés [24]. Pour cela, *l'extraction de termes candidats est spécifique au domaine d'application.* Les méthodes d'extraction de termes-clés présentées dans cet chapitre traitent des documents supposés sans spécificités particulières, les méthodes d'extraction de termes candidats sont donc les mêmes que celles expérimentées par Hulth [23], mais il est envisageable de les adapter à des domaines présentant des spécificités particulières.

Utilisés avec les méthodes non-supervisées, les termes candidats sont ordonnés selon un score d'importance obtenu soit à partir d'eux-mêmes, soit à partir de l'importance des mots qui les composent. Si une méthode s'appuie uniquement sur les mots, alors le score d'un terme candidat est généralement calculé en faisant la somme des mots qui le composent. Cependant, ceci n'est pas toujours juste, c'est donc un inconvénient important des méthodes travaillant sur les mots pour extraire les termes-clés. En effet, la sommation peut privilégier des termes qui contiennent beaucoup de mots non-importants vis-à-vis de termes contenant très peu de mots, mais importants. Utilisés dans les méthodes supervisées, les termes candidats sont classés en tant que termes-clés ou non termes-clés grâce à des méthodes de classification.

3.2.1 Méthodes non-supervisées

Les méthodes non-supervisées d'extraction de mots-clés ont la particularité de s'abstraire du domaine et de la langue des documents à analyser. Cette abstraction est due au fait que les termes candidats sont analysés avec des règles simples déduites à partir de traits statistiques issus seulement du texte analysé, ou bien d'un corpus de référence non annoté. De nombreuses approches sont proposées. Certaines se fondent uniquement sur des statistiques alors que d'autres les combinent avec des représentations plus complexes des documents. Ces représentations peuvent aller de groupes de mots sémantiquement similaires à des graphes

dont les nœuds sont des unités textuelles (mots, expressions, phrases, etc.) liées par des relations de recommandation.

3.2.1.1 Approches statistiques

Plusieurs approches cherchent à définir ce qu'est un mot-clé en s'appuyant sur certains traits statistiques et en étudiant leur rapport avec la notion d'importance d'un terme candidat. Plus un mot candidat est jugé important vis-à-vis du document analysé, plus celui-ci est pertinent en tant que mot-clé. TF-IDF (cf. équation 1) de Jones [25] et Likey (cf. équation 2) de [26] sont deux méthodes qui comparent le comportement d'un terme candidat dans le document analysé avec son comportement dans une collection de documents (corpus de référence). L'objectif est de trouver les termes candidats dont le comportement dans le document varie positivement comparé à leur comportement global dans la collection. Dans les deux méthodes ceci s'exprime par le fait qu'un terme à une forte importance vis-à-vis du document analysé s'il y est très présent, alors qu'il ne l'est pas dans le reste de la collection.

$$\mathrm{TF-IDF(terme)} = \mathrm{TF\ (terme)} \times \log\left(\frac{N}{DF(\mathrm{terme})}\right) \quad (1)$$

$$Likey\ (\mathrm{terme}) = \frac{\mathrm{rang}_{\ \mathrm{document}}\ (\mathrm{terme})}{\mathrm{rang}_{\ \mathrm{corpus}}(\mathrm{terme})} \quad (2)$$

Dans TF-IDF :

TF : représente le nombre d'occurrences d'un mot dans le document analysé.

DF : représente le nombre de documents dans lequel il est présent.

N : étant le nombre total de documents. Plus le score TF-IDF d'un terme candidat est élevé, plus celui-ci est important dans le document analysé.

Dans Likey :

Le rang d'un terme candidat dans le document et dans le corpus est obtenu à partir de son nombre d'occurrences, respectivement dans le document et dans le corpus de référence. Plus

le rapport entre ces deux rangs est faible, plus le terme candidat évalué est important dans le document analysé.

Okapi (ou BM25) est une mesure alternative à TF-IDF. En Recherche d'Information (RI), celle-ci est plus utilisée que le TF-IDF. Bien que l'extraction automatique de termes-clés soit une discipline à la frontière entre le TAL et la RI, la méthode de pondération Okapi n'a, à notre connaissance, pas été appliquée pour l'extraction de termes-clés. Dans l'article de Claveau [27], Okapi est décrit comme un TF-IDF prenant mieux en compte la longueur des documents. Cette dernière est utilisée pour normaliser le TF (qui devient TF_{BM25}) :

$$\text{Okapi(terme)} = TF_{BM25}(\text{terme }) \times \log\left(\frac{N - DF(\text{terme }) + 0{,}5}{DF(\text{terme }) + 0{,}5}\right) \qquad (3)$$

$$\text{TF}_{BM25} = \frac{\text{TF(terme)} \times (k_1 + 1)}{\text{TF(terme)} + k_1 \times \left(1 - \text{b} + \text{b} \times \frac{\text{DL}}{DL_{\text{moyenne}}}\right)} \qquad (4)$$

Dans la formule (4) :

k1 et **b** : sont des constantes fixées à 2 et 0, 75 respectivement.

DL : représente la longueur du document analysé.

DL moyenne : la longueur moyenne des documents de la collection utilisée.

Barker et Cornacchia [28] estiment que les grands phrasèmes sont plus informatifs et qu'ils doivent être privilégiés. Pour cela, leur approche est très simple : plus un groupe nominal est long et fréquent dans le document analysé, plus il est jugé pertinent en tant que terme-clé de ce document. Cependant, pour éviter la répétition dans le texte, les auteurs des documents utilisent les mêmes expressions sous des formes alternatives (plus courtes, par exemple). La fréquence d'une expression ne reflète donc pas forcément sa fréquence réelle d'utilisation, car celle-ci est répartie dans les différentes alternatives. De ce fait, Barker et Cornacchia [28] repèrent dans les groupes nominaux la tête nominale et utilisent en plus la fréquence de celle-ci. Tomokiyo et Hurst [29] tentent de vérifier deux propriétés, en utilisant

des modèles de langue uni-grammes et n-grammes et en calculant leur divergence (Kullback-Leibler). Les deux propriétés qu'ils tentent de vérifier sont les suivantes :

- La grammaticalité : un mot-clé doit être bien formé syntaxiquement.
- L'informatives : un mot-clé doit capturer au moins une des idées essentielles exprimées dans le document analysé.

Pour un terme candidat donné, plus sa probabilité en passant du modèle uni-gramme généré à partir du document vers le modèle n-gramme généré à partir du même document augmente, plus il respecte la propriété de grammaticalité. De même, plus sa probabilité en passant du modèle n-gramme généré à partir d'un corpus de référence vers le modèle n-gramme généré à partir du document analysé augmente, plus le terme candidat n'est informatif. La méthode que propose Ding et al. [30] utilise TF-IDF comme indicateur de l'importance d'un terme-clé. Dans un ensemble, cette importance doit être maximisée pour chaque terme-clé, mais les auteurs estiment que ceci n'est pas suffisant. Comme Tomokiyo et Hurst [28], ils définissent deux propriétés qui doivent être respectées :

- La couverture : un ensemble de mots-clés doit couvrir l'intégralité des sujets abordés dans le document représenté.
- La cohérence : les mots-clés doivent être cohérents entre eux.

La propriété de couverture est évaluée avec le modèle Latent Dirichlet Allocation (LDA) qui donne la probabilité d'un terme candidat sachant un sujet. La cohérence est évaluée pour chaque paire de termes-clés de l'ensemble avec la mesure d'information mutuelle. Ces deux propriétés sont définies comme contraintes que les auteurs utilisent avec une méthode de programmation par les entiers (technique d'optimisation), la maximisation de la pertinence de chaque terme-clé étant l'objectif à atteindre. Les traits statistiques utilisés dans les méthodes précédentes sont uniquement utilisés pour déterminer un score de pertinence des termes candidats en tant que mots-clés.

Une donnée statistique non citée précédemment, mais pourtant récurrente dans les méthodes d'extraction de termes-clés, est la fréquence de co-occurrences entre deux phrasèmes (termes). Deux phrasèmes co-occurrent s'ils apparaissent ensemble dans le même

contexte. La co-occurrence peut être calculée de manière stricte (les phrasèmes doivent être côte-à-côte) ou bien dans une fenêtre de mots. Compter le nombre de co-occurrences entre deux termes permet d'estimer s'ils sont sémantiquement liés ou non. Ce lien sémantique à lui seul ne peut pas servir à extraire des termes-clés, mais il permet de mieux organiser les termes d'un document pour affiner l'extraction [31].

3.2.1.2 Approches par regroupement

L'objectif des approches par regroupement est de définir des groupes dont les unités textuelles partagent une ou plusieurs caractéristiques communes. Ainsi, lorsque des termes-clés sont extraits à partir de chaque groupe, cela permet de mieux couvrir le document analysé selon les caractéristiques utilisées.

Dans la méthode de Matsuo et Ishizuka [32], ce sont les termes (phrasèmes) qui sont regroupés. Parmi ceux-ci, seuls les plus fréquents sont concernés par le regroupement. Celui-ci s'effectue en fonction du lien sémantique 4 entre les termes. Après le regroupement, la méthode consiste à comparer les termes candidats du document analysé avec les groupes de termes fréquents, en faisant l'hypothèse qu'un terme candidat qui co-occurre plus que selon toute probabilité avec les termes fréquents d'un ou plusieurs groupes est plus vraisemblablement un terme-clé.

Dans l'algorithme KeyCluster, Liu et al. [33] utilisent aussi un regroupement sémantique, mais dans leur cas ils considèrent les mots du document analysé et ils excluent les mots outils. Dans chaque groupe sémantique, le mot qui est le plus proche du centroïde est sélectionné comme mot de référence. L'ensemble des mots de référence est ensuite utilisé pour filtrer les termes candidats en ne considérant comme termes-clés que ceux qui contiennent au moins un mot de référence (tous les mots de référence devant être utilisés dans au moins un terme-clé).

3.2.1.3 Approches à base de graphe

Les approches à base de graphe consistent à représenter le contenu d'un document sous la forme d'un graphe. La méthodologie appliquée est issue de PageRank , un algorithme d'ordonnancement de pages Web (nœuds du graphe) grâce aux liens de recommandation qui

existent entre elles (arcs du graphe). TextRank [35]et SingleRank [36] sont les deux adaptations de base de PageRank pour l'extraction automatique de mots-clés. Dans celles-ci, les pages Web sont remplacées par des unités textuelles dont la granularité est le mot et un arc est créé entre deux nœuds si les mots qu'ils représentent co-occurrent dans une fenêtre de mots donnée.

Le graphe est noté G= (N,A), où **N** est l'ensemble des nœuds du graphe et où **A** est l'ensemble de ses arcs entrants et sortant : $A_{\text{entrant}} \cup A_{sortant}$. Pour chaque nœud du graphe, un score est calculé par un processus itératif destiné à simuler la notion de recommandation d'une unité textuelle par d'autres (cf. équation 5). Ce score à chaque nœud ni permet d'ordonner les mots par degré d'importance dans le document analysé. La liste ordonnée des mots peut ensuite être utilisée pour extraire les termes-clés.

$$S(n_i) = (1-\lambda) + \lambda \times \sum_{n_j \in A_{\text{entrant}}(n_i)} \frac{p_{j,i} \times S(n_j)}{\sum_{n_k \in A_{\text{sortant}}(n_j)} p_{j,k}} \qquad (5)$$

λ est un facteur d'atténuation qui peut être considéré ici comme la probabilité pour que le nœud ni soit atteint par recommandation.

pj,i représente le poids de l'arc allant du nœud **nj** vers le nœud **ni** , soit le nombre de cooccurrences entre les deux mots i et j .

Dans leurs travaux, Wan et Xiao [37] s'intéressent à l'ajout d'informations dans le graphe grâce à des documents similaires (voisins) et aux relations de co-occurrences qu'ils possèdent (ExpandRank). L'objectif est de faire mieux ressortir les mots importants du graphe en ajoutant de nouveaux liens de recommandation ou bien en renforçant ceux qui existent déjà. L'usage de documents similaires peut cependant ajouter ou renforcer des liens qui ne devraient pas l'être. Pour éviter cela, les auteurs réduisent l'impact des documents voisins en utilisant leur degré de similarité avec le document analysé. Une alternative à ExpandRank, CollabRank, également proposée par [37], fonctionne de la même manière, mais certains choix des auteurs rendent impossible l'usage du degré de similarité pour réduire l'impact des documents voisins. Les résultats moins concluants de CollabRank tendent à confirmer l'importance de l'usage du degré de similarité. Dans l'optique d'améliorer encore

TextRank/SingleRank, [36] proposent une méthode qui cherche cette fois-ci à augmenter la couverture de l'ensemble des termes-clés extraits dans le document analysé (Topical Page Rank). Pour ce faire, ils tentent d'affiner le rang d'importance des mots dans le document en tenant compte de leur rang dans chaque sujet abordé. Le rang d'un mot pour un sujet est obtenu en intégrant à son score PageRank la probabilité qu'il appartienne au sujet (cf. équation 6). Le rang global d'un terme candidat est ensuite obtenu en fusionnant ses rangs pour chaque sujet.

$$S_{\text{sujet}}(N_i) = (1-\lambda) \times p(\text{sujet}|i) + \lambda \times \sum_{N_j \in A_{\text{entrant}}(N_i)} \frac{p_{j,i} \times S(N_j)}{\sum_{N_k \in A_{\text{sortant}}(N_j)} p_{j,k}} \qquad (6)$$

Les approches à bases de graphe présentées ci-dessus effectuent toutes un ordonnancement des mots du document analysé selon leur importance dans celui-ci. Pour extraire les termes-clés il est donc nécessaire d'effectuer du travail supplémentaire à partir de la liste ordonnée de mots. Dans la méthode TextRank, les k mots les plus importants sont sélectionnés et retournés (après que ceux apparaissant en collocation dans le document aient été concaténés). La technique utilisée dans les autres méthodes consiste à ordonner les termes candidats en fonction de la somme du score des mots qui les composent. Cependant, puisque l'un des avantages du graphe est que les nœuds peuvent avoir une granularité contrôlée, Liang et al. [38] décident d'utiliser des mots et des multi-mots au lieu de simples mots et de tirer profit de traits supplémentaires, la taille du terme ou encore sa première position dans le document analysé.

3.2.2 Méthodes supervisées

Les méthodes supervisées sont des méthodes capables d'apprendre à réaliser une tâche particulière, soit ici l'extraction de termes-clés. L'apprentissage se fait grâce à un corpus dont les documents sont annotés en mots-clés. L'annotation permet d'extraire les exemples et les contres exemples dont les traits statistiques et/ou linguistiques servent à apprendre une classification binaire. La classification binaire consiste à indiquer si un terme candidat est un mot-clé ou non. De nombreux algorithmes d'apprentissage sont utilisés dans divers domaines. Ils peuvent potentiellement s'adapter à n'importe quelle tâche, dont celle de l'extraction automatique de mots-clés. Les algorithmes utilisés pour celle-ci construisent des modèles probabilistes, des arbres de décision, des Séparateurs à Large Marge (SVM) ou encore des réseaux de neurones.

KEA [39] est une méthode qui utilise une classification naïve bayésienne pour attribuer un score de vraisemblance à chaque terme candidat, le but étant d'indiquer s'ils sont des termes-clés ou non . Witten et al[39] utilisent trois distributions conditionnelles apprises à partir du corpus d'apprentissage. La première correspond à la probabilité pour que chaque terme candidat soit étiqueté oui (mot-clé) ou non (non mot-clé). Les deux autres correspondent à deux différents traits qui sont le poids TF-IDF du terme candidat et sa première position dans le document :

$$\mathrm{P}\,(\text{terme}) = \frac{\mathrm{P_{oui}}(\text{terme})}{\mathrm{P_{oui}}(\text{terme}) + \mathrm{P_{non}}\,(\text{terme})} \qquad (7)$$

$$\mathrm{P_{oui}}(\text{terme}) = \mathrm{P}\,(\text{terme}|\text{oui}) \times \prod_{\text{trait}\in\{\text{TF-IDF,position}\}} P_{\text{trait}}\,(\text{trait}(\text{terme})|\text{oui})$$

$$\mathrm{P_{non}}(\text{terme}) = \mathrm{P}\,(\text{terme}|\text{non}) \times \prod_{\text{trait}\in\{\text{TF-IDF,position}\}} P_{\text{trait}}\,(\text{trait}(\text{terme})|\text{non})$$

L'un des avantages de la classification naïve bayésienne est que chaque distribution est supposée indépendante. L'ajout de nouveaux traits dans la méthode KEA est donc très aisé. Parmi les variantes de KEA proposées, [40] ajoutent un troisième trait : le nombre de

fois que le terme candidat est un terme-clé dans le corpus d'apprentissage. L'ajout de ce trait permet d'améliorer les performances de la version originale de KEA, mais uniquement lorsque la quantité de données d'apprentissage est très importante. Une autre amélioration de KEA, proposée par [41] tente d'augmenter la cohérence entre les termes candidats les mieux classés. Pour ce faire, une première étape de classification est effectuée avec la méthode originale. Cette première étape permet d'obtenir un premier classement des termes candidats selon leur score de vraisemblance.

Ensuite, de nouveaux traits sont ajoutés et une nouvelle étape de classification est lancée. Les nouveaux traits ont pour but d'augmenter le score de vraisemblance des termes candidats ayant un fort lien sémantique avec certains des termes les mieux classés après la première étape. Enfin, Nguyen et Kan [42] proposent l'ajout des informations concernant la structure des documents. En effet, certaines sections telles que l'introduction et la conclusion dans les articles scientifiques sont plus susceptibles de contenir des mots-clés qu'une section présentant des résultats expérimentaux, par exemple. Dans leur version modifiée de KEA, ils proposent aussi l'usage de traits linguistiques tels que les parties du discours qui ont prouvées jouer un rôle non-négligeable pour l'extraction des mots-clés .

En même temps que KEA [39] et [41] met au point l'algorithme génétique GenEx. GenEx est constitué de deux composants. Le premier composant, le géniteur, sert à apprendre des paramètres lors de la phase d'apprentissage. Ces paramètres sont utilisés par le second composant, l'extracteur, pour donner un score d'importance à chaque terme candidat. Plus les paramètres sont optimaux, meilleure est la classification des termes. Pour ce faire, les paramètres sont représentés sous la forme de bits qui constituent une population d'individus que le géniteur fait évoluer jusqu'à obtenir un état stable correspondant aux paramètres optimaux.

Dans son article présentant GenEx, [41] discute une autre méthode pour l'extraction de termes-clés. Cette méthode utilise de nombreux traits qui servent à entraîner 50 arbres de décision C4.5 (technique de Random Forest). Dans un arbre de décision, chaque branche représente un test sur l'un des traits d'un terme candidat. Les tests permettent un routage du mot candidat vers la feuille de l'arbre qui détermine sa classe. Grâce à la technique de Random Forest, soit l'usage de plusieurs arbres entraînés sur un échantillon différent du

corpus d'apprentissage, l'extraction automatique de mot-clé est réduite à un vote de chaque arbre pour chaque terme candidat. Cela permet un classement des mots candidats en fonction de leur nombre de votes positifs. Les mots-clés extraits correspondent aux mots candidats les mieux classés.

La même année que les travaux de Hulth [23] sur le bien-fondé d'utiliser des traits linguistiques pour l'extraction automatique de mots-clés, Sujian et al. [42] proposent une méthode utilisant un modèle d'entropie maximale (cf. équation 8) dont l'un des traits repose sur les parties du discours des mots qui composent les termes candidats. Un modèle de maximum d'entropie consiste à trouver parmi plusieurs distributions, une pour chaque trait, laquelle a la plus forte entropie. La distribution ayant la plus forte entropie est par définition celle qui contient le moins d'informations, ce qui la rend de ce fait moins arbitraire pour l'extraction des mots-clés.

$$\text{score}(\text{terme}) = \frac{P(\text{oui}|\text{terme})}{P(\text{non}|\text{terme})} \qquad (8)$$

$$P(\text{classe}|\text{terme}) = \frac{\exp\left(\sum_{\text{trait}} \alpha_{\text{trait}} \times \text{trait}(\text{terme}, \text{classe})\right)}{\sum_{c \in \{\text{oui,non}\}} \exp\left(\sum_{\text{trait}} \alpha_{\text{trait}} \times \text{trait}(\text{terme}, c)\right)}$$

Le paramètre **α** trait définit l'importance du trait auquel il est associé. Les Séparateurs à Large Marge sont aussi des classifieurs utilisés par les méthodes d'extraction automatique de mots-clés. Ils exploitent divers traits afin de projeter des exemples et des contre-exemples sur un plan, puis ils cherchent l'hyperplan qui les sépare. Cet hyperplan sert ensuite dans l'analyse de nouvelles données.

Dans le contexte de l'extraction de mots-clés, les exemples sont les mots-clés et les contre-exemples sont les termes candidats qui ne sont pas des mots-clés. Ce mode de fonctionnement des SVM est utilisé par Zhang et al. [43], mais un autre type de SVM est plus largement utilisé dans les méthodes supervisées d'extraction de termes-clés. Il s'agit de SVM qui utilise de multiples marges représentant des rangs. Ces classifieurs permettent donc

d'ordonnancer les mots-clés lors de leur extraction . La méthode KeyWE de Eichler et Neumann [44] utilise ce type de SVM avec le trait TF-IDF ainsi qu'un trait booléen ayant la valeur vraie si le terme candidat apparaît dans un titre d'un article Wikipédia (un mot candidat apparaissant dans le titre d'un article de Wikipédia a une plus forte probabilité d'être un mot-clé). L'ordonnancement des termes candidats par le SVM permet ensuite de contrôler le nombre de mots-clés à extraire (choix des k termes candidats les mieux classés).

Tout comme [41], et [45] utilisent eux aussi une forêt d'arbres C4.5 dans leur méthode d'extraction de mots-clés. Ils utilisent des traits classiques et leur contribution se situe au niveau de l'utilisation d'un trait calculé à partir de chaînes lexicales.

Une autre méthode pour l'extraction automatique de mots-clés consiste à utiliser un perceptron multi-couches [46]. Un perceptron multi-couches est un réseau de neurones constitué d'au moins trois couches, chaque couche étant composée de neurones. Dans les deux couches extrêmes les neurones représentent respectivement les entrées et les sorties. Les couches centrales sont des couches cachées qui permettent d'acheminer les valeurs des entrées vers les sorties, où de nouvelles valeurs sont obtenues grâce à la pondération des transitions d'un neurone d'une couche vers un neurone de la couche suivante. Les entrées correspondent aux traits d'un terme candidat (ici TF-IDF, la position, la taille, etc.) et les sorties représentent les classes qu'il peut prendre (mot-clé ou non mot-clé).

Dans leurs travaux, [37] proposent une méthode d'extraction de termes-clés basée sur un modèle génératif. Leur méthode est très différente de celle de Witten et al. [39] puisqu'ils décident d'utiliser une approche de traduction automatique. L'usage original de cette approche est justifié par le fait qu'un ensemble de mots-clés doit décrire de manière synthétique le document. Leur hypothèse est donc qu'un ensemble de mots-clés est une traduction d'un document dans un autre langage. Le modèle est appris à partir de paires de traductions dont l'un des termes est issu des titres ou des résumés des documents du corpus d'apprentissage et dont l'autre terme est issu des corps de ces mêmes documents. Les titres et les résumés sont utilisés comme langage synthétique et les corps des documents comme le langage naturel de ceux-ci. **[22]**

4. Evaluation sur la liste de mot clés

4.1 Evaluation manuelle

L'évaluation peut être faite manuellement en faisant appel à des annotations manuelles, mais cela retourne généralement un état d'accords inter-annotateurs très faibles. D'autres prennent le parti d'accoler bout à bout des séquences appartenant à des textes différents ; les ruptures lexicales sont alors les ruptures entre textes. L'évaluation manuelle est généralement très couteuses des évaluations automatique ou semi-automatique sont considérées comme des bonnes alternatives.

4.2 Evaluation semi-automatique

Nous pouvons faire une évaluation semi-automatique en comparant les résultats produits automatiquement (mot clés) par le système et d'autres produits manuellement par un expert humain, dans ce cas nous faisons appel par exemple à la métrique F-mesure qui calcule les scores de rappel et de précision.

Cependant, les scores habituels de précision et de rappel ne sont pas toujours adaptés pour évaluer un système de détection de de mot clés.

4.3 Evaluation automatique

Pour évaluer de tel système de façon automatique, d'autres scores ont été proposés, dont les plus usités sont les mesures Pk et WindowDiff . La mesure WindowDiff consiste à calculer la différence du nombre de ruptures dans une fenêtre glissante. Les mesures ROUGE, pour *Recall-Oriented Understudy for Gisty Evaluation*, ont été introduites par Lin .Ces mesures sont fondées sur la comparaison de ngrammes entre un ou plusieurs résumés de référence et un résumé à évaluer. Il n'existe pas un unique résumé de référence, et il est donc essentiel de comparer les résumés automatiques à plusieurs résumés de référence établis manuellement afin d'obtenir des mesures plus précises de la qualité des résumés. Ces mesures nécessitent donc la rédaction
de résumés de référence par un ou plusieurs experts au préalable de la mesure de qualité du résumé.[47]

5. Les travaux relatif

A partir de notre recherche sur le web, nous n'avons pas trouvé des travaux indépendants qui traitent le problème d'identification de mots clés dans les textes arabes, par ailleurs, il existe des travaux qui intègrent cette tâche dans leur processus de traitement. Les travaux recensés concernent particulièrement les sujets suivants :

- **Etude sémantique des mots-clés et des marqueurs lexicaux stables dans un corpus technique :** Ce travail présente les résultats d'une analyse sémantique quantitative des unités lexicales spécifiques dans un corpus technique, relevant du domaine des machines-outils pour l'usinage des métaux. L'étude vise à vérifier si et dans quelle mesure les mots-clés du corpus technique sont monosémiques.
- **Identification d'opinions dans les textes arabes en utilisant les ontologies :** Les opinions explicites peuvent être extraites par projection directe des concepts ontologiques sur le texte. Cependant, les opinions implicites ont besoin d'une exploration profonde de la couche sémantique de l'ontologie, en exploitant les relations entre les concepts, les individus et les attributs.
- **Résumé automatique :** Résumer un texte consiste à réduire ce texte en un nombre limité de mots. Le texte ainsi réduit doit rester fidèle aux informations et idées du texte original, et dans la mesure du possible rendre compte du style et de l'intention de l'auteur. Cette discipline, quoique très ancienne, est mal formalisée. Le processus de résumé est en effet dépendant à la fois du type de texte à résumer et de l'utilisation qui en sera faite. Ainsi, un résumé de type rapport d'activités sera dans la forme comme dans le fond radicalement différent d'un résumé d'une œuvre littéraire, d'un résumé d'ouvrage scientifique, d'un résumé de dépêches ou d'une revue de presse.

6. Conclusion

L'identification automatique des mots-clés est donc une tache très importante qui permet la valorisation d'un document (représentation synthétique, mise en évidence des points clés dans le document, etc.) et facilite l'accès au contenu pertinent dans les textes.

Dans ce chapitre on a essayé de recenser les différentes définitions possibles de terme mot-clé, on a ainsi exploré les différentes méthodes et techniques utilisées dans l'identification automatique de mots-clés dans les textes écrits.

Dans ce qui suit, on concentre particulièrement sur l'identification de mots-clés dans les textes arabes et les possibilités d'utiliser les techniques déjà explorées, ces techniques ont prouvés de bon résultats dans les langues latines alors pourquoi pas l'arabe.

La partie qui suit sera donc consacrée à la conception proprement dite du système d'identification, cette partie sera marquée par le choix d'un corpus de textes écrits en arabe standard moderne, à savoir, des articles publiés dans une revue économique ainsi le choix d'une ou de combinaison de techniques de détection automatique de mots-clés.

Chapitre 04

Conception de Système

CHAPITRE 4
Conception du système

1. Introduction

Par définition, l'identification automatique de mot-clé est la tâche d'identifier un petit ensemble de mots, de phrases clés ou des segments clés d'un document qui permettent de décrire la signification du document . Un système d'identification est donc jugé efficace s'il permet d'extraire ou abstraire une liste d'expressions langagières qui représente globalement (assure une couverture thématique) le sens du document (texte ou article).

Notre système d'identification est basé principalement sur une méthode d'extraction à partir de textes arabes. Les textes étudiés sont des articles économiques, une catégorie de textes écrits en arabe standard moderne.

Afin d'extraire automatiquement des mots clé à partir d'un article, notre système KWAI (KeyWords Arabic Identificator) suit tout un processus qui commence par une étape de segmentation et filtrage en passant par les traitements linguistiques (normalisation et lemmatisation) ensuite la pondération des termes et enfin la sélection des mots clés.

2. Caractéristiques du corpus

Notre corpus d'étude, de test et d'évaluation est formé d'une collection d'articles économiques publiés dans la revue économique internationale « Economie contemporaine » (الاقتصاد المعاصر) éditée par la faculté des sciences économiques et de gestion de l'université Djilali Bounaâma – Khemis Miliana, Les articles sont écrits en arabe standard moderne et ils sont généralement non voyellé. Le corpus est constitué d'environ 50 articles d'une longueur moyenne de 15 pages Word. Le nombre moyen de mots par article est environ 2000 mots.

D'après une exploration de la collection d'articles, nous avons tiré les traits linguistiques suivants :

- Une liste de mots clés est associée à chaque article, cette liste est définit par le ou les auteurs de l'article : c'est la raison pour laquelle nous avons choisi ce type de texte, en effet, la rareté ou même l'absence de corpus arabes annotés avec mots clés nous a poussé à choisir un type d'article annoté par l'auteur lui-même, cette annotation nous permettra non seulement de tirer quelques informations linguistiques sur les mots clés mais aussi d'évaluer notre système par rapport à la liste des mots clés référence défini par l'auteur.
- La liste des mots clés définie par l'auteur est composée généralement de syntagmes nominaux (noms, adjectifs, adverbes), l'utilisation des verbes est très rare ou même inexistantes : La connaissance de la catégorie grammaticale des mots est en effet indispensable est importante pour un système de détection automatique de mots clés, pour cette raison nous devions choisir un analyseur qui permet l'étiquetage grammatical (POS Tagging). Nous avons choisi un analyseur morphosyntaxique très puissant : AlKhalil Morpho Sys, un système qui au plus des taches de lemmatisation (légère : Stemming, et profonde : Calcul de racines), fourni des indices supplémentaires sur le type grammatical des mots et bien d'autres informations (voyellation, schèmes,..).
- L'auteur utilise généralement le même vocabulaire ou un vocabulaire proche pour discuter et détailler les mêmes idées, cette caractéristique est en réalité commune entre les textes traitant un domaine particulier (économie, biologie, médecine,..), à la différence des textes traitant des sujets ouvert ou le vocabulaire est généralement diversifié. Nous pouvons dire alors que l'utilisation d'une lemmatisation profonde (calcul des racines) pour calculer la distribution des occurrences des termes est vivement conseillée, par contre le choix d'une lemmatisation légère (appelée parfois stemming[1]) est vraiment subjective, en effet, la lemmatisation légère se base essentiellement sur la détermination des schèmes et produit des formes correspondantes aux schèmes, ce qui peut fausser le calcul des occurrences d'un même terme. Par exemple, les deux mots الاستخراج et استخرج seront considérés comme des mots différents par rapport à leurs schèmes استفعال et استفعل alors que ce sont

[1] Le stemming est considéré dans quelques analyseurs comme calcul de racines

des occurrences d'un même mot. Dans le même exemple, les deux occurrences seront considérées comme une même racine خرج. Même si l'utilisation des racines au lieu des schèmes lexicaux peut fausser les sens et donc les résultats, son utilisation est acceptable dans des textes caractérisés par des vocabulaires proches, tel est le cas dans notre corpus. En plus, la racine regroupe le sens profond de tous les dérivés, par exemple, à partir de la racine (عبر) nous pouvons dériver les mots (عِبرة ، عبارة، عَبرة، عبير ،..) qui possèdent un sens profond commun qui est (الانتقال/transfert).

- Pour expliquer un topic, l'auteur débute ses paragraphes avec les mots les plus saillants, ces mots sont généralement ceux cités dans la liste des mots clés : l'utilisation de la position des termes dans le processus d'identification donne surement un plus. En effet, les mots qui se situent dans l'entête d'un paragraphe sont considérés plus importants que les autres et ils peuvent êtres des mots clés.
- Le segment textuel qui possèdent un sens complet dans un texte est le paragraphe : la segmentation du texte en paragraphes au lieu des phrases est majoritairement favorisée et plus simple, en effet, la détection des frontières des phrases et généralement difficiles car la ponctuation ou les prépositions ne jouent pas toujours le rôle de frontières, une fausse segmentation peut donc fausser complètement les résultats.

3. L'Approche proposée

L'approche que nous proposons est une approche extractive, hybride et semi-supervisée pour l'identification automatique des mots clés dans des articles économiques.

Notre approche est qualifiée hybride par rapport à la combinaison de plusieurs critères statistiques et d'autres linguistiques pour la détection des mots clés, les critères statistiques sont la distribution des occurrences des termes (fréquences) et la position des termes dans les paragraphes du texte. L'indice linguistique est la catégorie grammaticale (type) des termes.

L'importance des termes est quantifiée en utilisant la métrique OKAPI qui est basée sur le calcul des fréquences d'apparition et la distribution des termes.

Un score final est attribué à chaque terme du texte en fonction de son poids OKAPI et de son poids par rapport à sa position relative dans le paragraphe.

Les termes sont classés en fonction de leurs scores finaux, ensuite une étape de sélection selon le type (catégorie grammaticale) est faite pour sélectionner les mots clés candidats qui sont pratiquement des syntagmes (unités) nominaux d'où notre approche est qualifiée semi-supervisé.

Les mots clés candidats cités dans des positions proches sont considérés comme une seule expression.

Enfin, le nombre final de mots clés est déterminé selon un taux de compression (nombre maximal de mots clés) qui peut être défini par l'utilisateur du système. Le nombre moyen de mots clés dans les articles de notre corpus est 5, cette valeur est considérée comme valeur par défaut.

4. Architecture de système

KWAI (KeyWords Arabic Identificator) est un système d'identification automatique de mot clés basé principalement sur des techniques d'extraction. La mise en œuvre fonctionnelle de KWAI est représentée à la figure 09. Afin d'en extraire automatiquement une liste de mots clés d'un texte, une série de traitements est exécutée, en partant de la segmentation, filtrage, normalisation, lemmatisation ensuite le calcul des fréquences et étiquetage grammatical et enfin la sélection des mots clés.

L'architecture globale de notre système est représentée par la figure suivante :

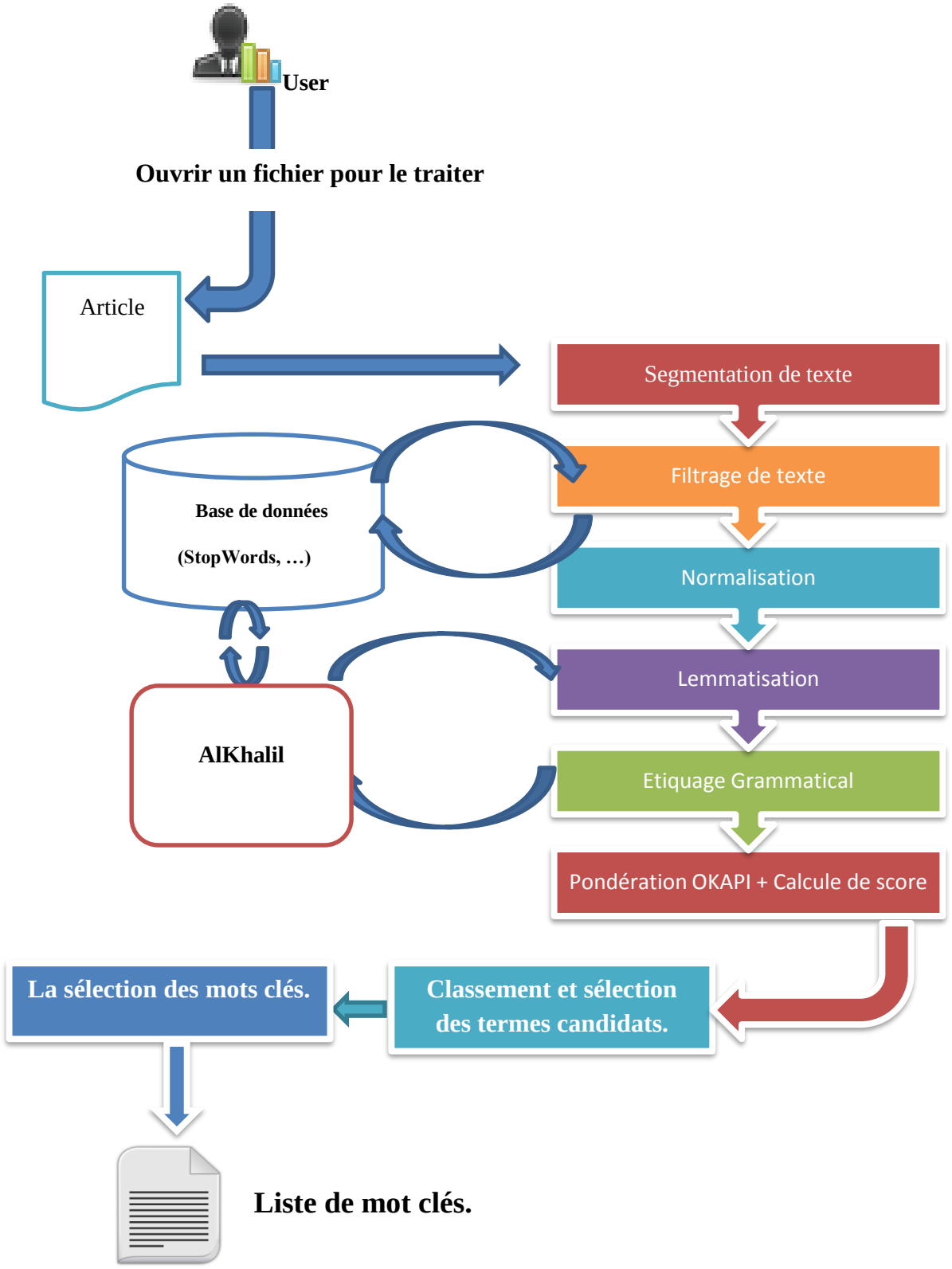


Figure 09 : architecture du Système (Key Words Arabic Identificator)

4.1 Pré-traitements

4.1.1 Filtrage manuel des articles (suppression des schémas, figures et tableaux)

Le filtrage manuel des articles commence par la suppression des figures et les schémas qui sont inutiles parce qu'on est en train de traiter des fichiers textuel (seulement le contenue textuelle qu'il est important), puis la conversion d'extension d'un article de type (.doc) fichier word a fichier texte (.txt).

4.1.2 Encodage uniques des textes

L'encodage unique des textes en format standard, permet de représenter les textes sans aucune déformation au niveau de caractère lors de lecture. Tous les textes de notre corpus sont représentés avec un encodage UTF-8 (chapitre02).

4.2 La segmentation

La segmentation est une étape nécessaire et signifiante dans tout traitement de la langue naturelle. La fonction d'un segmenteur est de couper un texte courant en segments, de sorte qu'ils puissent être introduits dans un capteur morphologique ou dans un étiqueteur de position pour un traitement ultérieur. Dans notre approche nous avons opté à une segmentation en paragraphe. Le paragraphe étant le segment qui garantit la couverture d'une unité thématique et un sens complet. Une frontière d'un paragraphe est facilement détectée par un point et/ou un saut de ligne.

4.3 Le filtrage

Le filtrage consiste à prendre chaque segment de l'étape précédente (des paragraphes) et éliminer tous les mots non significatifs. Pour chaque mot reconnu, on le compare avec un des éléments dans la base de données qui contient tous les mots non-significatifs. Si un mot en fait partie, il ne sera pas pris en considération pour le calcul de sa fréquence. La base de données regroupe tous les particules et/ou les mots vides (stopwords).

4.4 Analyse et traitement linguistique

4.4.1 La normalisation

Afin de manipuler les variations du texte qui peuvent être représentées en arabe, on applique plusieurs genres de normalisation sur le texte. Par exemple, dans l'arabe écrit, les voyelles sont souvent omises dans les textes, néanmoins, on peut parfois trouver quelques voyelles présentes avec les mots. Alors, l'élimination de ces voyelles est nécessaire pour fin de normalisation. Certaines lettres subissent une simple modification dans l'écriture qui n'influe pas considérablement sur le sens du mot. Mais l'encodage de ces lettres change d'un mot à un autre. Une autre raison pour ce prétraitement est que l'on a tendance fréquemment à mal écrire ces différentes formes de hamza. Ce genre d'erreurs est très répandu dans les textes arabes. Par exemple, le mot « أكل » est généralement écrit « اكل ». Aussi la lettre « ة » à la fin des mots qui peut être écrite de deux façons : « ة » ou « ه ». Les deux mots arabes « عادة » et « عاده » signifient le même mot (habitude) malgré que leur dernière lettre soit représentée différemment. La normalisation concerne les étapes suivantes :

- Enlever la ponctuation.
- Retirer les signes diacritiques (principalement voyelles faibles).
- Retirer les non-lettres arabes.
- Remplacer le إ ou le أ initial par l'alif nu ا.
- Remplacer le آ par le ا.
- Remplacer le ىء d'ordre par le ى
- Remplacer le ي final par le ى
- Remplacer le ة final par le ه

La liste des signes de ponctuation, des signes diacritiques, et des non-lettres est celle utilisée par l'Analyseur ALkhalil.

4.4.2 La lemmatisation (Stemming)

Un des traitements les plus importants pour la langue arabe, en vue de la recherche d'information et d'indexation est la lemmatisation des mots. L'objectif de la lemmatisation est de trouver la forme représentative d'index d'un mot à partir de sa forme représentée dans le document et réduire son espace caractéristique par l'application de la troncature des affixes.

Comme nous l'avons introduit précédemment, un mot arabe est formé

généralement par une séquence de {antéfixe, préfixe, noyau, suffixe, postfixe}. Ainsi un mot arabe peut avoir une forme plus compliquée si tous ces affixes sont attachés à sa forme standard. De plus la lemmatisation des mots nous donne une valeur exacte de la distribution des mêmes termes dans le texte.

Comme le vocabulaire dans les articles économiques est généralement proche, nous avons appliqué une lemmatisation profonde (calcul de racine) en utilisant l'analyseur morphosyntaxique AlKhalil.

Exemple :

« إن تطوير روح الإبداع المحاسبي في المؤسسات مرهون بمدى توفر مجمو عة من الشروط والعوامل لعل أهمها المناخ الملائم ومدى التحفيزات التي تقدمها المؤسسة لمبدعيها بالخصوص المحاسبين المبدعين. كما أن تسيير نشاطات الإبداع قد يختلف من مؤسسة لأخرى إلا أنه وماهو متفق عليه، فإن احترام العلاقات الإنسانية في تسيير الإبداعات المحاسبية وترقيتها.

كما تظهر أهمية الإبداع في كونه يعطي للمؤسسة القدرة التنافسية على مواجهة منافسيها وهذا على المستوى الجزائي بينما على المستوى الكلي يعتبر الإبداع أداة رئيسية لتحقيق التنمية الاقتصادية. » [21]

Comme on remarque dans le paragraphe le mot « *الإبداع*» il a plusieurs schèmes mais ils ont le même contexte dans ce cas il est mieux de prend la racine d'un mot qu'un le lemme.

4.4.2.1 Analyse morphosyntaxique par Alkhalil

AlKhalil Morpho Sys (برنامج الخليل الصرفي) est considéré comme un des meilleurs systèmes morphologiques arabes. En fait, Alkhalil a remporté la première position, parmi les 13 systèmes morphologiques arabes à travers le monde dans un concours organisé par la Ligue arabe pour l'éducation, la culture et la science (ALECSO,2010).
Ainsi pour analyser un terme (mot) Alkhalil procède comme suit :

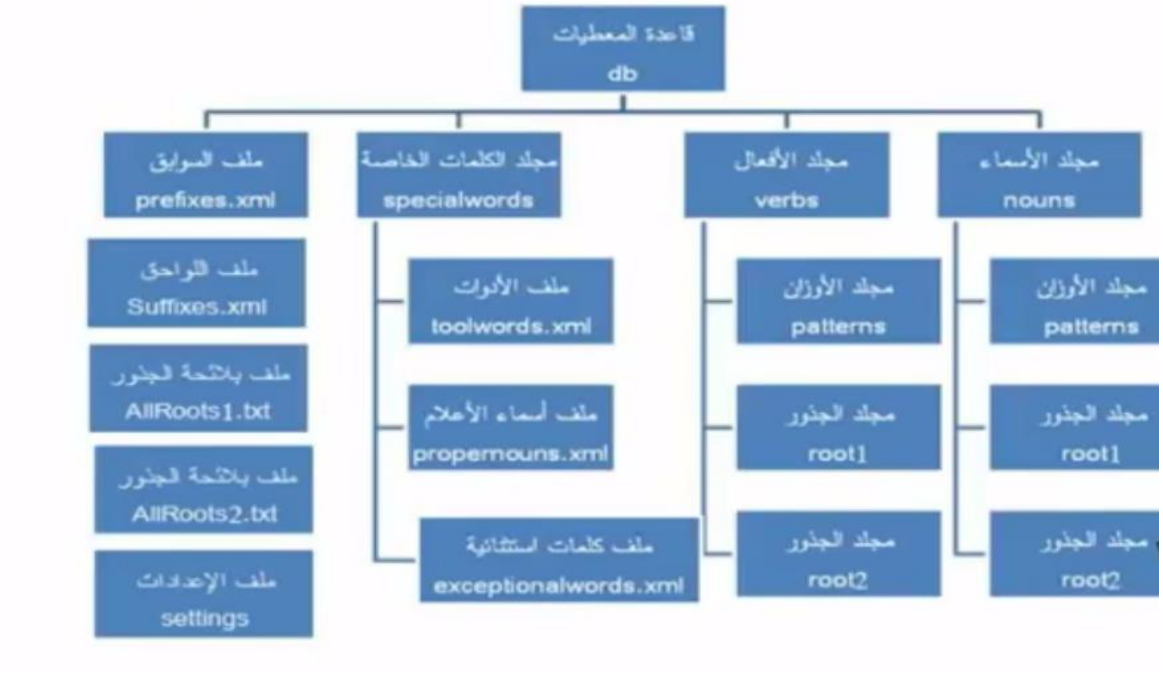
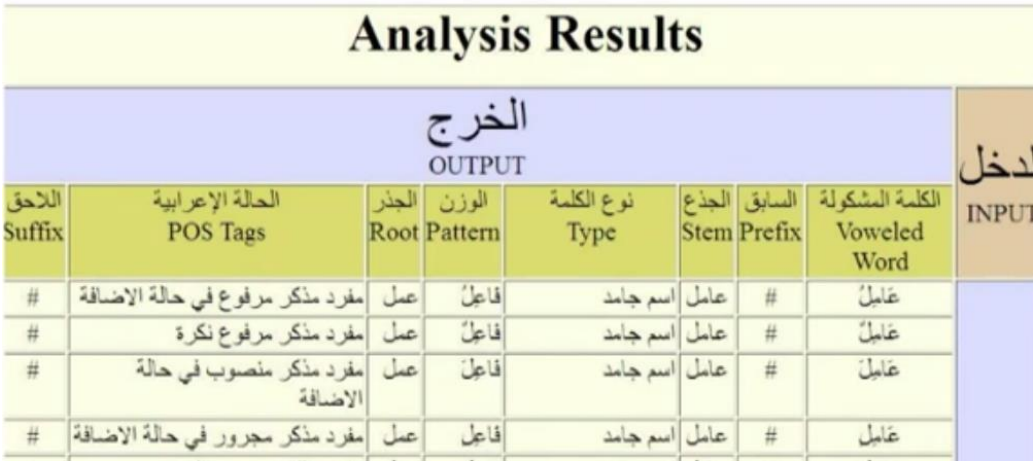


Figure 10 : Architecture d'analyseur morphologique Alkhalil

4.4.2.2 Etiquetage grammatical par Alkhalil

L'objectif de l'étiquetage grammatical est d'associer à chaque unité lexicale sa catégorie grammaticale (nom, verbe,..) à l'aide d'Alkhalil. Le principal intérêt de cet étiquetage est qu'il permet d'opérer un premier traitement de désambiguïsation des mots. L'étiqueteur AlKhalil peut ainsi associer plusieurs étiquettes (catégories) à chaque unité lexicale (surtout pour un mot ambigu). Dans notre projet, on se contente de considérer la première étiquette choisie par AlKhalil comme catégorie grammaticale de mot. En effet, l'information qui nous intéresse est la distinction entre terme nominal (nom, adverbe ou adjectif) et terme verbal (verbe). Ainsi l'étiquetage grammatical nous permet à la fin du processus d'identification des mots clés de faire une sélection favorisant les syntagmes nominaux.



Analysis Results

الخرج OUTPUT								الدخل INPUT
اللاحق Suffix	الحالة الإعرابية POS Tags	الجذر Root	الوزن Pattern	نوع الكلمة Type	الجذع Stem	السابق Prefix	الكلمة المشكولة Voweled Word	
#	مفرد مذكر مرفوع في حالة الاضافة	عمل	فاعِلُ	اسم جامد	عامل	#	عَامِلُ	
#	مفرد مذكر مرفوع نكرة	عمل	فاعِلٌ	اسم جامد	عامل	#	عَامِلٌ	
#	مفرد مذكر منصوب في حالة الاضافة	عمل	فاعِلَ	اسم جامد	عامل	#	عَامِلَ	
#	مفرد مذكر مجرور في حالة الاضافة	عمل	فاعِل	اسم جامد	عامل	#	عَامِلِ	

Figure 11 : Exemple sur l'Etiquetage grammatical par Alkhalil.

4.5 Analyse statistique

4.5.1 Pondération des fréquences des termes

Le calcul de la représentativité d'un terme repose sur sa fréquence d'apparition dans le texte. Afin de mesurer l'importance d'un terme le texte, nous avons utilisé la métrique Okapi (chapitre03), une nouvelle mesure qui vient de remplacer la célèbre tf-idf.

Pour calculer l'importance d'un terme dans un document, Okapi considère un document par rapport une collection de document (comme dans l'équation 3. Chapitre 3). Afin de considérer un segment par rapport à un ensemble de segment dans un texte, Okapi peut être calculée ainsi :

$$\text{Okapi}(\text{terme}) = TF_{BM25}(\text{terme}) \times \log\left(\frac{N - DF(\text{terme}) + 0{,}5}{DF(\text{terme}) + 0{,}5}\right)$$

$$\text{TF}_{BM25} = \frac{\text{TF}(\text{terme}) \times (k_1 + 1)}{\text{TF}(\text{terme}) + k_1 \times \left(1 - \text{b} + \text{b} \times \frac{\text{DL}}{DL_{\text{moyenne}}}\right)}$$

Telle que :

- **N** : Nombre de segment de texte.
- **TF** : Le nombre d'occurrence d'un mot dans le segment analysé de texte.
- **DF** : Le nombre de segment dans le quelle il est présent (mot).
- **DL** : La langueur de segment analysé.
- **DL moyen** : La langueur moyenne des segments du texte.
- **K1 et b** : Constants « 2 », « 0,75 » respectivement.

4.5.2 Calcul des positions des termes

Un score de position est attribué aux termes situant dans les entêtes des segments, cette position et identifiée relativement avec la longueur du segment (nombre de termes). Nous définissons la longueur de l'entête à 10% de la longueur du segment. D'après la collection des articles de notre corpus, ce taux représente efficacement l'entête d'un segment. Notant que ce taux peut être ajusté ou personnalisé par l'utilisateur du système.
Le score d'un terme par rapport à sa position est calculé comme suit :

$$Score\ position(terme) = (\ j(terme) * 100\ )/PrcSeg$$

$$PrcSeg\ (segment) = (prc * langSeg)/100$$

Telle que :

- **LangSeg** : Langueur d'un segment.
- **Prc** : Pourcentage de la longueur de l'entête.
- **J** : position de terme.

4.5.3 Calcul du score final

Après avoir calculé le score d'un terme par rapport à son score Okapi et sa position, un score final est attribué à chaque terme en combinant les deux scores déjà calculés. L'équation du score final d'un terme est calculée comme suit :

$$\text{Score final (terme)} = Okapi(terme) + Score\ position\ (terme)$$

Après avoir prétraités le texte, on arrive à l'étape fondamentale de notre travail, c'est l'identification de mot clés en utilisant les règles mentionnées précédemment. Les mots extraites sont stockés dans une base de données, que on va les utiliser dans la phase de classification.

4.5.4 Classement et sélection des termes candidats

Les termes sont classés sur la base de leurs scores finaux, de plus pesant au moins pesant.Telle que les mots les plus pesants doivent être sélectionnée en fonction de leur catégorie grammaticale (type : nom ou verbe), les élus (candidats) pour être mots clés sont ceux étiquetés comme syntagmes nominaux et une étape de classement est effectuée pour ranger les termes nominaux selon leurs poids.

4.5.5 La sélection des mots clés

Le nombre de mots clés sélectionnés pour être présents dans la liste finale des mots clés est calculé selon un taux de compression (nombre de mots clés souhaité) peut être définit par l'utilisateur du système. Nous initialisons ce taux à 5 (5 mots clés), ce nombre est généralement représentatif dans la collection de textes de notre corpus.

Notant qu'un mot clé n'est pas toujours un seul terme, il peut être composé de deux ou trois termes. La solution que nous proposons pour former les expressions clés (mots clés) est de ranger les termes ayant les même scores et qui sont cités ensemble dans le texte, c.-à-d. ceux qui ont des positions proches (i, i+1, i+2).

5. Conclusion

Dans ce chapitre, on a décrit notre système KWAI, dont l'objectif est de concevoir un système capable d'identifier automatiquement des mots clés présents dans des textes arabes et plus précisément dans les revues économiques (articles). La conception de système est divisé sur 5 étapes applicable sur un corpus de teste. Cette conception sera mise en fonction dans le chapitre qui suit.

Chapitre 05

Implémentation et Test

CHAPITRE 5
Réalisation et implémentation

1. Introduction

On présente dans ce chapitre l'implémentation de notre système KWAI, on commence tout d'abord par la présentation de l'environnement de développement, en détaillant les différents outils utilisés, puis on explique le déroulement de l'application, et enfin on interprète et on commente les résultats obtenus.

2. Environnement de développement

On présente dans cette section, le langage de programmation Java utilisé, et l'environnement de développement NetBeans.

2.1 Java

Java est un langage de programmation récent (les premières versions datent le 1995) développé par Sun MicroSystems. Il est fortement inspiré des langages C et C++.
Comme C++, Java fait partie de la grande famille des langages orientés objets. Il répond donc aux trois principes fondamentaux de l'approche orienté objet (POO) : l'encapsulation, le polymorphisme et l'héritage.

Java a rapidement intéressé les développeurs pour les raisons principales :

- C'est un langage orienté objet dérivé du C, mais plus simple à utiliser et plus « pur » que le C++. On entend par le « pur » le fait qu'en Java, on ne peut faire que la programmation orienté objet contrairement au C++ qui reste un langage hybride, c'est-à-dire autorisent plusieurs styles de programmation. C++ est hybride pour assurer une compatibilité avec le C ;

- Il est doté, en standard, de bibliothèques de classe très riches comprenant la gestion des interfaces graphiques (fenêtres, boites de dialogue, contrôles, menus, graphisme), la programmation multi-threads (multitâches), la gestion des exceptions, les accès aux fichiers et au réseau…L'utilisation de ces bibliothèques facilitent grandement la tâche du programmeur lors de la construction d'applications complexes ;
- Il est doté, en standard, d'un mécanisme de gestions des erreurs (les exceptions) très utile et très performant.
- Il est multi plates-formes : les programmes tournent sans modification sur tous les environnements où Java existe (Windows, Unix et Mac).

2.2 NetBeans

NetBeans est un environnement de développement intégré (EDI), placé en *open source* par Sun en juin 2000 sous licence CDDL (Common Développement and Distribution License) et GPLv2. En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C++, JavaScript, XML, Groovy, PHP et HTML de façon native ainsi que bien d'autres (comme Python ou Ruby) par l'ajout de greffons. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

L'environnement de base comprend les fonctions générales suivantes :

- configuration et gestion de l'interface graphique des utilisateurs,
- support de différents langages de programmation,
- traitement du code source (édition, navigation, formatage, inspection ..),
- fonctions d'import/export depuis et vers d'autres IDE, tels qu'Eclipse ou JBuilder,
- accès et gestion de bases de données, serveurs Web, ressources partagées,
- gestion de tâches (à faire, suivi …),
- documentation intégrée.

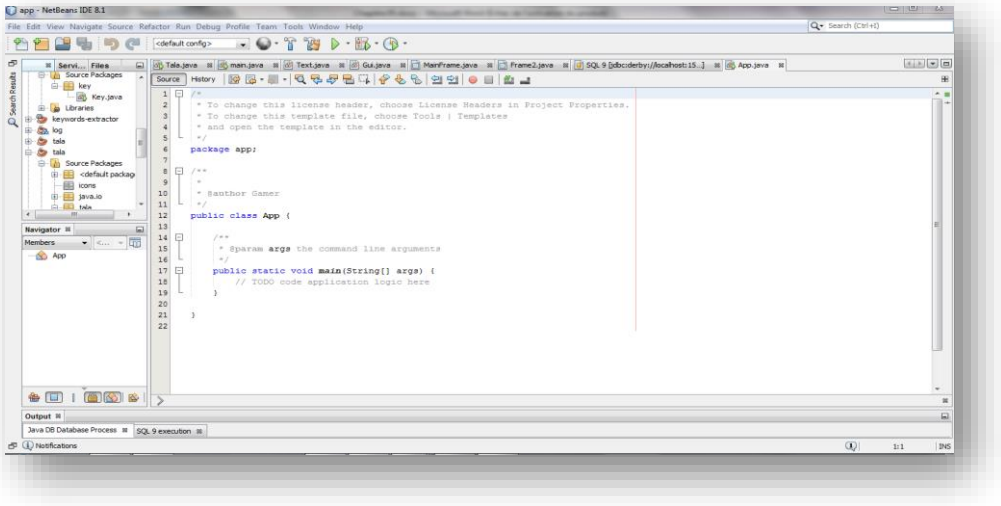


Figure 12 : Environnement NetBeans.

3. Description de KWAI

Notre système KWAI développé en Java à l'aide de l'environnement NetBeans, il est muni d'une interface graphique à travers laquelle, l'utilisateur peut entré des textes pour traitement et personnalisation des paramètres d'exécution. Nous avons utilisé plusieurs classes organisées sous forme de packages Java :

- Un package pour le prétraitement des textes ;
- Un package pour les différents traitements de texte ;
- Un package pour l'extraction des expressions subjectives ;
- Un package pour l'identification de mot clés ;
- Un package pour l'interface.

Nous avons ainsi utilisé les bibliothèques externes suivantes :

- Traitement des chaines et des flux de données :

  - Commons-io-1.3.2.jar
  - Commons-lang-2.4.jar

- Création et gestion des bases de données Apache DERBY:

  - Java DB Driver-derby.jar

- Java DB Driver-derbyclient.jar
- Java DB Driver-derbynet.jar

- Part Of Speech Tagging : AlKhalil

  - AlKhalil.jar

4. Déroulement

On présente dans cette section les différentes étapes de déroulement du processus d'identification de mots clés par notre système, dès la sélection de textes jusqu'à la classification des mots clés, en passant bien sûre par les étapes intermédiaires : prétraitement, traitement syntaxique, linguistique, et identifications des mots clés.

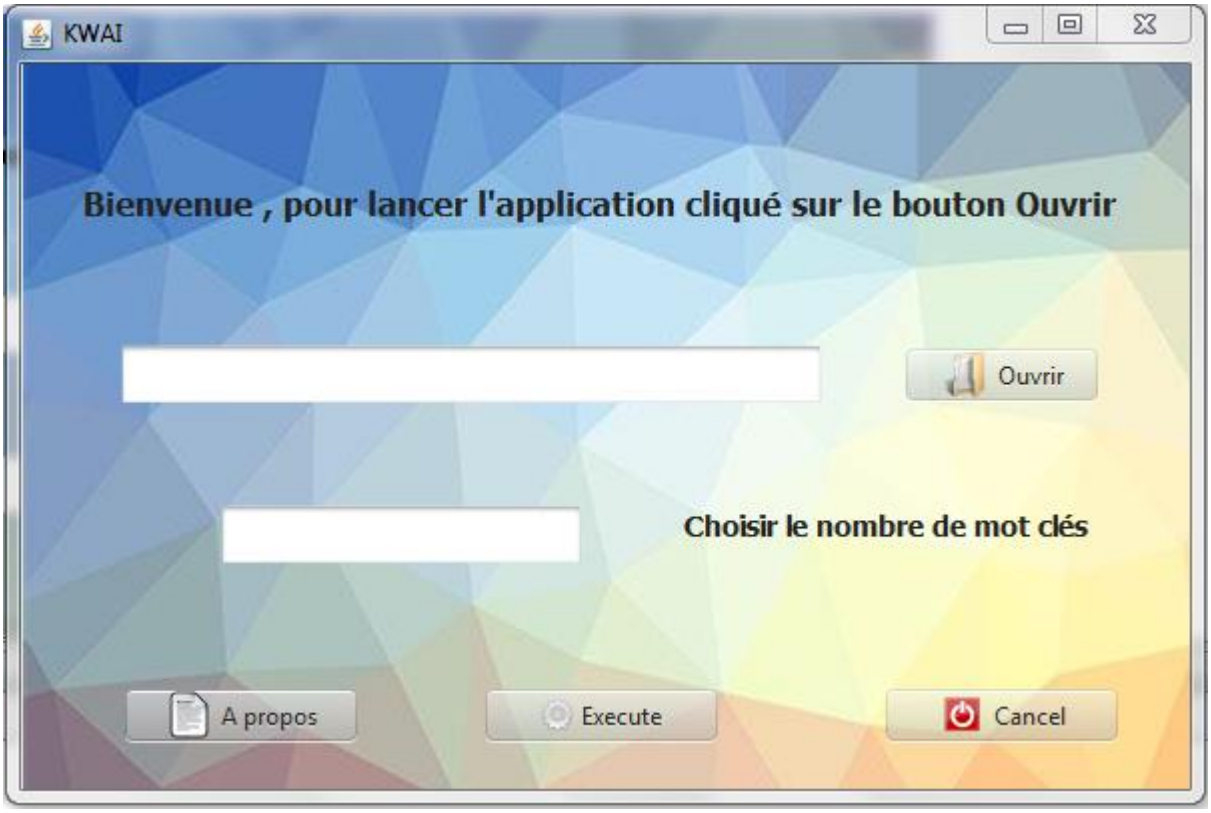
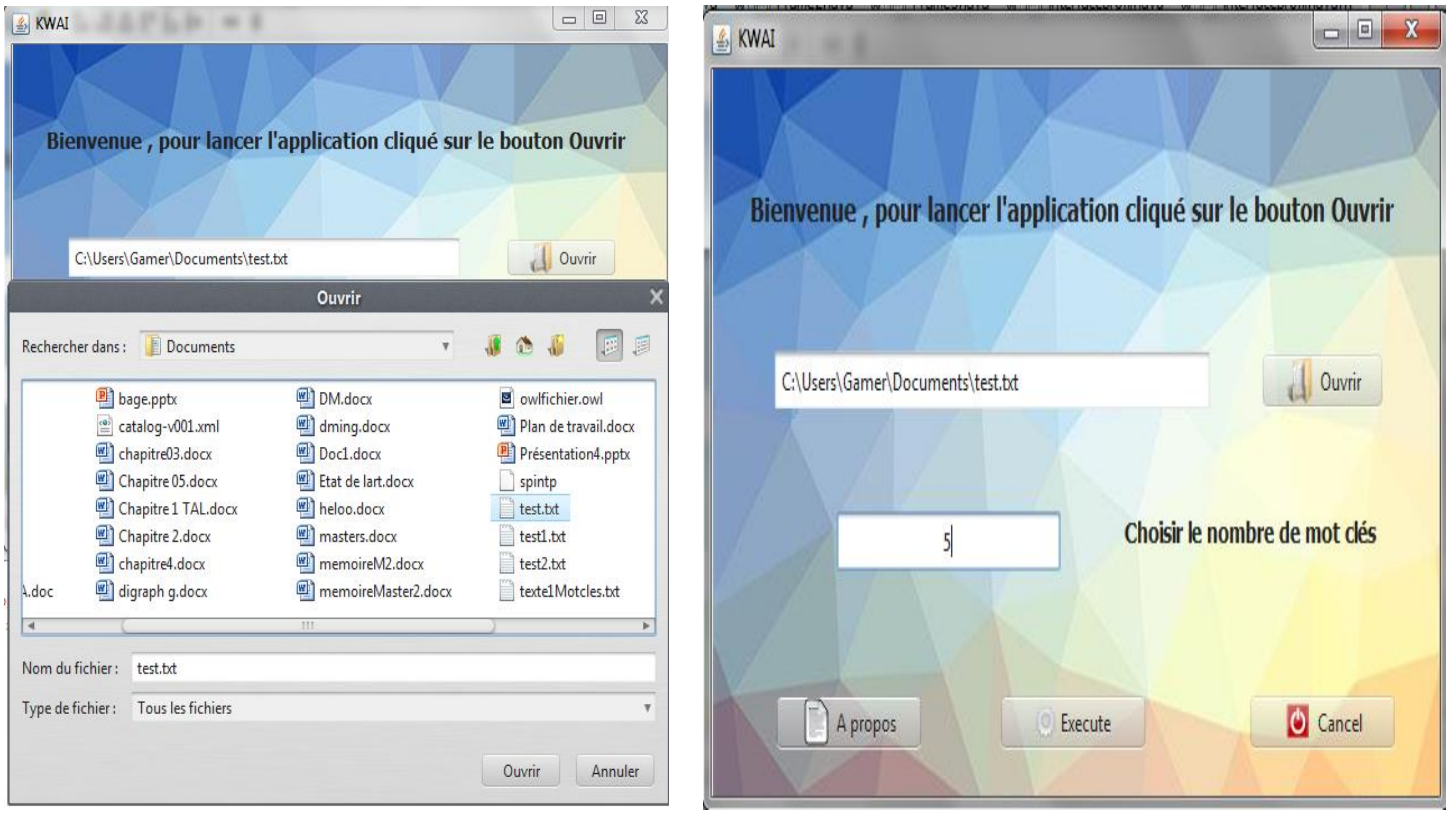


Figure 13 : Fenêtre principale de l'application.

4.1 Sélection des textes

Les textes que nus avons utilisés sont prétraités à l'avance et ils sont enregistrés en format «txt». Le traitement automatique commence par la sélection d'un fichier à traiter afin d'en extraire les mots clés, en activant le bouton *ouvrir* , et en choisissant le nombre de mot clés souhaités, puis le processus d'identification commence en activant le bouton *exécuter* .



(1) **(2)**

Figure 14 : (1),(2) La sélection de texte .

4.2 Traitement de texte

Notre système offre la possibilité d'effectuer des traitements sur le texte comme la segmentation et la lemmatisation, et l'affichage des textes prétraités. La figure suivante, indique cette possibilité :

- Le premier traitement du texte « La segmentation »

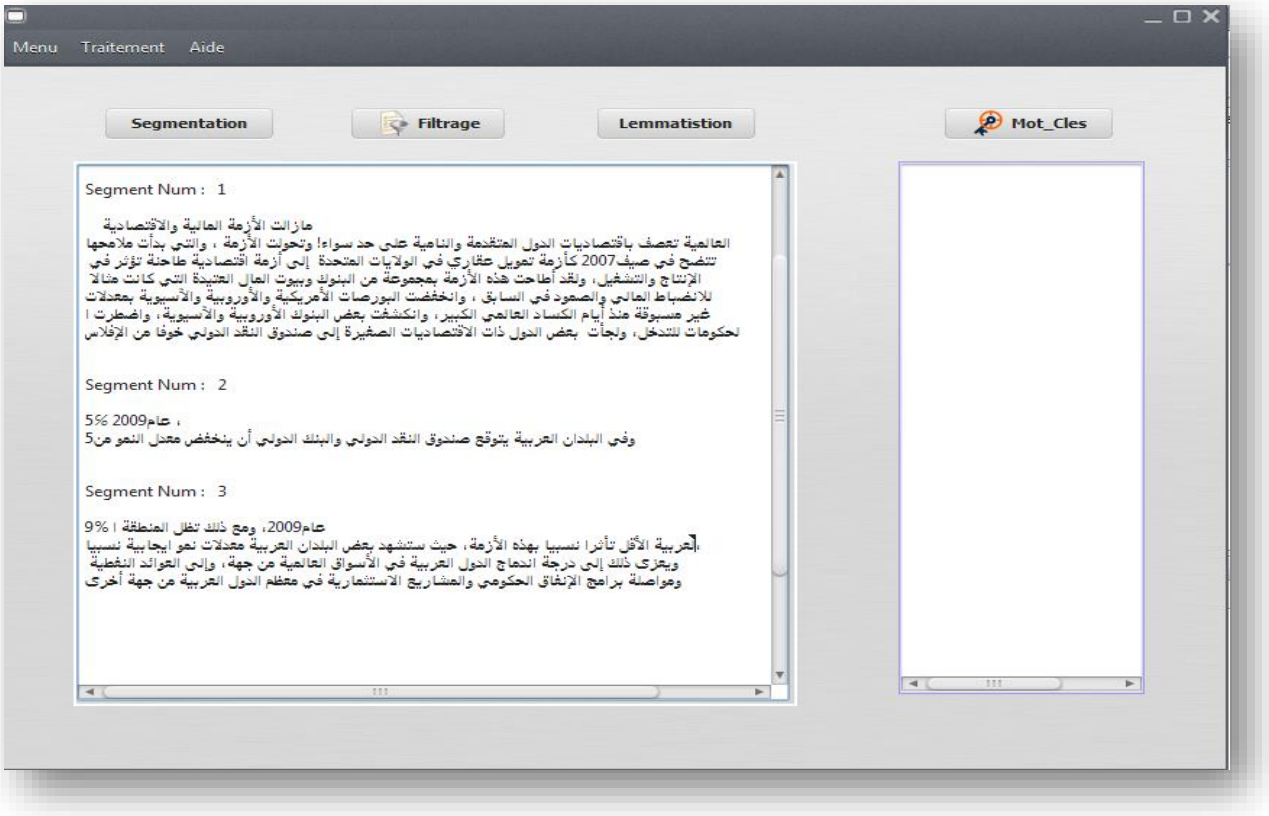


Figure 15 : Traitement de texte Segmentation.

- Le deuxième traitement du texte « Le Filtrage

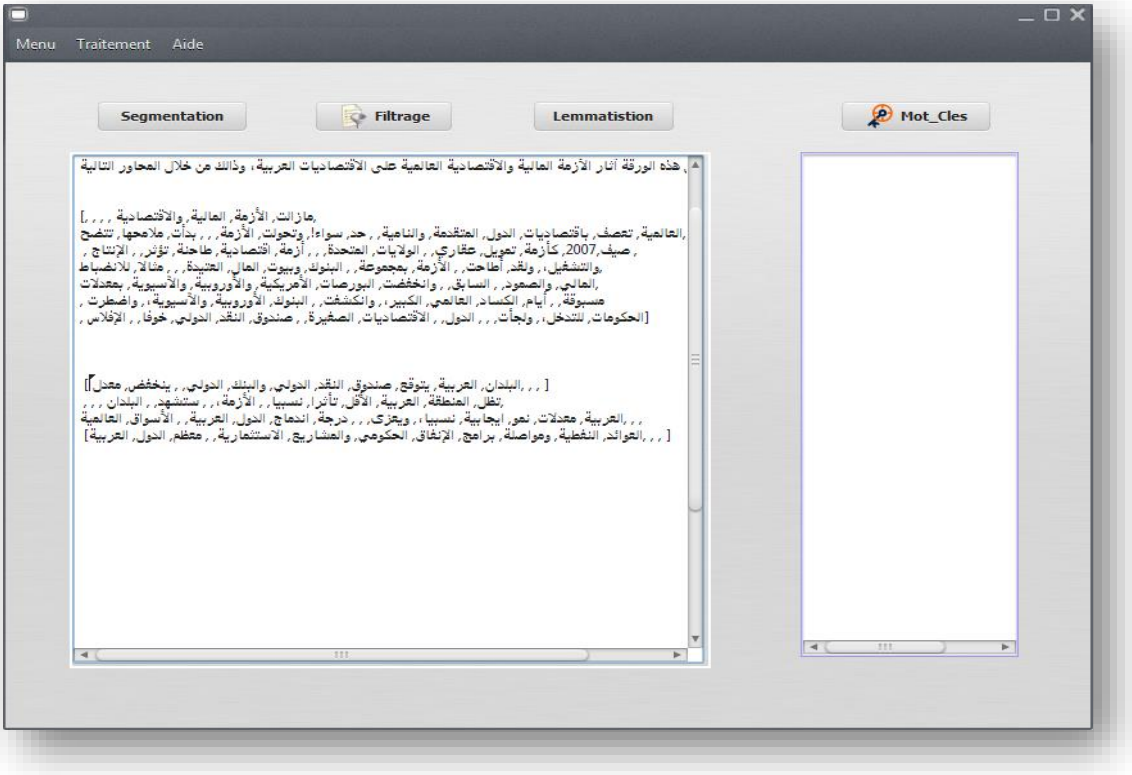


Figure 16 : Traitement de texte Filtrage.

- Le Teroisieme traitement « Lemmatisation »

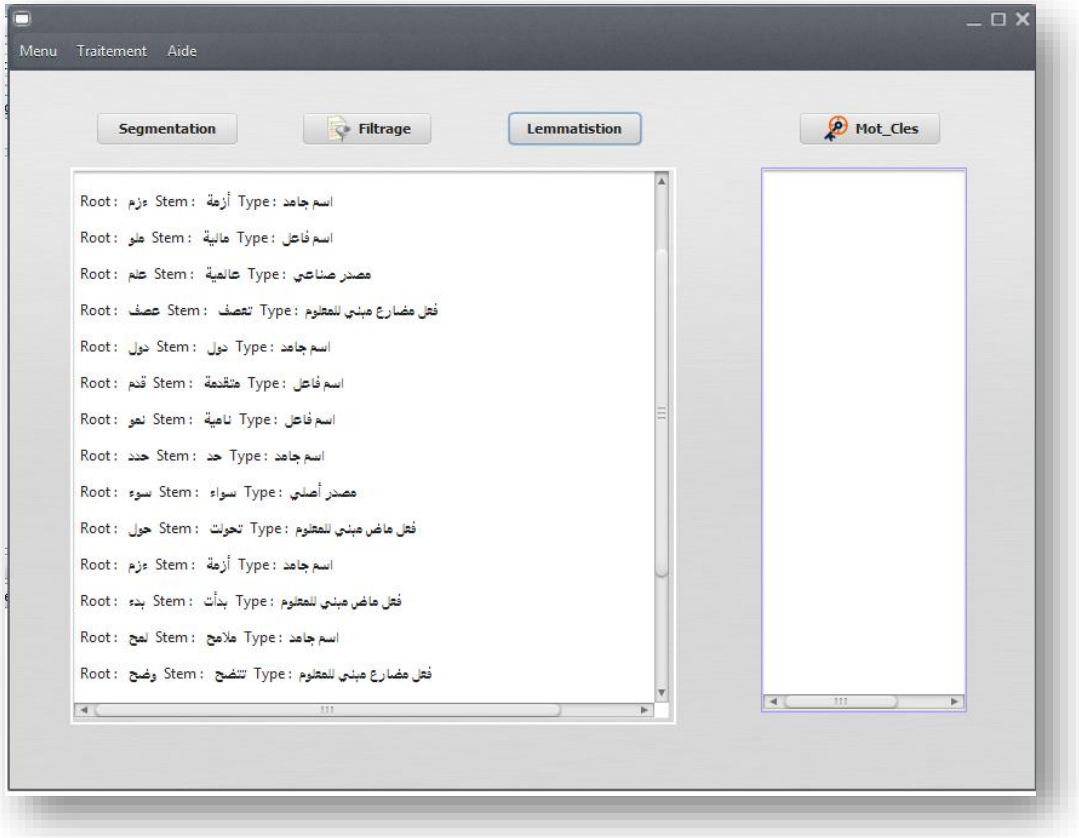


Figure 17 : Traitement de texte Lemmatisation.

4.3 Personnaliser les paramètres

A travers cet écran, un utilisateur avancé (linguiste) pourra personnaliser la longueur (en pourcentage) de l'entête d'un segment, cette valeur est initialisée à 10% par défaut, elle peut être ajustée selon la catégorie et la longueur moyenne des textes traités.

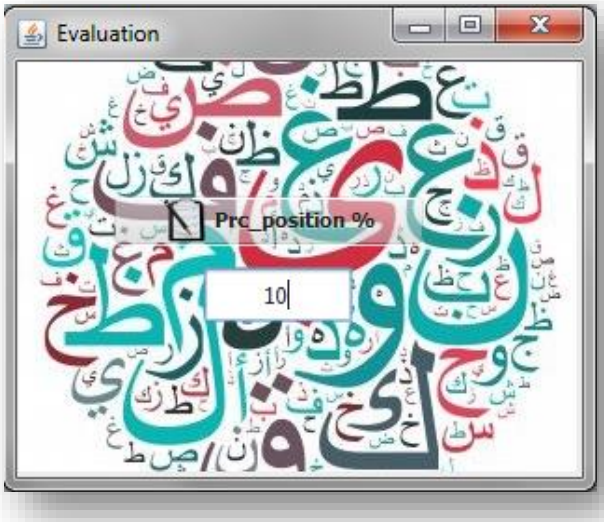


Figure 18 : Personnaliser les paramètres**.**

4.4 Identification automatique de mot clés

Dans cette étape, on arrive à l'étape d'extraction des éléments et des termes clés (des expressions, mots) présentes dans les textes. Après avoir entré un article, un simple clique sur le bouton *Mot Clés* affichera une liste de mots clés. Le nombre de mots clés sélectionnés est initialisé à 5, une valeur qui peut être aussi personnalisable.

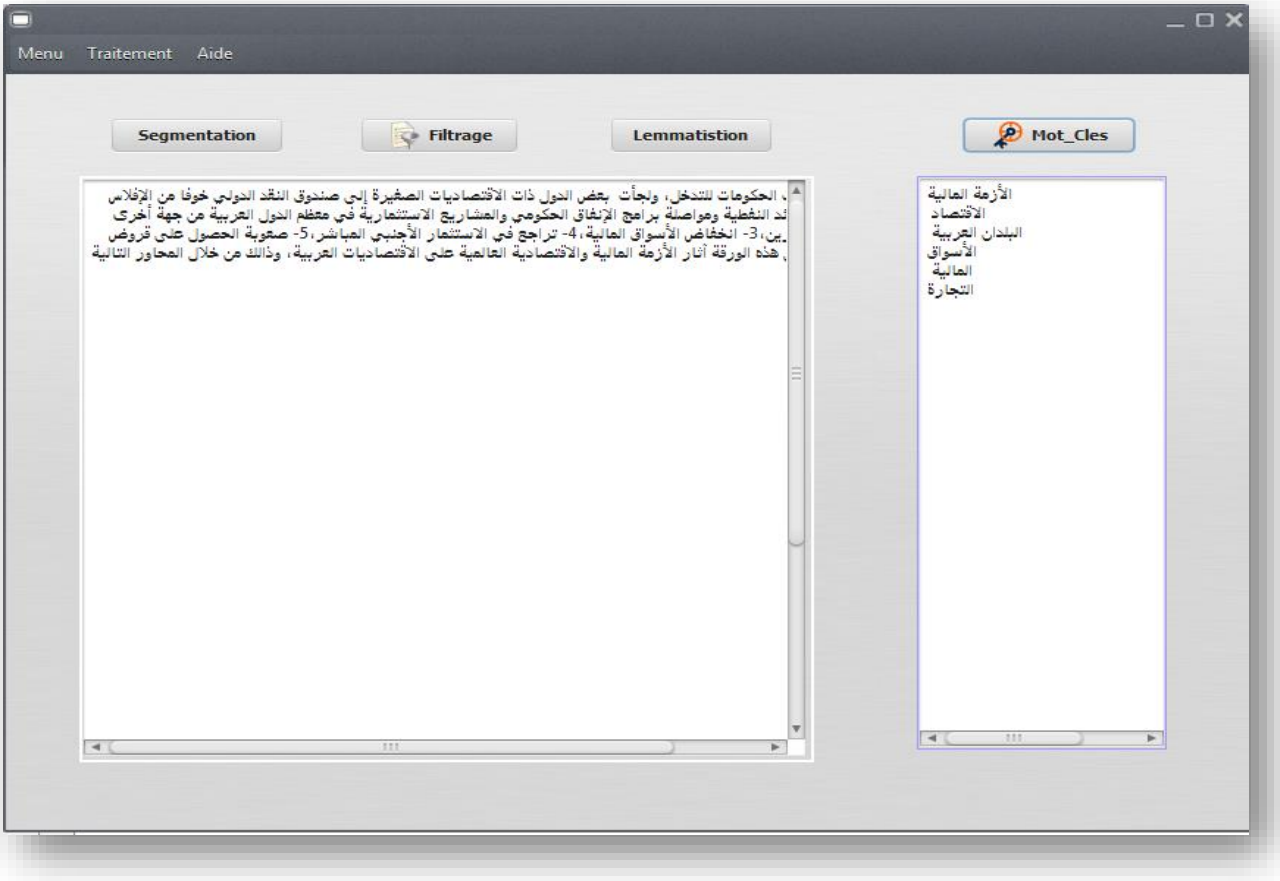


Figure 19 : Identification automatique de mot clés.

5. Evaluation du système

Afin d'évaluer la performance de notre système, nous avons fait une évaluation semi-automatique de notre système sur un ensemble de 10 articles de longueur moyenne.

L'évaluation sert à comparer la liste des mots sélectionnés par le système et ceux définit préalablement par l'auteur. Nous avons fait une comparaison par mot en utilisant les métriques Rappel et Précision afin de calculer F-mesure.

Cette technique est originalement utilisée pour évaluer des résumés automatiques (Moreno, 2011) mais elle peut être facilement adaptée à diverses applications.

Les mesures Rappel, Précision et F-mesure sont calculées par les équations suivantes :

- $Rappel = \frac{Corrects}{Corrects+Oubliés}$
- $Précision = \frac{Corrects}{Corrects+Incorrects}$
- $F-mesure = \frac{2.(Rappel.Précision)}{Rappel+Précision}$

Tel que :

*Correct*s : Nombre de mots sélectionnés par le système et par l'auteur.

Incorrects : Nombre de mots sélectionnés par le système et non pas par l'auteur.

Oubliés : Nombre de mots sélectionnés par l'auteur et non pas par le système

Ici Tableau

Cette petite comparaison montre que notre système d'identification produit des listes de mots clés acceptables. Par ailleurs, l'évaluation de tel système est généralement subjective car le style d'organisation des textes diffère d'un auteur à un autre et la liste des mots clés sélectionnés par l'auteur n'est pas toujours liée à l'importance des termes.

Document (Longueur)	Longueur du mot clé	Rappel	Précision
Article 1 **4 Segment** **192 mots**	7 mots	0.57	0.5
Article 2 **14 Segment** **806 mots**	12 mots	0.62	0.71
Article 3 **10 Segment** **559 mots**	12 mots	0.5	0.6

Article 4 **5 Segment** **300 mots**	10 mots	0.4	0.4
Article 5 **14 Segment** **700 mots**	7 mots	0.6	0.62
Total		0.53	0.56
F-mesure : **0.55**			

Tableau 10 : Evaluation du système.

6. Conclusion

A travers ce chapitre, nous avons présenté l'environnement de développement de notre système ainsi que les différentes interfaces graphiques qui à travers lesquelles nous pouvons superviser les différents traitements du système.

Notre système a pour rôle d'extraire les mots clés à partir des articles économiques écrit en langue arabe. Les résultats de notre système sont liés étroitement au choix de la technique hybride utilisée et à la catégorie d'articles traitée.

L'utilisation des catégories grammaticales détectées par le POS AlKhalil dans la sélection des mots clés a prouvé son intérêt dans le type de textes que nous avons étudié.

Conclusion et perspectives

L'identification automatique de mot clés dans les textes arabe a pour but de détecter et d'extraire les unités les plus saillantes d'un texte. Ces unités ou termes sont généralement cachés derrière des mots, des phrases et des paragraphes. Une expression de mot clés est l'unité la plus petite à partir de laquelle les termes sont identifiés. Les mots exprimant des idées générales sur le texte (comme un résumé), la démarche d'identification est basée, en premier lieu, sur l'extraction des segments (paragraphes), et ensuite, l'identification des termes candidats.

A travers notre projet, nous avons pu réaliser un système (KWAI) capable d'identifier automatiquement les mots clés à partir des textes écrit en langue arabe. Nous avons étudié cette langue d'un point de vue informatique en faisant ressortir les traits linguistiques et statistiques à partir du texte.

Nous avons utilisé une méthode hybride semi supervisée basée sur la combinaison des critères : fréquences pondérées, positions et catégories grammaticales des termes.

La méthode que nous avons adoptée s'est avérée adaptable et nous avons pu faire nos expérimentations et évaluations sur un corpus qui regroupe un ensemble d'articles économiques écrit en arabe.

Comme perspectives, nous pensons que l'ajout d'autres traits linguistiques comme la synonymie ou l'hyperonymie pour la détection et la sélection des mots clés améliorera énormément les résultats. D'une autre part, la segmentation formelle en paragraphe semble insuffisante et provoque des défauts de sens, par ailleurs, si on prend en considération les structures syntaxiques du discours au plus des signes de ponctuation, nous obtiendrons surement de meilleurs résultats.

Références bibliographiques

[1] M. A. Chéragui, Y. Hoceini et M. Abbas, “A Morphological Analysis of Arabic Language based on Multicriteria Decision Making: TAGHIT System”, IEEE, International Conference On Machine and Web Intelligence, (2010).

[2] M. A. Chéragui, Y. Hoceini et M. Abbas, “Une Approche Multicritères pour lever l'ambiguïté Morphologique dans le Texte Arabe » COSI : Colloque d’optiomisation des Systèmes d’Informations, (2010).

[3] M. Aljlayl and O. Frieder, On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach, *In 11th International Conference on Information and Knowledge Management (CIKM), November 2002, Virginia (USA).*

[4] Larkey L. S., Ballesteros L. and Connell M., Improving Stemming for Arabic Information Retrieval: Light Stemming and Cooccurrence Analysis, *In Proceedings of the 25th Annual International Conference on Research and Development in Information Retrieval (SIGIR 2002), Tampere, Finland, August 2002.*

[5] J. Leclerc, L'aménagement linguistique dans le monde, *http://www.tlfq.ulaval.ca/axl/monde/famarabe.htm* .

[6] Exploration d'approches statistiques pour le résumé automatique de texte

[7] Baloul Sofiane Développement d’un système automatique de synthèse de la parole a partir du texte arabe standard voyellé, le 27 mai 2003.

[8] De l’´évaluation en Traitement Automatique des Langues mémoire d’Habilitation `a Diriger des Recherches École Doctorale d’Informatique Université Paris-Sud Patrick Paroubek Ingénieur de Recherche au LIMSI-CNRS (groupe ILES).

[9] (Aloulou et al. 02, Baccour et al. 03, Boualem 93, Debili 01, Dichy et al. 02, Gaubert 01, Jaccarini 97.)

[10] Hammo B., Abu-Salem H., Lytinen S., Evens M., QARAB: A Question Answering System to Support the Arabic Language, *Workshop on Computational Approaches to Semitic Languages. ACL 2002, July 2002, Philadelphia, PA.*

[11] A. Chalabi : Sakhr Web-based Arabic<>English MT engine, *ACL/EACL 2001 Workshop on Arabic Language Processing, Toulouse July2001(http://www.elsnet.org/arabic2001/chalabi.pdf)*

[12] A. Boudlal, A. Lakhouaja, A. Mazroui and A. Meziane, Alkhalil Morpho SYS1: A Morphosyntactic Analysis System for Arabic Texts, In Proceedings of the 2010 International Arab Conference on Information Technology (ACIT), Lybia, 2010.

[13] Shereen Khoja, Arabic stemmer.

[14] Une petite introduction au Traitement Automatique des Langues Naturelles François Yvon.

[15] Traitement automatique des langues pour l'accès au contenu des documents?Christian Jacquemin?, Pierre Zweigenbaumy? LIMSI-CNRS BP 133 91403 ORSAY Cedex FRANCE jacquemin@limsi.fr et http://www.limsi.fr/Individu/jacquemi/ y DIAM : Service d'informatique médicale, DSI/AP-HP et Département de Biomathématiques, Université Paris 6 91, bd de l'Hôpital, 75634 Paris Cedex 13 pz@biomath.jussieu.fr et http://www.biomath.jussieu.fr/~pz/

[16]Journal of Computational Information Systems4:3(2008) 1169-1180 Available at http://www.JofCI.org 1553-9105/ Copyright © 2008 Binary Information Press March, 2008 Automatic Keyword Extraction from Documents Using Conditional Random Fields.

[17] Mohamed Hedi Maaloul. Approche hybride pour le r_esum_e automatique de textes. Application _a la langue arabe.. Traitement du texte et du document. Universit_e de Provence - Aix-Marseille I, 2012. Fran_cais. <tel-00756111v1>.

[18] http://www.larousse.fr/dictionnaires/francais/mot-cl%C3%A9_mots-cl%C3%A9s/52770

[19] G. Salton, "Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computers", *Addison-Wesley, Reading*, Pennsylvania (1989)

[20] https://fr.wikipedia.org/wiki/Mot_clef

[21] المركز الجامعي خميس مليانة – معهد العلوم الاقتصادية وعلوم التسيير مجلة الاقتصاد المعاصر

عين الدفلى – الجزائر

[22] HAL Id: hal-00821671 https://hal.archives-ouvertes.fr/hal-00821671

[23] Improved Automatic Keyword Extraction Given More Linguistic Knowledge. *In Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*.

[24] NOBATA, C., COTTER, P., OKAZAKI, N., REA, B., SASAKI, Y., TSURUOKA, Y., TSUJII, J. et ANANIADOU, S. (2008). Kleio : a Knowledge-enriched Information Retrieval System for Biology. *In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*.

[25] JONES, K. (1972). A Statistical Interpretation of Term Specificity and its Application in Retrieval.

[26] PAUKKERI, M. et HONKELA, T. (2010). Likey : Unsupervised Language-Independent Keyphrase Extraction. *In Proceedings of the 5th International Workshop on Semantic Evaluation*.

[27] CLAVEAU, V. (2012). Vectorisation, Okapi et Calcul de Similarité pour le TAL : pour Oublier Enfin le TF-IDF. *In Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*.

[28] BARKER, K. et CORNACCHIA, N. (2000). Using Noun Phrase Heads to Extract Document Keyphrases. *In Proceedings of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence : Advances in Artificial Intelligence*.

[29] TOMOKIYO, T. et HURST, M. (2003). A Language Model Approach to Keyphrase Extraction. *In Proceedings of the ACL 2003 workshop on Multiword expressions : analysis, acquisition and treatment-Volume 18*.
[30] DING, Z., ZHANG, Q. et HUANG, X. (2011). Keyphrase Extraction from Online News Using Binary Integer Programming. *In Proceedings of 5th International Joint Conference on Natural Language Processing*.
[31] MATSUO, Y. et ISHIZUKA, M. (2004). Keyword Extraction from a Single Document Using Word Co-occurrence Statistical Information.

[32] LIU, Z., LI, P., ZHENG, Y. et SUN, M. (2009). Clustering to Find Exemplar Terms for Keyphrase Extraction. *In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 1*.
[33] MIHALCEA, R. et TARAU, P. (2004). Textrank : Bringing Order Into Texts. *In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.
[34] WAN, X. et XIAO, J. (2008a). Collabrank : Towards a Collaborative Approach to Single-Document Keyphrase Extraction. *In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*.
[35] LIU, Z., LI, P., ZHENG, Y. et SUN, M. (2009). Clustering to Find Exemplar Terms for Keyphrase Extraction. *In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 1*.
[36] LIANG, W., HUANG, C., LI, M. et LU, B. (2009). Extracting Keyphrases from Chinese News Articles Using Textrank and Query Log Knowledge. *In Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*.
[37] WITTEN, I., PAYNTER, G., FRANK, E., GUTWIN, C. et NEVILL-MANNING, C. (1999). KEA : Practical Automatic Keyphrase Extraction. *In Proceedings of the 4th ACM conference on Digital libraries*.
[38] FRANK, E., PAYNTER, G., WITTEN, I., GUTWIN, C. et NEVILL-MANNING, C. (1999). Domain-Specific Keyphrase Extraction.
[39] TURNEY, P. (1999). Learning Algorithms for Keyphrase Extraction.
[40] NGUYEN, T. et KAN, M. (2007). Keyphrase Extraction in Scientific Publications. *In Proceedings of the 10th international conference on Asian digital libraries : looking back 10 years and forging new frontiers*.
[41] ZHANG, K., XU, H., TANG, J. et LI, J. (2006). Keyword Extraction Using Support Vector Machine.

[42] HERBRICH, R., GRAEPEL, T. et OBERMAYER, K. (1999). Support Vector Learning for Ordinal Regression. *In Artificial Neural Networks, 1999*.
[43] JOACHIMS, T. (2006). Training Linear SVMs in Linear Time. *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*.
[44] JIANG, X., HU, Y. et LI, H. (2009). A Ranking Approach to Keyphrase Extraction. *In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*.
[45] EICHLER, K. et NEUMANN, G. (2010). DFKI KeyWE : Ranking Keyphrases Extracted from Scientific Articles. *In Proceedings of the 5th International Workshop on Semantic Evaluation*.
[46] SARKAR, K., NASIPURI, M. et GHOSE, S. (2010). A New Approach to Keyphrase Extraction Using Neural Networks.
[47] Contribution au resume automatique multi-documents Aurelien Bossard