République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université 8Mai 1945 – Guelma
Faculté des sciences et de la Technologie
Département d'Electronique et Télécommunications

Mémoire de fin d'étude
pour l'obtention du diplôme de Master Académique

Domaine : **Sciences et Technologie**
Filière : **Télécommunications**
Spécialité : **Réseaux et Télécommunications**

Développement d'un outil facilitant la recherche d'information: Amélioration d'un lemmatiseur arabe

---

Présenté par :

Rebbani Hamza et Djaballah Lyes

Sous la direction de :

Dr. Abainia Kheireddine

Juillet 2019

Remerciements

Nous tenons à exprimer nos remerciements et notre profonde gratitude avant tout au bon DIEU qui nous a donné le courage et la force pour mener à bien ce modeste travail.

Nous ne pouvons qu'être infiniment reconnaissants envers nos parents pour leur soutien indescriptible, leur patience, leur confiance et leurs sacrifices. Nous leur dédions avec plaisir ce travail ainsi qu'à toute notre grande famille.

Nous tenons à exprimer nos remerciements les plus vifs à notre encadreur Dr. Abainia Kheireddine, qui a su nous guider et nous aider dans ce travail avec beaucoup de tact et de gentillesse et qui nous a permis de découvrir un domaine très intéressant celui des lemmatiseurs. Qu'il trouve ici notre estime et notre profond respect.

Nos remerciements vont à tous nos autres amis, à notre entourage et à toutes les personnes qui nous ont fait confiance, nous ont soutenu, nous ont encouragé et nous ont aidé de près ou de loin.

Nos remerciements vont également à tous ceux qui ont accepté avec bienveillance de participer au jury de ce mémoire.

Résumé

Dans la résolution des problèmes de correspondance dans la recherche scientifique, l'arabe est considéré comme l'une des langues les plus difficiles à cause de sa morphologie spécifique. La technique la plus populaire et la plus utilisée pour résoudre ce genre de problème est la lemmatisation. Cette technique est une étape de prétraitement dans la plupart des systèmes de récupération d'informations. La lemmatisation a pour but la réduction des différentes formes grammaticales d'un mot.

Dans la littérature, il existe plusieurs algorithmes traitant la lemmatisation des mots arabes. Toutefois, la plupart d'entre eux sont malheureusement réservés à un nombre limité de mots, et présentent certaines confusions entre les lettres originales et les affixes. En outre, ils utilisent généralement un dictionnaire de mots ou de modèles.

Le but de notre travail est l'amélioration d'un algorithme de lemmatisation assoupli (léger) déjà conçu, i.e. ARLStem. Cette amélioration est fondée sur l'introduction de nouvelles règles permettant de supprimer les préfixes, suffixes et infixes de manière intelligente. Par ailleurs, une comparaison de l'efficacité de notre algorithme avec d'autres lemmatiseurs existants a été réalisée en utilisant les paramètres de Paice, à savoir l'indice de sous-lemmatisation (UI) et l'indice de sur-lemmatisation (OI). La comparaison a été faite sur le corpus ARASTEM, où les résultats obtenus ont montré que notre lemmatiseur possède des performances élevées et plus efficace que les lemmatiseurs auxquels il a été comparé.

الملخص

اللغة العربية تعتبر واحدة من أهم و أصعب اللغات في مجال التعرف على الأشكال و معالجة النصوص، وهذا لميزتها بقواعد النحو خاصة و صعبة. حيث نجد أن أكثر التطبيقات المستعملة هي استخراج الجذور تلقائيا، و التي تعتبر من أهم المراحل قبل معالجة النصوص و البحث عن المعلومات، حيث ترتكز على تحويل الكلمات إلى جذورها و بالتالي تجميع الكلمات التي تشترك في نفس الجذر.

يوجد الكثير من الأبحاث المقترحة و التي تعالج هذا الموضوع، لكن أغلب الخوارزميات المتواجدة تعالجن فئة معينة من الكلمات. علاوة على ذلك، هي تمتاز بخلط الحروف الأصلية للكلمات مع الحروف الزائدة، أو تستعمل قاموس من الكلمات أو الأشكال.

في هذا البحث نتطرق إلى تحسين خوارزمية لاستخراج الجذور، ألا وهي ARLSTEM. حيث أن هذا التحسين يتمثل في اقتراح قواعد جديدة لحذف الحروف الزائدة في أول الكلمات، منتصفها و آخرها. الخوارزمية المقترحة قورنت مع العديد من الأخريات باستعمال مؤشرات Paice و قاعدة البيانات ARASTEM، حيث أن النتائج المتحصل عليها أثبتت أن مقترحنا أحسن من باقي الخوارزميات و يمكن الاعتماد عليه في البحث عن المعلومات باللغة العربية.

Abstract

Arabic is considered as one of the most difficult languages because of its specific morphology, as well as it is considered as a challenging research area in computational linguistics (CL) and pattern recognition. The most popular technique and the most used one to solve the majority of CL applications is the stemming (lemmatization), which is a preprocessing step of several applications. The purpose of the stemming is to reduce different grammatical forms of a word into a single form (stem or root).

In the literature, there are several algorithms dealing with the Arabic stemming of Arabic words. However, most of them are unfortunately reserved for a limited number of words, have a lot of confusions between original letters and affixes, and generally use a dictionary of words or models.

The goal of our work is to improve an existing stemming algorithm (ARLStem). Our improvement is based on introducing some new rules to remove affixes (prefixes, suffixes, and infixes) in a smart way. A comparison in terms of Paice's parameters (UI and OI) has been conducted on ARASTEM corpus between our improvement and other stemmers, where the results showed that the proposed algorithm is suitable and outperformed the other ones.

Sommaire

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ABREVIATIONS

RI: Recherche d'information

SRI: Un système de recherche d'information

CT: catégorisation textuelle

TREC: Text REtrieval Conference

HTML: HyperText Markup Language

URL: Uniform Resource Locator

TALN: Traitement du langage naturel

UTF-8: Universal Character Set Transformation Format - 8 bits

Unicode: Universal Character Encoding

UI: under stemming Index

OI: Over stemming Index

SW: Stemming Weight

DMT: Desired Merge Total

DNT: Desired Non-merge Total

GDMT: Global Desired Merge Total

GDNT: Global Desired Non-merge Total

UMT: Unachieved Merge Total

GUMT: Global Unachieved Merge Total

WMT: Wrongly Merged Total

GWMT: Global Wrongly Merged Total

DT : Decision Trees

WCC: Words per Conflation Class

ARLSTem: Arabic Light Stemmer

ARASTEM: ARAbic STEMming of noisy texts

ISRI: The Information Science Research Institute

Introduction générale

La recherche d'informations (RI) vise à retrouver des documents dont le contenu peut être du texte, des images ou tout autre produit multimédia traitant un ou plusieurs sujets d'information. L'objectif de répondre au besoin d'un utilisateur en information, nécessite d'une part, une meilleure compréhension de la demande, exprimée le plus souvent par une requête libre. D'autre part, une organisation adéquate du fond documentaire, concrétisée par une construction de l'index. La difficulté à laquelle est confronté le champ de récupération d'informations est le problème de correspondance. Il existe de nombreux cas où deux mots ne sont pas tout à fait identiques, mais auxquels on souhaiterait avoir une correspondance. En fonction de la nature du langage, plusieurs techniques ont été développées pour traiter ce genre de problème.

Les langues naturelles sont fondées sur des règles grammaticales, syntaxiques et morphologiques dont le niveau de difficulté et de complexité dépend de la langue elle même. La langue arabe est considérée comme une des langues les plus difficiles à traiter, car elle est hautement flexionnelle et a donc un besoin particulier pour une lemmatisation efficace.

La lemmatisation joue un rôle très important dans les systèmes d'indexation et de recherche actuels. Le fondement d'une lemmatisation est basé sur la réduction des mots en leur racine, et ce généralement par suppression des suffixes et des préfixes (affixes) qui leur sont attachés.

En ce qui concerne la langue arabe, les lemmatiseurs existants ne sont pas très fiables et peuvent induire certaines erreurs dans la pratique, ce qui en diminue les performances (Baeza-Yates, 1992). C'est pourquoi on s'est intéressé dans notre travail à l'amélioration d'un lemmatiseur déjà existant appelé ARLStem par l'introduction de nouvelles règles susceptibles d'augmenter ses performances. Pour pouvoir estimer l'efficacité du nouveau lemmatiseur, une étude comparative basée sur la méthode de Paice, qui consiste à calculer les erreurs de sous-lemmatisation (UI) et les erreurs de sur-lemmatisation (OI), sera menée. Cette étude comparative entre notre lemmatiseur et d'autres lemmatiseurs assouplis (légers ou *light-stemmers* en anglais) existants sera réalisée sur le corpus ARASTEM.

Ce mémoire comporte trois chapitres. Le premier chapitre sera dédié à la présentation de la technique de recherche d'information. Nous y définirons la technique et donnerons les concepts de base de la recherche d'information avec cette technique. Nous y définirons aussi la catégorisation textuelle accompagnée d'un bref historique, et on présentera les

différents domaines de cette technique. Ce chapitre sera clôturé par l'exposition des difficultés de la catégorisation des documents textuels et ses applications.

Dans le second chapitre, nous définirons la morphologie arabe et nous y discuterons les différents types. Afin de citer tous les éléments de base la langue arabe, ses propriétés morphologiques, ainsi qu'une description schématique sur sa morphologie seront présentées. Ce chapitre sera clos par la présentation de la technique de lemmatisation et la méthode utilisé pour son évaluation.

Le dernier chapitre de notre mémoire renfermera les différentes étapes utilisées par notre lemmatiseur, ainsi que le corpus utilisé par ce dernier suivi par une description sur les analyseurs morphologiques arabes connus. Enfin, nous terminerons ce chapitre par l'évaluation de son efficacité en comparant ses performances par rapport à d'autres lemmatiseurs.

Chapitre 1 : Généralité sur la catégorisation des textes

1.1. Introduction

Historiquement, la Recherche d'Information (RI) est un domaine lié aux sciences de l'information et à la bibliothéconomie, d'où elle a pour but de développer des systèmes capables de retrouver les documents pertinents suite à une requête d'utilisateur, i.e. à partir d'une base de documents volumineuse et l'indexation de ceux-ci. L'informatique a permis le développement d'outils pour traiter l'information et établir la représentation des documents au moment de leur indexation, ainsi que pour rechercher l'information.

Dans ce chapitre, nous discutons quelques aspects de la recherche d'information, ainsi des définitions. Ensuite, nous discutons ses processus et de son utilisation sur le web. Dans la deuxième partie de ce chapitre, nous présentons d'abord un bref historique sur la catégorisation des documents textuels, ainsi que les domaines et les systèmes de cette dernière. Enfin, nous exposons les difficultés de la catégorisation des textes et ses applications.

1.2. Recherche d'information

1.2.1. Définitions

Plusieurs définitions de la recherche d'information ont vu le jour dans ces dernières années, or nous citons dans ce contexte les trois définitions suivantes:

Définition 1: La R.I. est un outil qui répond au besoin d'information contenu dans de grandes collections (généralement stockées dans des ordinateurs), pour trouver généralement des documents de nature structurée ou non (généralement des textes).

Définition 2: La R.I. est l'ensemble des techniques de stockage, de récupération et souvent de diffusion des données enregistrées, notamment à travers l'utilisation d'un système informatisé.

Définition 3: La R.I. est une discipline de recherche qui intègre des modèles et des techniques dont le but est de faciliter l'accès à l'information pertinente pour un utilisateur ayant un besoin en information [Abbasi and Meftah, 2013].

1.2.2. Concepts de base de la recherche d'information

Un système de recherche d'information (SRI).est un ensemble de techniques qui assurent les fonctions nécessaires pour la R.I. Il a pour rôle de sélectionner les documents qui peuvent répondre au besoin en information de l'utilisateur formulé par une requête de recherche d'information. Cette dernière est considérée comme toutes les techniques permettant de trouver à partir d'une collection de documents, ceux qui sont susceptibles de répondre aux besoins de l'utilisateur [Abbasi and Meftah, 2013]. De cette définition, on distingue trois notions clés telles que la collection des documents, document, besoin d'information.

1.2.2.1. Document et collection de documents

On appelle document toute unité qui peut constituer une réponse à une requête d'utilisateur. Dans son acceptation courante, l'une des définitions possibles du terme document est de le considérer comme un support physique de l'information [Hammache, 2013]. Ce dernier peut être un texte, une page web, une image, une séquence vidéo, etc. L'ensemble des documents manipulés par un SRI se nomme collection de documents (ou *dataset* ou encore *corpus*).

1.2.2.2. Besoin en information et requête

La requête est une expression approximative du besoin en information de l'utilisateur. Ce dernier est un ensemble de mots clés d'informations que l'utilisateur recherche. Les requêtes soumises au SRI par les utilisateurs peuvent ne pas refléter leurs besoins en information [Hammache, 2013].

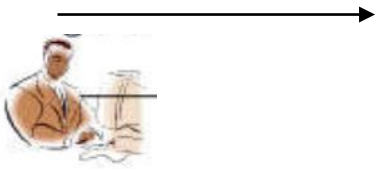

Figure 1.1: Architecture générale d'un SRI [Bourane and Berrekbia, 2017]

1.2.3. Processus de recherche d'informations

Pour répondre à une requête utilisateur, un SRI met en œuvre un certain nombre de processus pour le but d'établir une correspondance pertinente entre l'ensemble des documents disponibles d'une part, et celui de la requête utilisateur d'une autre part.

Le schéma ci-dessous (Figure 1.2) montre que le processus de recherche d'information se décompose en trois processus comme suit :

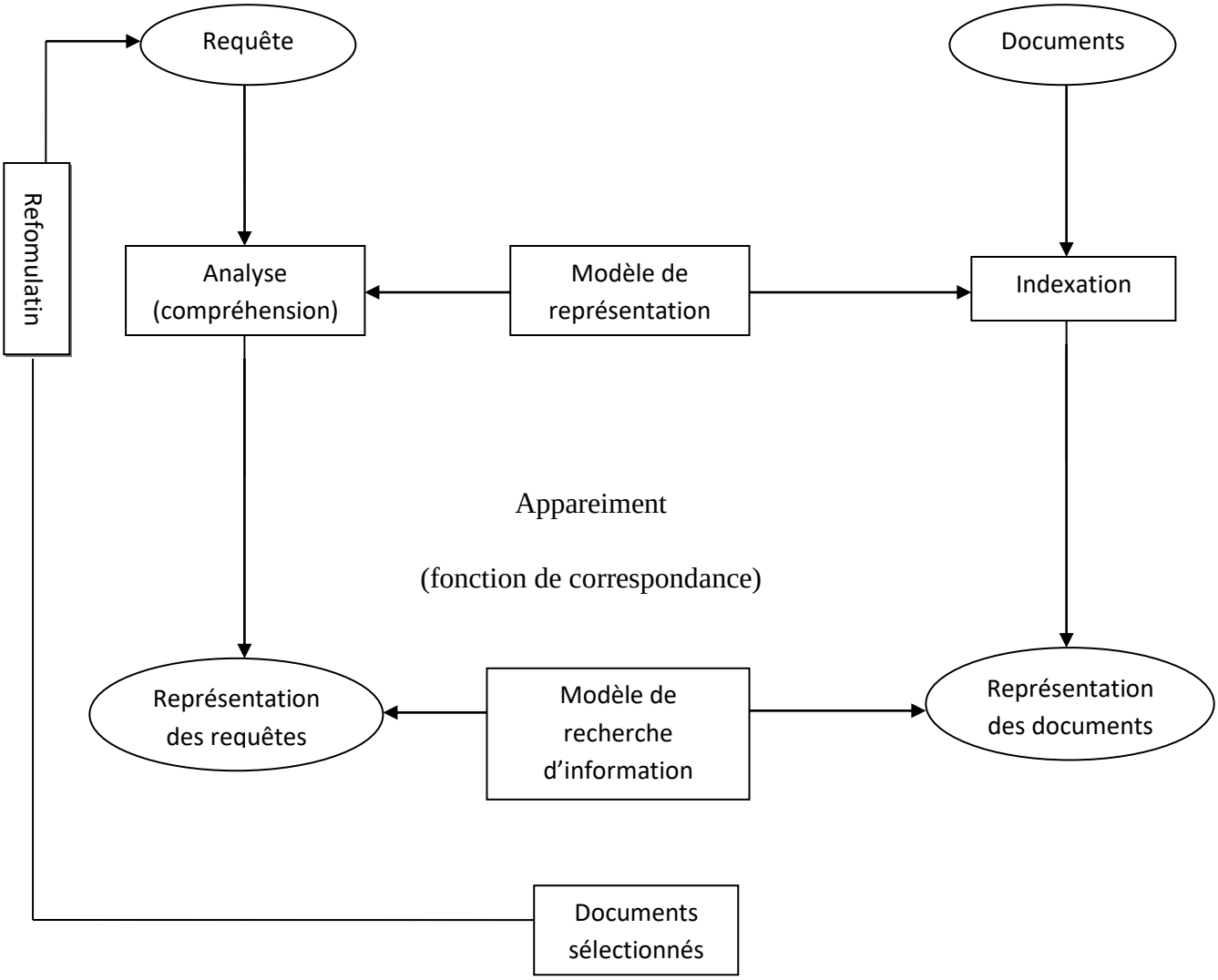

Figure 1.2: Processus de RI [Bouabdellah and Benmansour, 2012]

1.2.3.1. Processus d'indexation

L'indexation fait référence à la façon dont les documents sont restructurés pour être recherchés, et elle consiste à associer à chaque document une liste de mots clés appelée aussi descripteur [Kadri, 2008].

Les groupes de mots forment ce que l'on appelle un thesaurus. Ce dernier inclut des relations de type linguistiques (équivalence, association, hiérarchisation) et statistiques (pondération). Ces termes peuvent être extraits de trois manières :

- **Manuelle :** Chaque document est analysé par un spécialiste du domaine correspondant ou par un documentaliste. Elle est basée sur un vocabulaire contrôlé (lexique, liste hiérarchiques, thésaurus, ontologie).

- **Semi-manuelle :** La tâche d'indexation est réalisée ici conjointement par un programme informatique et un spécialiste du domaine. Le choix final reste au spécialiste du domaine correspondant ou documentaliste.
- **Automatique :** C'est le SRI qui génère les indexes des documents. Elle passe par plusieurs étapes pour créer d'une façon automatique l'index. Ces étapes sont: l'analyse lexicale, l'élimination des mots vides, la normalisation (lemmatisation ou radicalisation), la sélection des descripteurs, le calcul de statistiques sur les descripteurs et les documents (fréquence d'apparition d'un descripteur dans un document et dans la collection, la taille de chaque document, etc.) et enfin la création de l'index [Bourane and Berrekbia, 2017]

1.2.3.2. Appariement document-requête

La correspondance entre les termes de la requête d'un utilisateur et ceux des documents s'effectue au niveau de l'appariement document-requête. Afin de réaliser cela, le SRI représente le document et la requête avec un même formalisme, puis le SRI les compare. Le résultat de cette comparaison se traduit par un score qui détermine la probabilité de pertinence (degré de similarité ou degré de ressemblance). Ce score est calculé à partir d'une fonction notée par *Retrieval Status Value*[Hammache, 2013].

1.2.3.3. Modèles de recherche d'information

L'interprétation des termes choisis par l'indexation est garantie par un modèle, et ce dernier permet de définir une méthode de comparaison entre la représentation du document et de la requête, et ceci afin de déterminer leur degré de correspondance .Ces modèles ont en commun le vocabulaire d'indexation basé sur le formalisme mots clés et diffèrent principalement par le modèle d'appariement requête-document. Le vocabulaire d'indexation V = {ti}, i ∈ {1, ..., n} est constitué de *n* mots ou racines de mots qui apparaissent dans les documents. Un modèle de RI est défini par un quadruplet (D, Q, F, R (q, d)), où *D* est l'ensemble de documents, *Q* est l'ensemble de requêtes, *F* est le schéma du modèle théorique de représentation des documents et des requêtes et *R(q, d)* est la fonction de pertinence du document *d* à la requête *q* [Abbasi and Meftah, 2013].

Généralement, il y a trois types de modèles de RI : le modèle booléen, modèle vectoriel et modèle probabiliste.

a. modèle booléen : c'est le modèle le plus ancien et également le plus simple en RI. Dans ce modèle, les documents et les requêtes sont représentés par des ensembles de mots clés.

Chaque document est représenté par une conjonction logique de termes non pondérés qui constitue l'index du document.

b. modèle vectoriel : ce modèle est basé sur une formalisation géométrique. En effet, les documents et les requêtes sont représentés dans un même espace, défini par un ensemble de dimensions, où chaque dimension représente un terme d'indexation.

c. modèle probabiliste : il calcule la probabilité de la pertinence d'un document *D* par rapport à une requête *R*, et ceci dans le but de séparer les documents pertinents des autres non pertinents dans une collection. L'idée de base, dans ce modèle, est de tenter de déterminer les probabilités conditionnelles selon qu'un terme de la requête est présent, dans un document pertinent ou dans un document non pertinent.

1.2.4. Recherche d'information sur le web

Les problématiques posées en RI sur le web sont identiques à celles posées par la RI classique (indexation, appariement, etc.). Or, le web se diffère sur plusieurs points par rapport aux autres ressources documentaires, où on cite parmi les facteurs distinctifs, le volume du web et la dispersion [Hammache, 2013].

1.3. Catégorisation des textes

La catégorisation des documents textuels est une tâche qui fait partie de l'extraction d'information (IR). Elle consiste à classifier de manière automatique des documents suivant certains critères (e.g. thème du texte, son style, etc.), d'où elle est devenue un domaine de recherche très actif qui attiré l'intérêt de la communauté scientifique. C'est dans cette perspective que plusieurs travaux de recherche se concentrent ces dernières années.

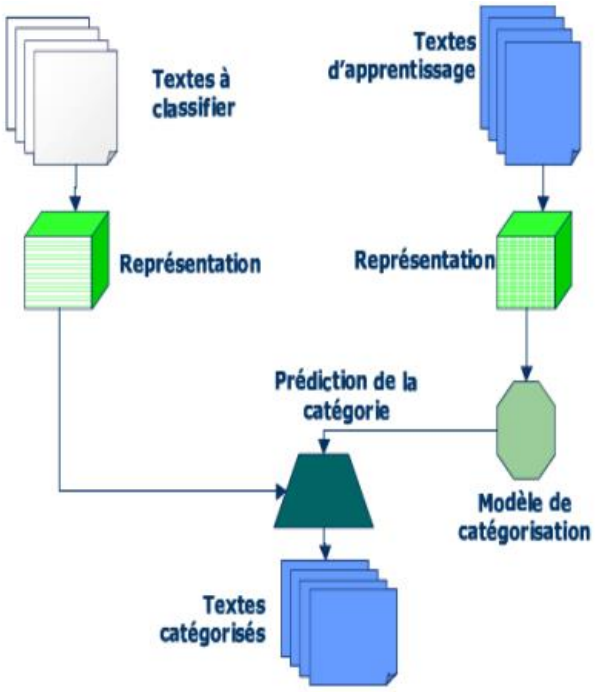

Figure 1.3: Tâche de catégorisation [Jalam, 2003]

1.3.1. Historique

Le domaine de la catégorisation des documents textuels remonte au début des années 1620, où Gabriel Naudé proposa un classement selon cinq grands thèmes : théologie, jurisprudence, histoire, sciences et arts, belles lettres [Matallah, 2011].

L'encyclopédie de Diderot (parue entre 1751 et 1772) est organisée selon l'ordre alphabétique avec des renvois associatifs, alors que celle de Panckoucke (parue de 1776 à 1780) suit une organisation méthodique selon un ordre arborescent [Fayet and Scribe, 1997]. Le système de catégorisation par thème, apparu dès le début de l'écriture, et institutionnalisé à Alexandrie conduisit à la création d'un système de classification par Dewey en 1876 «universel» [Matallah, 2011].

En 1787, peu après la guerre d'indépendance de l'Amérique, trois hommes politiciens (i.e. James Madison, Alexander Hamilton et John Jay) ont publié 85 articles anonymes sous le nom « Publius » dans le but d'adopter la nouvelle constitution, ainsi que de discuter plusieurs problèmes généraux de la politique [Federalist, 2012]. Parmi ces articles anonymes, 12 articles, dont les auteurs ne sont pas reconnus, ce qui a fait l'objet d'une recherche à reconnaitre l'auteur de chacun de ces articles, ainsi que l'auteur de chacun des paragraphes constituant ces derniers en étudiant les ressemblances du style d'écriture.

Jusqu'au début des années 80, pour construire un classificateur, il fallait consacrer d'importantes ressources humaines à cette tâche. Après la révolution industrielle, d'après Sanderson le traitement numérique des documents textuels est apparu la première fois avec l'introduction de la machine Univac. Ce dernier a été proposée par Holmstrom en 1948 dans la conférence UK's Royal Society [Sanderson and Croft, 2012].

Au début des années 90, les travaux proviennent essentiellement de la communauté de RI. En effet, les méthodes de numérisation, les algorithmes de classification et les méthodologies de test ont été adaptés à la catégorisation textuelle (CT) en particulier au cours des conférences TREC (Text REtrieval Conference).

Vue la puissance des machines actuelles et le grand progrès technologique, ainsi que l'énorme quantité d'information disponible, une nouvelle branche est apparue, appelée Big Data, traitant d'énormes quantités de données.

1.3.2. Domaine de la catégorisation

L'objectif de la CT est de classer les documents dans des domaines qui ont été définies. On peut citer, la catégorisation par auteur, par thème, par langue, par genre, etc.

1.3.2.1. Catégorisation des documents textuels par auteur

La catégorisation des documents textuels par auteur (stylométrie) est l'une des catégories les plus utilisées dans ce domaine. L'un des premiers qui a utilisé la stylométrie pour l'étude des pièces de Shakespeare en 1887 (reconnaitre le vrai auteur des pièces) est le célèbre physicien Mendenhall [Stamatatos, 2009]. L'identification de l'auteur était basée sur l'exploitation des distributions des fréquences des mots de différentes longueurs [Mendenhall, 1887]. Parmi les plus anciens et influents travaux d'identification d'auteur, c'est le travail de Mosteller sur l'identification de l'auteur des douze articles « Federalist Papers » [Mosteller et al., 1963] [Mosteller and Wallace, 1964].

1.3.2.2. Catégorisation des documents textuels par thème

Le deuxième sous domaine de catégorisation des documents textuels est celui de la catégorisation par thème ou par sujet. La première utilisation de cette méthode fut dans la bibliothéconomie afin d'archiver les documents traitant d'un même sujet et un contexte similaire. L'expansion de l'internet et des différents moyens numériques a rendu impossible la catégorisation manuelle, et vu que certains documents abordent plusieurs thèmes (e.g.

relation entre la politique et l'économie), il est souvent difficile de cerner le thème [Abainia, 2016].

1.3.2.3. Catégorisation des documents textuels par langue

Le troisième type de catégorisation des documents textuels est celui de la catégorisation par langue, et qui est fondu sur la reconnaissance de la langue d'un texte donné. Il était considérée comme facile par certains ou problème résolu par d'autres [Xia et al., 2010] [McNamee, 2005]. Cependant dans nos jours, la reconnaissance de la langue dans le traitement des documents multi-langues ou de très courts documents (comme les messages Twitter) représente un défi scientifique [Baldwin et al., 2010].

1.3.2.4. Autres domaines de catégorisation des textes

Dans le domaine de la catégorisation des documents textuels, il existe aussi trois sous-domaines peu ou rarement abordés par les chercheurs tel que la catégorisation par genre (poème, article scientifique, article de presse, etc.) et par âge (tranche d'âge). Enfin, la catégorisation par opinion (avis des personnes) est due au développement des réseaux sociaux et du e-marketing.

1.3.3. Les systèmes de catégorisation

Actuellement, il existe plusieurs systèmes de catégorisation dont la même tâche, mais avec différents points de vue. L'objectif de la CT est de classer de façon automatique les documents dans des catégories qui ont été définies soit préalablement par un expert (classification supervisée ou catégorisation), soit de façon automatique (classification non supervisée ou clustering).

1.3.3.1. Catégorisation (supervisé)

La catégorisation de textes correspond à la procédure d'affectation d'une ou de plusieurs catégories ou étiquettes prédéfinies à un texte. Des experts fournissent un ensemble de documents qui constituent des exemples positifs qui servent pour l'apprentissage. Plus précisément, c'est la tâche qui consiste à organiser et hiérarchiser les connaissances (Figure 1.4) qui sont structurées sous forme d'un graphe représentant les relations sémantiques entre les concepts (connaissances).

Par ailleurs, cette problématique a dernièrement trouvé de nouvelles applications dans les domaines du traitement du langage tels que : l'affectation de sujets en recherche d'information, l'aide de l'utilisateur pour l'indexation de documents [Hayes and Weinstein, 1990], la veille technologique, le filtrage personnalisé des documents intéressant un internaute connaissant ses préférences de sujets (catégories) [Lang, 1995], le routage de textes (tels que le courrier) et l'amélioration de la recherche sur le web [Armstrong and al., 1995], et enfin l'organisation des sources textuelles plus nombreuses.

Cette technique utilise largement des méthodes issues de l'apprentissage automatique (Naïve bayes, K plus proches voisins, arbres de décision, machines à vecteurs support, réseaux de neurones, etc.).

1.3.3.2. Clustering (non supervisé)

La classification non supervisée consiste à trouver de manière automatique une organisation cohérente à un groupe de documents homogènes pour construire des regroupements cohérents appelé clusters. De plus, ce processus regroupe uniquement les documents similaires selon un contexte donné (e.g. regroupement par thème, auteurs, etc.) sans connaitre les clusters au préalable.

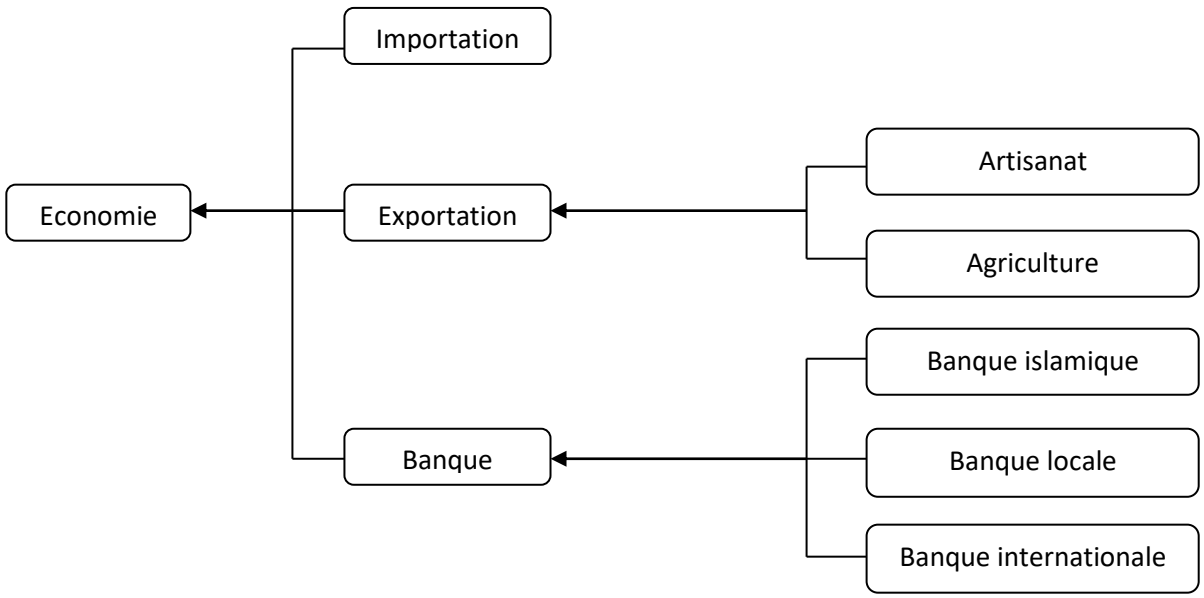

Figure 1.4: Exemple d'ontologie des termes dérivés du domaine de l'économie

Les techniques pour réaliser de tels regroupements constituent un domaine d'étude très riche, qui a donné lieu à de multiples propositions dont le recensement n'est pas l'objet de ce document. L'apprentissage non supervisé est utilisé dans plusieurs domaines tels que : la médecine, la biologie, le traitement de la parole, etc. Il existe plusieurs types d'algorithmes d'apprentissage non supervisé tels que les algorithmes de partitionnements et les algorithmes de classification hiérarchique.

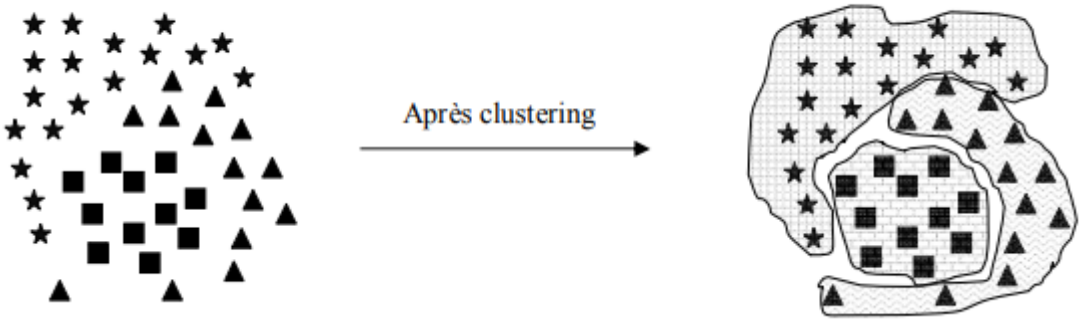

Figure 1.5: Exemple de clustering des formes géométriques

1.3.4. Difficultés de la Catégorisation des Documents Textuels

La catégorisation des documents textuels fait face à quatre difficultés majeures, et qui sont en l'occurrence la source des documents, l'évolution linguistique et technologique, le nombre de catégories utilisé dans le système dont l'augmentation complique la catégorisation [Stamatatos, 2010], et enfin la ressemblance et la relation entre les catégories qui altèrent les performances de la catégorisation [Abainia, 2016].

1.3.4.1. Source des Documents

La source des documents, lors d'une évaluation d'une approche ou d'un algorithme, est un des paramètres les plus importants dans la catégorisation textuelle [Abainia, 2016]. Parmi les sources déjà utilisées on peut citer les pages web (i.e. pages HTML), les articles d'actualités (news), les articles scientifiques, les livres, des messages de forums, les messages de chat, les messages de réseaux sociaux, et enfin les transcriptions de paroles. Aucune de ces sources n'est totalement fiable et tout dépend de l'application de catégorisation (ou spécialité). Certains chercheurs ont prétendu que le problème d'identification de la langue est résolu. Mais toutefois, la source de leurs documents était des collections d'articles de news alors que ces derniers (de même que les livres) sont bien

structurés et bien écrits, d'où il facilite la tâche de la catégorisation. Ce type de catégorisation est aussi facilité par l'utilisation des pages web du fait qu'on y trouve des méta-informations [Martins et al., 2005]. Quant à la catégorisation par thème, selon la source des documents, on peut avoir des taux de catégorisation élevés (articles scientifiques) ou moins élevés (messages de forums, de chat et de réseaux sociaux) [Abainia et al., 2014].

1.3.4.1.1. Forums de Discussion

Un forum de discussion est accessible, sur le net, à tout le monde et permet à quiconque de poster des messages et de communiquer avec tout le monde. Il existe deux types de forum dont le premier est organisé par thèmes et sous thèmes et le deuxième par domaines globaux (ex. économie, politique, etc.). Dans le premier les discussions sont classées par date du dernier message alors que dans le deuxième elles le sont par vote [Abainia, 2016]. Généralement, les utilisateurs de ces forums se font connaitre par des pseudos et non par leur vrai nom ce qui leur permet de faire du plagiat. Selon l'utilisateur, le message peut contenir plus ou moins de bruits textuel tels que : abréviations, style d'écriture, fautes d'orthographe, fautes de frappe, caractères représentant des émotions (à la place des émoticônes), balises HTML dues aux mauvaises actions, URLs, citations des autres personnes, citations dans d'autres langues, textes publicitaires etc. [Abainia, 2016]

Quant aux forums de discussion Arabes et à cause des différents dialectes, on constate d'autres types de bruits que ce soit dans le vocabulaire ou dans le style d'expression [Abainia et al. 2015e]

1.3.4.1.2. Réseaux sociaux

Dans le réseau social le plus utilisé à travers le monde (Twitter), les utilisateurs partagent des informations et des idées instantanément en utilisant au maximum 280 caractères. Les informations récoltées à travers ce réseau et les informations additionnelles relatives aux utilisateurs (données GPS) ont poussé les chercheurs à s'investir dans l'extraction d'information et le traitement du langage naturel (TALN) dans ce type de source de textes [Abainia, 2016].

- الفيروس **بيتشال** عادى انا **شلتة** المشكلة ف تشفير الملفات والصور للأسف **مشهتتفتح**
- أدمن الله يرحم الوالدين **شنو** الحل مع القنوات لي **كيتقطع** فيهم بحال سكاي سبور
- هي قناة rti 1**هتذيع** كأس الامم و لا **هتتشفر**
- السلام عليكم **هوا** كاس افريقيا **هيتذاع** على اموس و لا قنوات **ايه**

Figure 1.6: Exemples de quelques messages extraits d'un forum de discussion Arabe

@lequipe :

Aujourd'hui sur **@Snapchat** on revient sur les plus beaux moments des finales de Roland-Garros avec Borg, Hingis, Williams, Kuerten… http://ow.ly/Bl1a50uyQqC **#rolandgarros#RG2019**

Figure 1.7: Exemple d'un message Twitter

En plus, du fait que l'utilisation de Twitter comme source de catégorisation présente les mêmes difficultés que l'utilisation des forums de discussion. Donc, la catégorisation en utilisant cette source est parfois plus compliquée (voire impossible) dans certains domaines de catégorisation, et ceci à cause des textes qui sont très courts.

1.3.4.2. Evolution linguistique et technologique à travers le temps

Dans l'évolution linguistique et technologique qui fait aussi face à la catégorisation des documents textuels, il est parfois impossible de catégoriser certains sous-domaines en raison de leur énorme dépendance du temps.

1.3.4.2.1. Evolution linguistique

L'échange interculturel et les déplacements commerciaux sont les causes principales de l'évolution et du changement des langues au fil du temps. Cette évolution est parfois bien visibles sur certains voyageurs et /ou commerçants. Grâce à cette richesse linguistique, les linguistes ont pu simplifier leur langue locale et ont permis sa compréhension aux étrangers. Entre le français du 17ème siècle et du 21ème siècle, par exemple, il y a des différences dans le style d'écriture et le vocabulaire, pas mal de termes ne sont plus utilisés, en revanche des termes d'autres langues y ont été introduits (e.g. termes anglais) [Abainia, 2016]. L'âge d'or islamique qui provoqua d'énormes échanges interculturels ce qui entraina la simplification linguistique et la vulgarisation scientifique a eu un effet considérable sur le monde entier en enrichissant le vocabulaire des occidentaux et des orientaux (traduction des manuscrits scientifiques de médecine, chimie, hydraulique, mécanique, etc.). Enfin, le dernier facteur ayant eu une influence sur l'évolution linguistique, en particulier sur les

langages locaux, c'est la colonisation, notamment les colons Allemands, Britanniques, Français, Portugais et Espagnols qui ont généralement imposé leur langue au pays colonisé. Les algériens en sont un exemple typique puisque leur langage est emprunté des langues de différents colons (Figure 1.8).

ياك ***الميساج*** *في الفايسبوك باطل تحية للناس لي تنكر العشرة*

Figure 1.8: Exemple de message extrait de Facebook contenant des mots Français écrits en Arabe

Au vu de l'évolution linguistique, la catégorisation automatique des textes est difficilement réalisable surtout pour des textes de différentes époques.

1.3.4.2.2. Evolution technologique

L'évolution linguistique a été essentiellement boostée par le développement technologique qui créa de nouveaux termes scientifiques et techniques, où ces derniers ont créé au même temps un handicap de plus à la catégorisation. Par exemple, le conseil d'une personne à une autre pour la recherche d'une solution à un problème était entre 2000 et 2019 présenté respectivement de la manière suivante : « connectez sur internet, et tapez ce terme dans le moteur de recherche » et « googlez ce terme », ce qui réduit toute une phrase à un seul mot [Abainia, 2016]. Le style d'écriture a aussi été influencé par la technologie, et ce grâce à la télécommunication et l'inter-échange d'informations (ex. chat). A une époque, l'utilisation de certains mots ou énoncés était réservée à une communauté restreinte, mais toutefois grâce au développement technologique et les nouveaux moyens de communication, beaucoup de termes sont utilisés par tout le monde. Van Herk (chercheur en psycholinguistique) a affirmé que : *«Les gens craindraient les effets nuisibles de la technologie depuis longtemps. Au milieu du 19e siècle, un auteur anonyme avait prédit que les dialectes britanniques allaient bientôt disparaitre à cause des nouvelles technologies de l'époque qu'étaient le train à vapeur et le télégraphe.»*. Donc, d'après l'évolution linguistique et technologique, la catégorisation avec ses différents sous domaines, doit tenir compte de la période ou le temps dans lequel le texte est écrit.

1.3.5. Applications de la catégorisation

La catégorisation est une technique utilisée dans plusieurs domaines, à cause de sa capacité prédictive qui la rend rapide et efficace. Parmi les applications dans lesquelles la classification est utilisée, l'identification de la langue, la reconnaissance d'auteur, la catégorisation de documents multimédia, et bien d'autres.

On cite également la détection de spams (les courriers indésirables) pour ensuite les supprimer, le routage qui permet d'affecter un document à une ou plusieurs catégories parmi un ensemble (diffusion sélective d'information). Lors de la réception d'un document l'outil choisit à quelles personnes le faire parvenir en fonction de leurs centres d'intérêt. Ces centres d'intérêt correspondent à des profils individuels [Liddy et al., 1994].

1.4. Conclusion

La catégorisation de textes s'est avérée au cours des dernières années comme un domaine majeur de recherche pour les entreprises comme pour les particuliers. Dans ce chapitre nous avons présenté les principales notions et concepts de la recherche d'information, des systèmes de recherche d'information et ceux des outils de recherche sur le web. Ainsi nous avons exposé quelques notions de la catégorisation des textes, un bref historique de cette dernière est présenté, avec les domaines et les systèmes de la CT.

A travers les différentes sections que nous avons présentées, nous concluons que la recherche d'information, s'attache à définir des modèles et des systèmes afin de faciliter l'accès à un ensemble de documents. Le but est de permettre aux utilisateurs de retrouver les documents dont le contenu répond à leur besoin en information, il s'agit donc de catégoriser les documents.

Chapitre 2 : Prétraitement de la catégorisation des textes arabes

2.1. Introduction

Le volume d'informations stockées électroniquement augmente rapidement grâce aux nouvelles technologies et à Internet. Ce qui impose à établir des techniques efficaces pour répondre à une requête de recherche avec une manière pertinente parmi ces grandes collections de données. [Zeroul and Lakhouaja, 2017]

Les chercheurs ont montré que les tâches de stemming et la lemmatisation augmentent considérablement la performance de RI dans plusieurs langues telles que l'anglais, portugais, russe, Chinois, néerlandais et allemand. La plupart des chercheurs expliquent les limites de leurs systèmes de RI en arabe par la morphologie de l'arabe, qui diffère par la structure des affixes des langues indo-européennes. [Zeroul and Lakhouaja, 2017]

L'arabe est parlé dans plus de 22 pays, du Maroc jusqu'à l'Iraq et dans toute la péninsule arabe. C'est la première langue pour plus de 250 millions de personnes, et en étant la langue du Coran, elle est devenue la langue d'une civilisation et ne sert plus seulement à désigner les seuls habitants de la péninsule arabe qui la parlaient. [Chouchoui and Brahimia, 2016]

Dans ce chapitre, nous présentons d'abord les particularités de la langue arabe et sa morphologie. Par la suite, nous nous attardons d'une part, sur les prétraitements nécessaires, et d'autre part sur les techniques de lemmatisation et ses différentes méthodologies d'évaluation des lemmatiseurs. [Cheragui et al., 2015]

2.2. Langue arabe et ses particularités

L'arabe est doté d'une richesse morphologique due principalement à ses propriétés de dérivation et de forte flexion. Malgré cette richesse morphologique est perçue par les linguistes comme un point fort, cette vision n'est pas partagée par les chercheurs qui travaillent sur l'automatisation de la langue arabe, car elle est même vue comme le point névralgique de plusieurs problèmes comme l'ambiguïté (que se soient morphologique, syntaxique et même sémantique). Ce qui rend le processus de lemmatisation d'une importance primordiale dans la phase d'analyse.

Par rapport aux autres langues, l'arabe a des caractéristiques qui lui sont propres. L'arabe qui s'écrit déjà de droite à gauche possède un alphabet abjad [Khemakhem, 2006] qui se compose essentiellement de consonnes. En outre, elle a la particularité des longues voyelles « ا », « ي » et « و» dont les deux dernières sont des réalisations contextuelles des « ي » et « و» glides. Dans l'écriture arabe on peut aussi inclure des signes annexes

facultatifs, hormis pour le coran, qui servent à noter les trois voyelles brèves (ً (a), "ُ " (u) et "ِ " (i)). On y trouve aussi "ْ " (sukūn) pour signifier l'absence de voyelle et la "ّ " (šadda) pour la gémination des consonnes[Bourezg, 2017] . Il est à noter qu'en arabe un mot sans article ou complément de nom (c'est-à-dire indéfini) prend généralement les désinences "ً " (an), "ٌ" (un) ou "ٍ " (in), dites nounation ou *tanwīn*. Ces dernières sont désignées par des diacritiques spéciales marquées par le redoublement du signe de la voyelle qui précède le suffixe « n » attendu en fin de mot. L'écriture arabe est dite monocamérale, car on n'y trouve pas les notions de lettre capitale et lettre minuscule. On dit aussi que l'arabe est une langue semi-cursive car elle ne comprend que six lettres qui ne s'attachent jamais à la lettre suivante, sinon la plupart des lettres s'attachent entre elles et leurs graphies diffèrent selon qu'elles soient précédées, suivies d'autres lettres ou qu'elles soient isolées [Bourezg, 2017]

Isolé	Initiale	Médiane	Finale
ق	قـ	ـقـ	ـق
	قرآن	القرآن	غسق

Tableau 2.1 : Représentation graphique de différentes formes de la lettre « ق » (qaf)

	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 2.2 : Exemple de dérivation des mots à partir de la racine « كتب » et « شعر »

2.3. Structure morphologique d'un mot arabe

La langue arabe, contrairement à l'anglais, a une structure et une morphologie plus compliquées. La forme des noms est déterminée par plusieurs critères tels que le genre, le nombre et les cas grammaticaux. De plus, les noms arabes ont un grand nombre de variantes et certaines d'entre elles peuvent être complexes à cause des préfixes, suffixes et infixes [Chen and Gey, 2002]. [Al-Abweeny et al., 2018]

La plupart des mots arabes sont générés à partir de milliers de racines. Dans cette section, nous éclaircirons les concepts de base pour éviter toute confusion quant à leur signification [Zeroul and Lakhouaja, 2017]

2.3.1. Catégories des mots arabes

La classification des mots de la langue arabe trouvée dans la plupart des références de la linguistique arabe repose essentiellement sur la distinction entre trois catégories ou trois sous-ensembles : verbes, noms et particules. Ce classement montre, rapidement, ses limites quand il s'agit d'un traitement informatisé de la langue. [Saadane, 2015]

2.3.1.1. Verbe

Un verbe est une entité portant un sens dépendant du temps et qui exprime une action, ou un événement. La plupart des mots arabes se dérivent d'un verbe de trois lettres (*trilittéral*) comme le verbe «**دخل** » (*dakhala* signifie entrer), ainsi on trouve des mots dérivés de verbes de quatre consonnes comme le verbe « **دَحْرَج** » (*dahrağa* signifie glisser ou faire glisser). Chaque verbe est donc la racine d'une famille de mots. [Saadane, 2015]

Autrement dit, le verbe est un mot qui se conjugue, et qui indique un état ou une action faite ou subie par le sujet. Nous pouvons classer les verbes arabes selon plusieurs critères :

A. Critère de temps: il existe trois types.

- **L'accompli:** correspond au passé et se distingue par des suffixes par exemple pour le pluriel féminin on a **كتبن** (elles ont écrit), et pour le pluriel masculin on a**كتبوا**(ils ont écrit).
- **Inaccompli présent:** les verbes conjugués à ce temps se distinguent par les préfixes. Pour notre exemple, au 3ème personne du singulier masculin on obtient **يفتح** (il ouvre) ; et pour le féminin singulier on obtient **تفتح** (elle ouvre)**.**
- **Inaccompli futur:** la conjugaison d'un verbe au futur nécessite d'ajouter l'antéposition au début du verbe conjugué à l'inaccompli. En ajoutant l'antéposition à notre exemple **س**, on obtient **سيفتح** (il ouvrira), qui désigne le futur. On peut également ajouter l'antéposition **سوف,** on obtient **سوف يفتح** (il va ouvrir). [Benhalima, 2017]

B. Sens et transitivité du sujet au complément: il y a deux types.

- **Intransitif:** ce sont les verbes qui n'admettent ni complément d'objet direct, ni indirect comme dans la phrase **نام الطفل** (l'enfant a dormi).

- **Transitif: l**es verbes transitifs peuvent avoir besoin d'un ou plusieurs compléments pour compléter le sens de la phrase, comme l'exemple dans la phrase **كتب الشاعر قصيدة** (l'écrivain a écrit un poème). [Mesfar, 2008]

C. Modes: il y a aux deux types.

- **la voix passive:** une phrase est à la voix passive quand le sujet subit l'action
- **la voix active:** une phrase est à la voix active quand le sujet fait l'action

D. Nombre des consonnes de la racine: la majorité des verbes (environ 85%) sont formés à partir de racines de 3 lettres et le reste entre les racines de 4 et 5 lettres. Ces racines peuvent donner plusieurs mots grâce à des transformations morphologiques selon les schèmes.

E. Schème et nombre de consonnes constituant la structure verbale:

- **verbes nus:** sont formés seulement par les consonnes de leurs racines et des voyelles brèves.
- **verbes augmentés:** sont dérivés des trois consonnes de la racine par modification des voyelles, en redoublant la deuxième lettre de la racine, par adjonction et même par intercalation d'affixes. Les verbes dérivés se conjuguent avec les mêmes préfixes et suffixes que le verbe nu. Les verbes trilitères peuvent être augmentés au maximum par trois lettres et les verbes quadrilitères par deux lettres. Alors, la longueur maximale d'un verbe arabe est de 6 lettres. [Benhalima, 2017]

F. Conjugaison: il existe le conjugué, le non conjugué et l'invariant

G. Nature des consonnes:

- Les verbes sains (**صحيح**) dont les racines ne contiennent pas des lettres (ي, و, ا) qui sont défectueuses.
- Les verbes défectueux (**معتل**): dont les racines contiennent une ou deux lettres défectueuses qui causent des altérations importantes au cours de la conjugaison. [Benhalima, 2017]

Lettre	Nom	Translation	Commentaire	ك	kâf	k	Lettre lunaire
ء	Hamzã	’		ل	lâm	l	Lettre solaire
ا	Alif	Ã	Voyelle longue	م	mîm	m	Lettre lunaire
ب	Bâ’	B	Lettre lunaire	ن	nûn	n	Lettre solaire
ت	Tâ’	T	Lettre solaire	و	wâw	w / ŭ	Lettre lunaire(*) /voyelle longue
ث	Tâ’	T	Lettre solaire	ي	yâ’	y / ï	Lettre lunaire/ Voyelle longue
ج	Jîm	Ğ	Lettre lunaire	ـَ	fathã	a	Voyelle brève
ح	hâ’	H	Lettre lunaire	ـُ	dammã	u	Voyelle brève
خ	Xâ’	H	Lettre lunaire	ـِ	kasrã	i	Voyelle brève
د	Dâl	D	Lettre solaire	ـً	tanwîn	ã / an	Voyelle brève/ Tanwîn
ذ	ḏâl	D	Lettre solaire	ـٌ	tanwîn	ũ / un	Voyelle brève/ Tanwîn
ر	Râ’	R	Lettre solaire	ـٍ	tanwîn	ĩ / in	Voyelle brève/ Tanwîn
ز	Zây	Z	Lettre solaire	ة	tâ’marbûtã	ä(at en annexion)	
س	Sîn	S	Lettre solaire	ى	Alifmaqsûrã	ą	
ش	Sîn	S	Lettre solaire	آ	Alifmamdûdã	å	
ص	Sâd	S	Lettre solaire	أ	hamzã	å	
ض	Dâd	D	Lettre solaire	أ	hamzã	ů	
ط	tâ’	T	Lettre solaire	إ	hamzã	ů	
ظ	zâ’	Z	Lettre solaire	ؤ	hamzã	ẃ	
ع	ʿayn	ʿ	Lettre lunaire	ئ	hamzã	ỳ	
غ	Gayn	G	Lettre lunaire	ـْ	sukun		Non transcrit
ف	Fâ’	F	Lettre lunaire	ـّ	šaddă	Lettre redoublée	Signe de gémination
ق	Qâf	Q	Lettre lunaire				

Tableau 2.3 : Tableau de translittération de l’alphabet arabe [Dariouache, 2016]

2.3.1.2. Nom

L'élément désignant un être, un objet ou un état qui exprime un sens indépendant du temps. La fonction du nom est sa relation avec un mot ou une expression de la phrase.

Le système nominal de l'arabe admet deux catégories, ceux qui sont dérivés de la racine verbale comme **مكتبة**(bibliothèque) de la racine **كَتَبَ** et ceux qui ne le sont pas comme les noms propres et les noms communs **غزال**(gazelle). [Benhalima, 2017]

La déclinaison des noms se fait selon les règles suivantes:

- ***Féminin singulier:*** On ajoute le **ة**, exemple **مسلم** musulman devient **مسلمة** musulmane.
- ***Féminin pluriel :*** De la même manière, on rajoute pour le pluriel les deux lettres **ات**, exemple (**كاتب** écrivain) devient (**كاتبات** écrivains).
- ***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 **مسلم** devient **مسلمين** ou **مسلمون** (musulmans).
- ***Pluriel irrégulier:*** il suit une diversité de règles complexes et dépend du nom. A titre d'exemple : (**طفل** un enfant) devient (**أطفال** des enfants). [Dariouache, 2015]

Catégorie	Dérivation	Conjugaison	Sous-catégorie	Exemple
Nom	Dérivation nel Irrégulier	Nom conjugable	Adverbe	أينَ, حَيثُ, قبلَ
			Nom de voix	كخ, نخ
			Nom de verbe	هَيْهَات, آه, أف
			Pronom personnel (affixé ou isolé)	هُوَ,أنا, تُ, تُنَ
			Pronom intérogatif	كَيْفَ,مَتَى, مَا
			Pronom conditionel	مَنْ, إذا
			Pronom allutif	كَمْ, كَأيّ
		Conjugable	Pronom relatif	الذي, التي
			Nom de nombre	ثلاثة, وَاحد, خَمْسَة
			Pronon démonstratif	هَذا, هَذه
			Nom propre	مُحَمَّدُ, هِنْدُ, صَحْرَاءُ
			Nom commun	قَلَمٌ, أَرْنَبٌ, رَجُل
	Dérivation nelrégulier	Conjugable	Masdar	كِتَابَة, القَتْلُ
			Participeactif	قاتلٌ, شاربٌ
			Participepassif	مَكْتوبٌ, مَضْروبٌ
			Nom d’une fois	جَلْسَة, ضَرْبَة
			Nom de maniére	نظرة, جلسة
			Nom de temps	مَغْربُ
			Nom de lieu	مَكْتَبٌ, مَقْبَرَةٌ
			Nom d’instrument	مِطْرَقَة, مِسْمَارٌ
			Adjectif	حَسَنٌ, جَميلٌ, بَطَلٌ
			Elatif	أحْسَنُ, أفْضَلُ
			Nom diminutif	كُتَيِّبٌ, شُوَيْعِرٌ
			Nom de relation	جَزَائري, مِصْري
			Intensif	قَتَّالٌ, غَوَّاصٌ

Tableau 2.4 : classement des sous-catégories de noms[Benhalima, 2017]

2.3.1.3. Particule

Les particules sont des lemmes invariables et en nombre limité. 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 telles que l’introduction, explication, conséquence, etc. [Saadane, 2015].

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

On peut distinguer plusieurs types, parmi lesquels on peut citer :

- Particules préposition : exemple (حتى ، عن، ل ، ب ، ك)
- Particules de coordination : exemple (أو، ثم، ف، و)
- Particules interrogatives : exemple (أ، هل، ما)
- Particules d'affirmation : exemple (،أجل، بلى ، نعم)
- Particules de négation : exemple (لم، لن، ال)
- Particules distinctive : exemple (أي)
- Particules relatives : exemple (أم)
- Particules de futur : exemple (س، سوف)
- Particules conditionnelles : exemple (لو، إن)

2.3.2. Morphologie arabe

La morphologie est un domaine de la langue qui permet de faire la description des règles régissant la structure du mot et ses changements par l'ajout de particules pour former des dérivés et des formes flexionnelles. La nature morphologique de la langue arabe est complexe. Puisque elle est caractérisée par le phénomène de dérivation et d'inflexion, la plupart des mots arabes sont générés à partir de milliers de racines. Dans cette section, nous fournissons une clarification des concepts de base pour éviter toute confusion quant à leur signification.

2.3.2.1. Racine

Les racines sont à l'origine de la plupart de mots arabes. La plupart des mots arabes ont des racines de trois lettres et rarement de quatre ou cinq lettres, et conservent toujours leur ordre. Ils sont joints à d'autres lettres (diacritiques et affixes) pour former des mots différents ayant des significations apparentées. Par exemple, la racine **كتب** (il a écrit) a la signification de base «écrire». Plusieurs mots sont dérivés à partir de cette racine, en la conjuguant sous plusieurs formes (présent, imparfait, futur simple, passe simple, impérative, etc.). Il y a aussi des formes supplémentaires telles que les noms verbaux. [Ed-Dariouache, 2015]

	La racine "كتب" (écrire)				
Verbes	**كَتَبَ**	Il a écrit		**يَكْتُبُ**	Il écrit
	كَتَبْنَا	Nous avons écrit		**يَكْتُبونَ**	Ils écrivent
	كَتَبَتْ	Elle a écrit		**تَكْتُبُ**	Tu écris
	تَكْتُبونَ	Vous écrivez		**نَكْتُبُ**	Nous écrivons
Nom	**كاتِبٌ**	Ecrivain		**كِتَابَةٌ**	Ecriture
	كِتابٌ	Livre		**مَكْتُوبٌ**	Ecrit
	مَكْتَبٌ	Bureau		**اكتتاب**	Enregistrement

Tableau 2.5 : Quelque dérivation du verbe "كتب"

2.3.2.2. Schème

Les schèmes sont les formes définies des mots dans lesquels les racines peuvent être insérées. Ensemble, les lettres racines placées à l'intérieur des schèmes sont des mots. Les schèmes ont également des significations similaires à celles utilisées pour les suffixes et les préfixes.

Quel que soit le mot, il est donc issu d'une racine et inséré dans un schème. Pour construire un mot à partir d'une racine, il suffit de modifier les lettres'ف', 'ع'et ' ل ' successivement par les lettres composant de la racine. [Ed-Dariouache, 2015]

La racine	Le schème	Résultat
كتب	**فِعالٌ**	**كِتابٌ**
غلب	**مَفْعُولٌ**	**مَغْلوبٌ**
لعب	**فاعِلٌ**	**لاعِبٌ**
روى	**فاعٍ**	**رَاوٍ**

Tableau 2.6 : Exemple de construction des mots à partir d'un schème [Ed-Dariouache, 2015]

2.3.2.3. Lemme

On appelle lemme, qui est une forme entièrement vocalisée, l'entrée lexicale d'un lexique ou d'un dictionnaire. Le lemme d'un mot, qu'il soit simple ou composé, représente sa forme canonique qui est fonction de la catégorie grammaticale de ce mot, dans le cas où c'est un nom il doit être au singulier **لاعب** (joueur) et dans celui d'un verbe il doit être à l'accompli de la troisième personne du singulier **فاز** (gagner), etc. Pour réduire le nombre d'entrées lexicales, on a des lemmes regroupant les mots ayant la même racine, le même schème original et le même sens. [Benzater, 2015]

2.3.2.4. Affixes

Ils peuvent être concaténés à une racine pour indiquer des caractéristiques grammaticales telles que le genre, le temps du verbe, le nombre et la personne. Il existe trois types d'affixes en fonction de leur position: les préfixes sont ceux attachés au début du mot, les suffixes sont attachés à la fin et les infixes se trouvent au milieu du mot. [Zeroul and Lakhouaja, 2017]

Les préfixes dépendent des mots auxquels ils s'attachent. En effet, la plupart des mots arabes commencent par le préfixe **ال التعريف**, Al altåryif, l'article de définition Al qui est utilisé en tant que terme déclaratif. Pour cela, il y a trois types des préfixes. Premièrement, les préfixes nominaux qui sont réservés pour les noms et les adjectifs. Deuxièmement, les préfixes verbaux qui sont réservés aux verbes. Et troisièmement, les préfixes généraux qui sont utilisés indépendamment de type des mots [Ed-Dariouache, 2015]

Il y a deux types de suffixes tels que les suffixes verbaux et les suffixes nominaux. Les premiers dépendent de la transitivité et de la personne conjuguée. Les suffixes nominaux indiquent la flexion casuelle du nom (nominatif, accusatif, et génitif), le genre (masculin et féminin), le nombre (singulier, duel et pluriel) [Ed-Dariouache, 2015].

Type	**Les préfixes**			
	Nom en français	**Signification**	**Nom arabe**	**Transcription**
Préfixes Nominaux	L'article de définition	Le	ال	Al(lamltaarif)
	Les prépositions	Avec	ب	B
		Pour	لـ	L
		Comme	ك	K
Préfixes Verbaux	La particule du futur	Sera	س	S
	Les particules du subjonctif	Pour	ل	L
Préfixes généraux	Les conjonctions de Coordinations	Et	ف	F
		Et	و	W
	L'article d'interrogation	Est-ce-que	أ	A

Tableau 2.7 : Un exemple des préfixes [Bourezg, 2017]

Type	Nombres	Suffixes		
		Signification	Nom en arabe	Transcription
Première personne	Singulier	Moi/mon	ني	Nyi
	Duel/pluriel	Nous/notre	نا	Na
Deuxième personne	Singulier	Toi/ton	ك	K
	Duel	Vous/votre	كما	Kma
	Pluriel	Vous/votre	كم	Km
		Vous/votre	كن	Kn
Troisième personne	Singulier	Lui/son	ه	H
	Duel	Eux/leur	هما	Hma
	Pluriel	Eux/leur	هم	Hm
		Eux/leur	هن	Hn

Tableau 2.8 : Un exemple des suffixes dévisés selon leur type [Bourezg, 2017]

2.3.2.5. Stem

Un stem est le résultat de la combinaison d'une racine avec des affixes flexionnels pour indiquer des caractéristiques grammaticales telles que le nombre, la personne, le temps, etc. **[**Zeroul and Lakhouaja, 2017**]**

Un Stem est la dérivation obtenue à partir d'une racine donnée selon un modèle. L'arabe classique a un grand nombre de stems qui ne sont pas tous utilisables, 2% seulement sont utilisables selon Rashwan [Rashwan et al., 2009]. Par exemple, le mot **مدارس** (écoles) est obtenu à partir de la racine **درس** (il a étudié) selon le modèle **مفاعل**. Les Stems produits ne sont pas tous utilisables. [AL Hajjar, 2010].

Racine	**Modèle**	**Stem**	**Utilisable**
<**كتب**, ktb, il a écrit >	<**فعل**,fal, faire >	<**كتب**, ktb, il a écrit >	Oui
<**درس** ,drs, il a étudié >	<**فاعل**, faal >	<**دارس**, drs, étudiant >	Oui
<**أكل**, akl, il a mangé >	<**مفعول**, mfawul >	<**مأكول**, makwul, mengeable >	Oui
<**لعب**,lab, il a joué >	<**أفعلاء**, afala' >	<**ألعباء**, alaba' >	Non

Tableau 2.9 : Un exemple de génération de stems [Bourezg, 2017]

2.3.2.6. Mots dérivés

Le lexique arabe comprend trois catégories de mots: verbes, noms, et particules, les mots des deux premières catégories sont dérivés à partir d'une racine. Les mots dérivés sont construits à partir d'un stem en y ajoutant des affixes. La plupart des mots arabes sont considérés comme des mots dérivés, puisqu'ils sont construits à partir de racines. [Sanan, 2008]

La conjugaison du verbe consiste à ajouter des affixes. Ces affixes dépendent de nombreux paramètres : le temps, nombre, genre, etc. Prenons l'exemple du mot **يلعبن** (elles jouent) qui se dérive de la racine **لعب** (il a joué) par l'ajout du préfixe **ي** et du suffixe **ن**. Dans ce cas, le temps est présent, le nombre est pluriel, le genre est féminin.

Dans le cas des noms, la dérivation est utilisée pour indiquer le genre, le nombre, etc. Par exemple, le féminin singulier nécessite d'ajouter le suffixe**ة**comme **مدرسة** (école). Mais le féminin pluriel nécessite l'ajout du suffixe **ات** comme **جامعات** (des universités). Par contre, il y a des mots qui ont des règles de composition plus complexes, comme le cas des pluriels

irréguliers, tels que le mot **ابواب** (portes) qui estle pluriel du mot **باب** porte [Sanan, 2008]. [Ed-Dariouache, 2015]

2.3.2.7. Mots outils

Les mots outils sont des entités qui servent à situer des faits ou des objets par rapport au temps ou au lieu. Ils jouent également un rôle clé dans la cohérence et l'enchainement d'un texte. Par exemple, nous avons des particules qui désignent un temps: **بعد** (après),**قبل** (avant), ou un lieu tel que **حيث** (où). Selon leur sémantique et leur fonction dans la phrase, ils peuvent jouer un rôle important dans l'interprétation d'une phrase en exprimant une introduction, explication, conséquence, etc. [Benhalima, 2017]. Nous en distinguons deux catégories :

• Les mots outils non déclinables ou invariables : leurs formes sont constantes et n'acceptent aucune déclinaison, par exemple: (**على** sur), (**منذ** depuis), etc.

• Les mots outils déclinables ou variables : ils suivent le système de déclinaison à trois cas selon leurs fonctions dans la phrase. Par exemple, le quantificateur **كل** (tout) peut accepter les trois voyelles casuelles filiales pour désigner le nominatif, accusatif ou génitif selon sa fonction dans la phrase. [Ed-Dariouache, 2015]

2.3.2.8. Mots isolés

Les mots isolés sont les mots qui n'ont pas de racines, comme les noms propres, les noms communs et les particules. Un nom propre désigne toute substance distincte de l'espèce à laquelle elle appartient, Il ne possède en conséquence aucune signification, ni aucune définition. Exemple : **باريس** Paris, etc. Quant un nom commun est toute substance non distincte de l'espèce à laquelle elle appartient, il est pourvu d'une signification et d'une définition [Benhalima, 2017].Exemple : **بلد** (pays), **حيوان** (animal), etc.

2.3.2.9. Diacritiques

Ce sont des signes qu'on ajoute au-dessus ou au-dessous des consonnes d'un mot pour spécifier sa prononciation ou pour en spécifier le sens. Il existe 3 de ces signes et ils sont transcrits de la manière suivante:

- La *fetha*(**الفتحة**)[a] est symbolisée par un petit trait sur la consonne (**تَ** ta/).
- La *damma*(**الضمة**)[u] est symbolisée par un crochet au-dessus de la consonne(**تُ** tu/).
- La *kasra*(**الكسرة**)[i] est symbolisée par un petit trait sous la consonne(**تِ** ti/).

Un petit rond symbolisant le *soukoun* (**سكون**) et apposé sur une consonne lorsque celle-ci n'est liée a aucune voyelle. [Baloul, 2003]

2.3.2.10. Šhadda

C'est le signe "ّ" (šadda) de la gémination en arabe, où il représente un doublement d'une consonne lors de sa prononciation, et ne peut être utilisé dans la 1ère lettre d'un mot. Exemple :**كلَّم** (Il a parlé à).

2.3.2.11. Tanwin

C'est des signes sont utilisés à la fin des mots indéterminés. Ils n'apparaissent jamais avec l'article de détermination AL (ال). 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*.

2.4. Prétraitements nécessaires

2.4.1. Encodage

Sur le Web, il existe plusieurs formats d'encodage de l'arabe (Unicode, ISO-8859-6 et CP1256) rendant ainsi les textes recherchés et les requêtes incomparables du fait de leurs différents encodages. Dans certains cas les documents sont représentés en Unicode (UTF-8), et les requêtes en ISO-8859-6. En utilisant des outils de conversion entre différents encodages et les tables de l'alphabet arabe on peut apparier les documents avec les requêtes. Dans notre cas tout a été transformé en format Unicode (UTF-8). [Kadri, 2008]

2.4.2. Tokenisation

L'identification des mots dans une séquence de lettres est appelée tokenisation. Par rapport aux textes anglais, pour la tokenisation des textes arabes, on retrouve, en plus de la ponctuation habituelle de l'anglais. Ainsi, on trouve d'autres signes de ponctuation arabe (encodésen arabe) tels que la virgule, le point virgule et le point d'interrogation qui sont considérés comme des séparateurs. Dans les langues européennes, ces signes ne séparent

pas obligatoirement des mots (ex. aujourd'hui), ce qui n'est pas le cas en arabe où ces signes sont obligatoirement des séparateurs [Lamrani et al., 2014]

2.4.3. Normalisation orthographique

Dans l'écriture arabe, bien que les voyelles soient souvent omises, sa lecture n'est pas pour autant difficile. Néanmoins, dans certains textes les mots sont représentés avec les voyelles. Donc, pour la normalisation, il est nécessaire d'éliminer ces voyelles. Dans l'écriture arabe certains mots subissent de légères modifications ne changeant pas considérablement le sens du mot, ce qui n'est pas le cas pour leur encodage. On peut citer à titre d'exemple la lettre *hamza* " أ" au début des mots, elle peut être représentéepar " أ" comme dans **أكل** (manger),"إ" dans **إبل** (chameaux) ou encore آ " "dans**آبار** (puits). La nécessité de ce prétraitement est aussi due au fait qu'on a souvent tendance à mal écrire ces différentes formes de *hamza*. Ce type d'erreurs est fréquent dans les textes arabes. Par exemple, le mot**أكل**est généralement écrit **اكل**. Une autre lettre causant un problème dans les textes arabe est la lettre " ة " qui, à la fin d'un mot, peut être écrite**ة**ou **ه** (**عادة** et **عاده**). A cause de toutes ces spécificités de la langue arabe et des problèmes de variation de représentationde ses caractères aussi bien dans les textes que dans les requêtes, voici quelques méthodes de normalisation sur le corpus avant l'indexation [Kadri, 2008] :

- Remplacer les *hamzas (*أ, إ , آ *) par alifbar 'ا'(A).*
- Remplacer' ى ' par 'ي'(Y) à la fin des mots.
- Remplacer la séquence 'ىء' par ' ي '
- Eliminer le caractère "*tatweel* ""*kashida*" (-) utilisé pour l'esthétique dans les textes arabes.
- Eliminer les diacritiques (voyelles) et la "*chedda*"

2.4.4. Construction de mots fonctionnels

Avant l'indexation des documents arabes, on élimine les mots fonctionnels(ou mots outils) qui n'ont pas un sens particulier utile pour la recherched'information, en utilisant des tables de mots outils qui ont été établies à cet effet. Parmi les tables conçues pour l'arabe, la plus reprise et répandue est celle de Khoja renfermant 168 entités [référence de Khoja] [Kadri, 2008]. En arabe, une multitude de formes peuvent être générées pour un même mot outil, rien qu'en y collant une préposition comme c'est le cas dans l'exemple suivant: **قبل** (avant), **قبله** (avant lui), **قبلها** (avant elle), **قبلهم** (avant eux), **قبلهن** (avant eux),**قبلهما** (avant

eux), قبلك (avant toi). Chen a pu établir une table d'environ 2942 entités, et ce en exploitant la collection des documents TREC, d'où il a extrait tous les mots uniques, les a traduit en anglais et a ensuite filtré tous ceux dont la traduction est un mot outil en anglais [référence de Chen]. On trouve également la liste de Abainia regroupant 600 mots [Abainia, 2016].

2.5. Lemmatisation

Pour l'arabe, le stemming ou lemmatisation est devenu l'un des principaux domaines de recherche de la RI. En fait, sur le plan morphologique, il a un rôle important dans le traitement automatique d'un langage complexe et joue un rôle essentiel dans la mise au point d'un bon système de récupération de documents. Les chercheurs ont conclu que la RI arabe peut être améliorée lorsque les racines ou les stems sont utilisées dans des opérations d'indexation et de recherche [Zeroul & Lakhouaja, 2017]]. Le stemming est aussi l'un des premiers traitements qui se répercute directement sur les performances de n'importe quel autre traitement morphologique ultérieur.

En ce qui concerne la classification des algorithmes de stemming en arabe, il existe deux approches principales concernant la nature de leurs règles appliquées. La lemmatisation légère ou light stemming est généralement basé sur un algorithme qui supprime les clitiques sans essayer de traiter les affixes ni de rechercher des racines. Le deuxième type est lemmatisation profonde ou root stemming utilisant un algorithme pour transformer les mots infléchis à leur racine. [AL-OMARI et ABUATA, 2014]

2.5.1. Difficultés de la lemmatisation des mots arabes

Dans le traitement automatique des langages naturels, l'arabe dont la morphologie très riche et variable rend la lemmatisation des mots difficile. Du fait que le but de la RI est de déterminer une forme appropriée d'index aux mots dans cette langue, il est impératif d'exécuter un traitement morphologique pour la recherche d'information. Le traitement de ce langage est aussi difficile à cause du nombre de complexités qu'il renferme.

- L'omission des voyelles des mots des articles de journaux sur lesquels le corpus de test est construit rend les choses plus ambigües ce qui peut mener à confusion avec d'autres mots ayant la même forme dans les requêtes ou dans les dictionnaires.
- La langue arabe admet des flexions grammaticales considérables, d'un mot, au point qu'un mot comme « فعل» qui peut avoir 30 dérivés entre nominaux et verbaux entraînant une multitude de formes pour un même mot. Réduire toutes ces formes à

une seule et unique forme serait très bénéfique à la recherche d'information. Cependant, la réalisation de cet objectif n'est pas aussi facile.

- En arabe, un mot peut engendrer d'autres formes, soit par attachement d'une préposition, d'une particule, ou d'un préfixe à son début, soit par attachement d'un pronom ou d'un suffixe à sa fin. L'insertion, la suppression ou la modification de lettres à l'intérieur d'un mot peut aussi produire certaines formes (comme le cas des verbes irréguliers).
- En arabe, même les formes les plus proches l'une de l'autre (singulier et pluriel) sont parfois irrégulières, parce qu'elles ne sont pas différenciées par de simples inflexions tel que l'ajout d'un préfixe ou d'un suffixe. Par exemple, قافله (caravane au singulier) devient قوافل. Il est extrêmement difficile d'écrire un algorithme à base de règles pour réduire ce genre de pluriel au singulier sans un lexique pour ces types de mots.
- Nous pouvons conclure, en résumé, que la morphologie de la langue arabe avec toutes ses caractéristiques et spécificités a rendu sa lemmatisation et le traitement de ses documents difficile. [Kadri, 2008]

2.5.2. Différents travaux sur la lemmatisation

Chaque langue se distingue par ses propres caractéristiques et dispositifs, ce qui ne permet pas l'utilisation de la configuration de lemmatisation d'une langue à une autre. Il est de même difficile d'appliquer la technique d'une langue à une autre, car l'une pourrait être pertinente à une langue, alors qu'elle ne l'est peut être pas pour d'autres langues. Parmi les techniques de lemmatisation de mots, on a les techniques à base de dictionnaires, d'analyse morphologique, de suppression des affixes, de statistiques et de traduction.

2.5.2.1. Lemmatisation des racines

Pour obtenir la racine d'un mot arabe donné avec cette approche, on utilise l'analyse morphologique. Cette dernière s'appelle lemmatisation profonde ou racinisation. De nombreux analyseurs morphologiques ont été développés et améliorés pour l'arabe, peu d'entre eux reçoivent une évaluation RI standard. La plupart de ces analyseurs morphologiques trouvent la racine, ou un nombre quelconque de racines possibles pour chaque mot.

- **Lemmatiseur de Khoja :**

Ce lemmatiseur agressif élimine le suffixe le plus long et le préfixe le plus long. Il fait ensuite correspondre le mot restant avec des schèmes verbaux et nominaux pour extraire la racine [Larkey and Connell, 2001]. Ce lemmatiseur bénéficie de plusieurs fichiers de données linguistiques tels qu'une liste de tous les caractères diacritiques, des caractères de ponctuation, des articles définis et 168 mots fonctionnels. L'algorithme de Khoja supprime initialement les suffixes, les infixes et les préfixes et utilise la correspondance de modèle pour extraire les racines, mais présente des problèmes en particulier avec les noms.

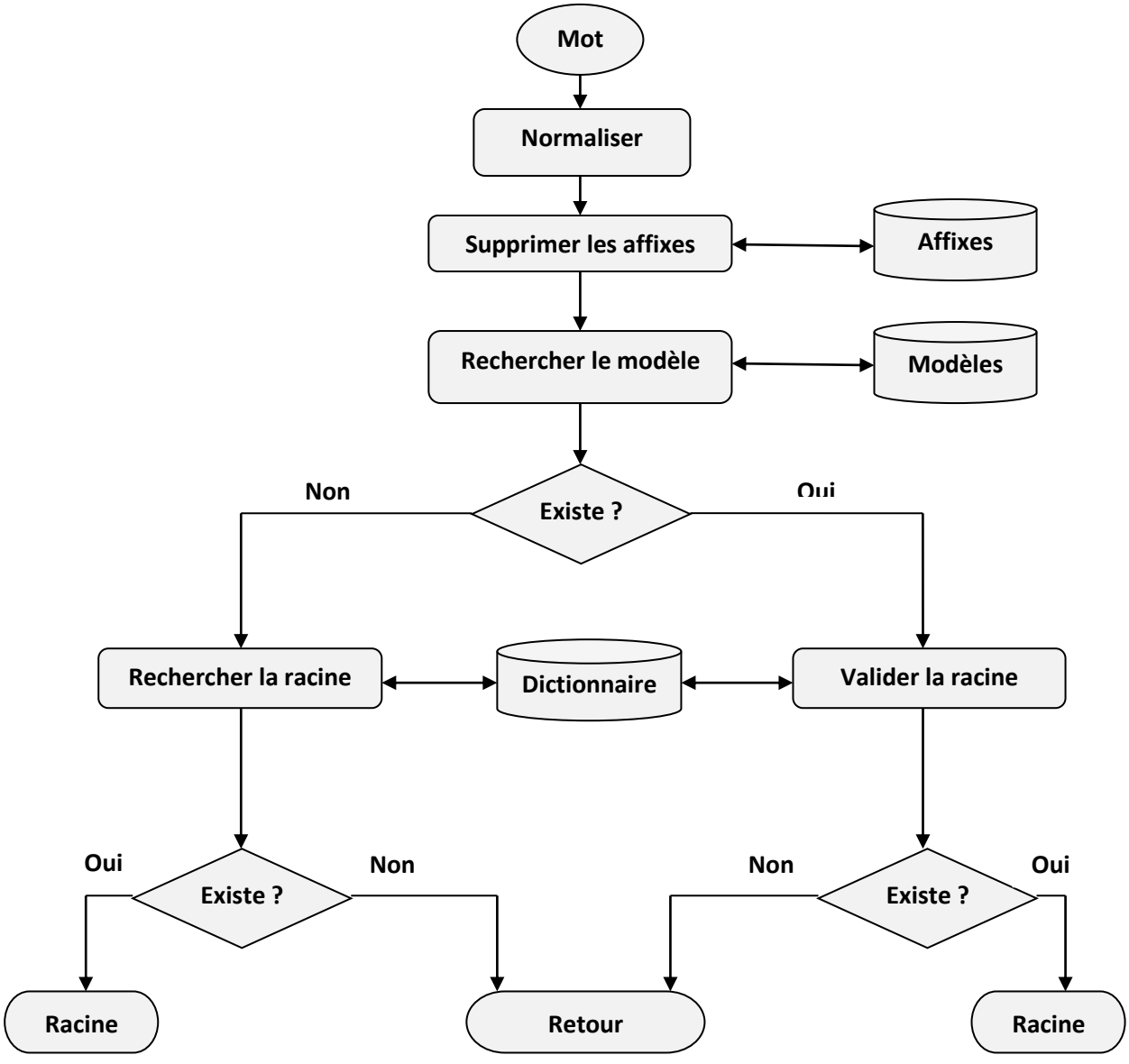

Figure 2.1: Organigramme lemmatiseur khoja [Benblal and Belouafi, 2015]

- **Lemmatiseur d'Al Kabi :**

Al-Kabi a indiqué que de nombreux algorithmes de stem en arabe ont été présentés au cours des quinze dernières années. Celui de Khoja est considéré par de nombreux chercheurs comme un standard pour la langue arabe. Par conséquent, son étude tente d'améliorer l'efficacité de cette méthode standard en adoptant de nouveaux modèles (schèmes et formes). L'ajout de ces modèles contribue à améliorer sa précision d'environ 5% [Al-Kabi, 2013].

- **Lemmatiseur d'Al-Shalabi:**

Al-Shalabi a développé un algorithme d'extraction de racines pour les racines trilittérales, qui ne fait appel à aucun dictionnaire [Al-Shalabi and Evens, 1998]. L'algorithme compte pour donner des poids aux lettres des mots, où pour chaque lettre le poids est multiplié par la position de celui-ci. Les lettres de consonnes étaient pondérées à zéro, où différentes des valeurs de pondération étaient attribuées aux lettres du mot s*a'altumuniha* - سألتمونيها, les affixes étant formés d'un mélange de ces lettres. Des calculs spécifiques sont ensuite effectués sur ces poids pour extraire la racine correcte. [Bahassine, 2014]

2.5.2.2. Lemmatisation des infixes

Pour la RI arabe on utilise une approche de lemmatisation qui lui est adaptée. Cette dernière est plus connue sous le nom *Light stemming*. C'est une approche assouplie et est inspirée par le processus de lemmatisation de l'anglais. Larkey , Darwish et Chen ont développé des lemmatiseurs de ce type basés sur l'application d'une légère troncature sur le début et la fin des mots [Larkey and Connell, 2001] [K. Darwish and D. W. Oard, 2002] [A. Chen and F. Gey, 2002]. Cette procédure est basée sur des listes de préfixes et de suffixes à une, deux et trois lettres qui ont été établies et dont le choix est généralement déterminé selon des statistiques de corpus. Sur la base de ces statistiques, on tronque un préfixe ou un suffixe d'un mot selon de simples règles comme par exemple sa longueur, en fonction des fréquences d'occurrence des préfixes et des suffixes sur les mots d'un grand corpus de textes. On ne peut, par exemple, tronquer un préfixe à trois lettres d'un mot à quatre lettres.

- **Lemmatiseurs de Larkey:**

Après la normalisation et la suppression des stop words (mots fonctionnels), On parcourt la liste des suffixes dans l'ordre (droite à gauche) indiqué dans le tableau ci-dessous

(Tableau 2.10.), et en supprimant tous que l'on retrouve à la fin du mot, si cela laisse trois caractères ou plus. Larkey [Benzater, 2015] a proposé plusieurs lemmatiseurs (i.e. Light1, Light2, Light8 et Light10). Pour ce faire, un nouvel ensemble de préfixes et suffixes est ajouté pour chaque lemmatiseur, or on trouve que Light10 est le plus performant en regroupant le plus grand ensemble de préfixes et suffixes.

	Supprimer les préfixes	**Supprimer les suffixes**
Light 1	ال, وال, بال, كال, فال	Aucun
Light 2	ال, وال, بال, كال, فال, و	Aucun
Light 3	’’	ه,ة
Light 8	’’	ها, ان, ات, ون, ين, يه, ية, ه, ة, ي
Light 10	ال, وال, بال, كال, فال, لل, و	ها, ان, ات, ون, ين, يه, ية, ه, ة, ي

Tableau 2.10 : Des préfix et suffixe

- **Lemmatiseur de Chen:**

Il a généré trois listes constituées par la première, les deux premiers, et les trois premiers caractères, et trois listes constituées du dernier, les deux derniers, et les trois derniers caractères respectivement, aux mots arabes. Ensuite, les six listes ont été triées par ordre décroissant du nombre de mots, dans lesquelles un préfixe ou un suffixe se produit.

- **Lemmatiseur d’Ababneh:**

L’approche proposée par Ababneh est basée sur un dictionnaire de mots exceptionnels (mots ne doivent pas être lemmatisés), contourne les limites de la lemmatisation légère qui produisent les erreurs de sous lemmatisation [Ababneh et al., 2012]. Cette technique recherche d’abord le mot parmi les mots exceptionnels (dictionnaire), et dans le cas où il est inexistant, l’algorithme cherche la ressemblance du mot à un des patterns (formes des mots) prédéfinis, sinon il en extrait le lemme par élimination des additions agglutinées à la forme tels que les préfixes et suffixes.

2.5.2.3. Lemmatisation hybride

Les méthodes hybrides de lemmatisation associent des techniques de lemmatisation et des techniques statistiques de correspondance, en particulier des approches basées sur le n-

gramme. Alhanini a proposé un algorithme d'arborescence arabe amélioré utilisant à la fois les méthodes de lemmatisation assoupli et la lemmatisation basée sur le dictionnaire, conçu pour surmonter les inconvénients des deux [Alhanini, 2011]. Concernant la lemmatisation de l'arabe et pour surmonter les faiblesses des algorithmes de lemmatisation existants, Hadni a proposé une méthode hybride intégrant trois techniques. Les trois techniques sont: Khoja Stemmer, Light Stemmer et N-Gram. Al-Nashashibi a proposé une méthode de lemmatisation hybride basée sur certaines règles [Al-Nashashibi, 2010]. La lemmatisation ne peut tronquer un préfixe que selon des règles spécifiques. Le tronqueur d'infixes est basé sur des modèles supprimant les infixes du mot en fonction de modèles spécifiques. Khedr a introduit un nouvel algorithme [Khedr, 2005] qui fournit une extension à un nouvel ensemble de règles et de modèles en combinant le lemmatiseur de Beltagy [Beltagy, 2009] et Light10.

Une autre méthode fondée au même temps sur des analyses morphologique et statistique a été introduite par De Roeck et Al-Fares [De Roeck and Al-Fares, 2000]. Dans cette méthode, on commence d'abord par l'élimination des affixes par application d'un lemmatiseur assoupli, ensuite on calcule les coefficients de ressemblance entre le stem obtenu et une liste de racines sélectionnées dans un dictionnaire à laquelle on ajoutera les racines correspondantes aux coefficients de ressemblance maximum.

2.5.3. Différentes méthodologies d'évaluation des lemmatiseurs

Les lemmatiseurs arabes sont utilisés dans de nombreuses tâches de traitement de texte telles que la catégorisation textuelle, l'extraction d'informations, la synthèse de texte et l'indexation par moteur de recherche. Par conséquent, l'évaluation et l'analyse comparative de ces lemmatiseur est cruciale pour que les chercheurs puissent choisir la meilleure formule répondant à leurs besoins dans un contexte donné. De plus, les auteurs ont généralement supposé de comparer leurs travaux avec d'autres afin de montrer leurs contributions et leurs améliorations.

2.5.3.1. Paramètres de Paice

Dans sa nouvelle méthodologie d'évaluation des lemmatiseurs, Paice [Paice, 1994] [Paice, 1996] a introduit trois nouveaux paramètres quantitatifs d'évaluation et qui sont en l'occurrence l'under stemming index (UI), l'over stemming index (OI) et le stemming Weight (SW). On parle d'under stemming error ou erreur de sous-lemmatisation lorsqu'un ensemble de groupes de mots reliés morphologiquement et sémantiquement est soumis à un

lemmatiseur, et que celui-ci produise deux ou plusieurs lemmes pour le même groupe. Par contre, lorsque le lemmatiseur produit le même lemme pour différents groupes, alors dans ce cas il produit une erreur de sur-lemmatisation (over stemming error). Cette méthodologie a été utilisée par AlShammari et par Abainia [Al-Shammari et al. 2008] [Abainia, 2016].

Généralement, les erreurs des lemmatiseurs légers produisent des erreurs de sous lemmatisation du fait qu'ils sont basés l'élimination des affixes [Al-Shammari and Lin, 2008a] [Al-Shammari and Lin, 2008b] [Larkey et al., 2002] [Paice, 1994]. Par contre, les erreurs produites par les racinisateurs, qui s'intéressent à la racine des mots sans se préoccuper de la sémantique et les diacritiques, produisent des erreurs de sur-lemmatisation.

Pour Paice, le meilleur lemmatiseur est celui qui produit des petites valeurs d'UI et OI qui sont les paramètres qu'il a introduit en plus du SW. Parmi les formules qu'il a introduites on a celle du Desired Merge Total (DMT) qui représente le nombre de toutes les paires de mots dans le groupe :

$$\text{DMT} = 0.5\, ng(ng-1) \quad \text{où } ng \text{ est le nombre de mots dans le groupe } g \quad (2.1)$$

Celle du Desired Non-merge Total (DNT) qui représente la confusion entre les mots d'un groupe avec d'autres d'un autre groupe (sémantiquement différents) :

$$\text{DNT} = 0.5\, ng(W-1) \quad \text{où } W \text{ est le nombre des mots dans l'échantillon} \quad (2.2)$$

L'addition des valeurs de DMT et DNT des groupes du même échantillon nous fournit le Global Desired Merge Total (GDMT) et Global Desired Non-merge Total (GDNT), respectivement.

$$\text{GDMT} = \sum_{g=1}^{n} DMTg \quad (2.3)$$

$$\text{GDNT} = \sum_{g=1}^{n} DNTg \quad (2.4)$$

Afin d'estimer l'erreur de sous-lemmatisation, Paice introduisit le paramètre UMT (Unachieved Merge Total).

$$\text{UMT} = 0.5 \sum_{i=1}^{s} ui(ng-ui) \quad (2.5)$$

Où ui est le nombre d'occurrences du $i^{ème}$lemme du groupe g

La somme de toutes les valeurs UMTs d'un échantillon nous donne le GUMT (Global Unachieved Merge Total). Le rapport entre le GUMT et le GDMT nous donne l'UI (Under-stemming Index) :

$$\text{GUMT} = \sum_{g=1}^{n} UMTg \quad (2.6)$$

$$\text{UI} = GUMT/GDMT \quad (2.7)$$

Pour l'erreur de sur-lemmatisation Paice introduisit le paramètre WMT (Wrongly Merged Total), en supposant qu'après la lemmatisation, chaque lemme ayant *N* occurrences dans l'échantillon et est confus dans *t* groupes, où chaque lemme confus a un nombre d'occurrences dans chacun des groupes, le WMT est calculé comme suit :

$$\text{WMT} = 0.5 \sum_{i=1}^{t} vi(ns - vi) \quad (2.8)$$

La valeur de l'over-stemming (OI) est obtenue par détermination du GWMT (Global Wrongly Merged Total) à l'aide de l'équation suivante :

$$\text{GWMT} = \sum_{g=1}^{n} WMTg \quad (2.9)$$

$$\text{OI} = GWMT/GDNT \quad (2.10)$$

Le dernier paramètre introduit par Paice est de stemming weight (SW) déterminé par la relation suivante (ratio entre OI et UI):

$$\text{SW} = \text{OI/UI} \quad (2.11)$$

On peut conclure que les mesures introduites par Paice ne sont assez significatives que lorsqu'on compare des lemmatiseurs de la même catégorie. Contrairement aux light stemmers qui produisent de faibles valeurs d'OI, les racinisateurs produisent de faible UI.

2.5.3.2. Evaluation sur la catégorisation des textes

La classification de texte est l'une des applications les plus importantes et les plus connues dans le domaine de l'exploration de données. De nombreuses recherches ont abordé le problème de la classification des textes dans de nombreuses langues, notamment l'anglais [Zaghloul et al., 2009], le chinois [He et al., 2000] et l'arabe [Al-Harbi et al., 2008; Al-Shargabi et al., 2011; Kanaan et al., 2009].

L'objectif de la classification des textes est d'atteindre une grande précision dans la classification des documents en fonction de catégories prédéfinies. Cette précision est

affectée par différents problèmes liés à la classification des textes. Ces problèmes incluent l'utilisation des algorithmes et des techniques de lemmatisation, les caractéristiques du jeu de données utilisé et les algorithmes utilisés pour la tâche de classification des textes.

- **Classifieur Bayesien**

Naïve Bayes est un classifieur probabiliste simple basé sur le théorème de Bayes. Lorsque ce classifieur est appliqué à la catégorisation, nous utilisons l'équation suivante:

$$P(\text{class}|\text{document}) = \frac{P(\text{class}).P(\text{document}|\text{class})}{P(\text{document})} \quad (2.12)$$

Où P (class| document) est la probabilité qu'un document D appartient à une classe *C*, et P(document) est la probabilité d'un document (constante peut être ignorée). P (class) est la probabilité d'une classe calculée à partir du nombre de documents dans une catégorie divisé par le nombre de documents dans toutes les catégories. P(document| class) est la probabilité de document dans une classe, et les documents peuvent être représentés par un ensemble de mots comme suit :

$$P(\text{document}|\text{class}) = \prod_i p(\text{wordiclass}) \quad (2.13)$$

$$P(\text{class}|\text{document}) = p(\text{class}).\prod_i p(\text{wordi}|\text{class}) \quad (2.14)$$

Où P(wordi | class) est la probabilité qu'un mot donné se produise dans tous les documents de la classe *C*, et est calculée par :

$$P(\text{wordi}|\text{class}) = \frac{(\text{Tct} + \lambda)}{(\text{Nc} + \text{V})} \quad (2.15)$$

Avec *Tct* est le nombre de fois que le mot apparaît dans cette catégorie *C*, *Nc* est le nombre de mots dans la catégorie *C*, *V* est la taille du tableau de vocabulaire, et λ est la constante positive, généralement 1 ou 0,5 pour éviter une probabilité nulle

- **Bahassine**

Il a présenté un nouvel algorithme d'extraction permettant de réduire les attributs à leur racine pour la classification des documents en arabe [Bahassine, 2014]. Le classificateur d'arbre de décision (DT ou Decision Trees) est utilisé pour créer le modèle, et Chi-square (X^2) est utilisé comme méthode de sélection des caractéristiques. Les résultats de

l'approche ont été comparés avec ceux de Khoja, où ils ont montré que le lemmatiseur proposé surpasse celui de Khoja.

La formule de statistique Chi-square est liée aux fonctions de sélection des caractéristiques informationnelles et théoriques, qui tentent de saisir l'intuition que les meilleurs termes de la classe *C* sont ceux qui se répartissent le plus différemment dans les ensembles d'exemples positifs et négatifs de *C*.

Pour calculer X^2 pour un terme *t* et une classe particulière *C*, le tableau de contingence (voir le tableau 2.11) d'un terme *t* et de la classe *C* peut être utilisé pour illustrer l'idée.

	C	Not c	Total
T	A	B	A+B
Not t	C	D	C+D
Total	A+C	B+D	N

Tableau 2.11 : la table de contingence de T et C

$$X^2(t,c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (2.16)$$

Où *N* est le nombre total des documents dans le corpus, *A* est le nombre de documents de la classe *c* contenant le terme *t*. *B* est le nombre de documents contenant le terme *t* dans d'autres classes. *C* est le nombre de documents de la classe *c* ne contenant pas le terme *t*, *D* est le nombre de documents qui ne contiennent pas le terme *t* dans d'autres classes.

Les résultats obtenus avec cet algorithme et celui de Khoja ont été comparés en termes de rappel. La mesure de rappel est définie comme suit:

$$\text{Rappel} = CC/TC \quad (2.17)$$

Où *CC* est le nombre de catégories correctes trouvées (vrai positif), et *TC* est le nombre total de catégories correctes (vrai positif + faux négatif).

2.5.3.3. Evaluation par correspondance des mots

- **Précision**

La précision des résultats renvoyés par un lemmatiseur exprime comment ces résultats sont corrects. Contrairement à la précision classique et aux scores de rappel, la précision est égale à 100% uniquement si le lemmatiseur renvoie tous les lemmes corrects pour tous les mots. En outre, il ne renvoie pas de lemmes incorrects supplémentaires.

Si la précision est égale à 100%, cela signifie que le lemmatiseur est performant. Dans le cas des lemmatiseurs, la précision peut être calculée comme suit:

$$\text{Accurcy} - \frac{\text{TP}}{\text{TP+FP+FN}} \quad (2.18)$$

Où *TP* est le nombre total de stems corrects, *FP* est le nombre total de ceux qui ne sont pas corrects et *FN* est le total des lemmes corrects non renvoyées par le lemmatiseur.

- **Nombre de mots par amalgame de classe**

Il s'agit du nombre moyen de mots correspondant au même radical [Namly et al., 2017]. Par exemple, si les mots مكتبة (bibliothèque) يكتبون (ils écrivent) et كتاب (livre) sont liés à la même racine كتب, alors le score de WCC (Words Per Conflation Class) est trois. La valeur du nombre de mots par classe de conflit reflète la force du lemmatiseur. Plus la valeur est élevée, plus le lemmatiseur devient fort. La métrique WCC est calculée comme suit:

$$\text{WCC} - \frac{\text{C}}{S} \quad (2.19)$$

Où *C* fait référence au nombre total de mots de corpus distincts avant lemmatisation, ou au nombre de types de mots. De la même manière, *S* désigne le nombre de lemmes distincts renvoyés par le lemmatiseur.

2.6. Conclusion

Dans ce chapitre, nous avons exposé la langue arabe et ses particularités, c'est une langue riche et sensible au contexte, ce qui présente de nombreux défis pour des domaines divers. Ensuite, nous avons parlé sur la structure morphologique d'un mot arabe, où nous avons mis l'accent sur les différentes catégories auxquelles il appartient. Nous avons ensuite présenté les prétraitements nécessaires sur le corpus de texte avant le traitement.

Enfin, nous avons présenté un survol des méthodes d'extraction de la racine (lemmatisation) à partir d'un mot arabe, avec les différents problèmes qui s'opposent à cette technique, et les différents travaux réalisés dans ce cadre avec quelques méthodologies d'évaluation de ces lemmatiseurs.

Chapitre 3 : Contribution

3.1. Introduction

Dans ce chapitre nous allons présenter la conception de notre algorithme de lemmatisation ARLStem v1.1 dont on a déjà la version séquentielle (ARLStem 1.0) et qui a été réalisé par Abainia en 2016.

Dans un premier temps, nous présentons la version séquentielle de notre lemmatiseur ARLSTem (Arabic Light Stemmer), qui est basée sur la suppression des préfixes, suffixes et infixes des mots arabes. Nous montrons ensuite les améliorations que nous avons faites sur cette version. Les résultats obtenus par notre nouvelle version sont évalués, puis comparés à ceux de la version précédente ainsi d'autres lemmatiseurs.

3.2. ARLStem

En 2016, Abainia a conçu un lemmatiseur basé sur la transformation des mots en leur lemme en supprimant les préfixes, suffixes et infixes des mots arabes. Ce système est appelé ARLSTem (Figure 3.1).

Cet algorithme s'exécute en 6 étapes dont les trois premières ont lieu au début du processus, et si à la 4ème n'effectue aucun changement sur le mot (échec), alors on passe à la 5ème étape. Si cette dernière échoue aussi (aucun changement n'est fait), ça permet de conclure que le mot d'entrée est un verbe (absence de préfixes et suffixes). Donc, cela entraine la lemmatisation du verbe qui est la 6ème étape de l'algorithme, puisque les articles définis ainsi que les conjonctions sont éliminés lors de la 2ème étape et que les noms féminins sont traités dans la 5ème.

Dans la grammaire arabe, les pronoms objets sont toujours à la fin des mots et leurs articles définis sont toujours au début, l'ordre d'exécution des étapes de l'algorithme doit être respecté. Donc, on ne peut exécuter la 4ème avant d'avoir au préalable exécuté la 2ème et la 3ème (suppression des préfixes et suffixes). Dans la littérature arabe, on ne peut non plus trouver les pronoms d'objets après la lettre du féminin ni de noms pluriels avec la lettre du féminin, d'où l'impossibilité d'exécution de la 5ème étape avant la 3ème et 4ème (la lettre « ة » est un bon indicateur des noms). Donc le suivi de l'ordre établi est impératif. Le système de lemmatisation est constitué des étapes décrites comme suivant:

Etape 1 : Normalisation des caractères

C’est une des étapes principales dans la lemmatisation, et est considérée nécessaire à cause des variations qui peuvent exister lors de l’écriture d’une même unité lexicale.

- remplacer les différentes formes de la lettre *Alif* (avec *Hamza*) « آ, إ, أ » avec la lettre *Alif* barre « ا » (sans *Hamza*);
- remplacer la dernière lettre *Alif MaqSura* « ى » avec la lettre *Yaa* « ي»;
- supprimer la lettre *Waaw* « و » du début du mot si le nombre restant des lettres est supérieur ou égal à trois lettres.

Etape 2 : Suppression des préfixes

La suppression des préfixes consiste a supprimer une liste de préfixes préparée à l'avance. En comparant les premières lettres des mots arabes avec la liste de préfixes (seulement 12 préfixes des 17 existants dans la langue arabe sont utilisés), on supprime les séquences correspondantes qui satisfont à d'autres critères possibles. Par exemple, la chaîne restante doit contenir au moins trois caractères,et ce pour ne pas confondre avec les lettres originales des mots.

Préfixes	Suffixes
بال, كال, وال, فال, وبال, فكال, فلل, ولل, ال, لل, فل, فب	ك, كي, كم, كما, كنّ, ه, ها, هم,هما, هنّ, نا

Tableau 3.1 : Liste des préfixes et suffixes utilisés par l’algorithme ARLSTem

Etape 3 : Suppression des suffixes

Comme pour la suppression des préfixes, celle des suffixes le nombre de lettres restantes doit aussi égal au minimum à trois. Dans l’algorithme ARLSTem,11 lettres sur les 12 existants sont utilisées.

Un exemple de suppression de suffixes, comme le cas de «هم», le mot «عملهم» devient «عمل» (l’équivalent, plus ou moins à « leur » en français). Par contre, on ne peut éliminer ce suffixe dans le mot « ساهم », car ce sont des lettres originales et il ne nous resterait que deux lettres, ce qui ne satisfait pas la condition de la 3ème étape.

Etape 4 : Transformation du pluriel au singulier

La transformation du pluriel au singulier se fait en suivant quelques règles que nous allons les citer prochainement et qui traitent les suffixes, préfixes et infixes des mots.

- supprimer les deux lettres « ان » à la fin du mot, si le nombre de lettres restantes est 3 lettres au minimum. Exemple : le mot arabe « كتابان » devient « كتاب »
- supprimer les deux lettres « ين » à la fin du mot, si le nombre de lettres restantes est 3 lettres au minimum. Exemple : le mot arabe « مفتاحين » devient « مفتاح »
- supprimer les deux lettres « ون » à la fin du mot, si le nombre de lettres restantes est 3 lettres au minimum Exemple : le mot arabe « عاملون » devient « عامل »
- supprimer les trois lettres « تان » à la fin du mot, si le nombre de lettres restantes est 3 lettres au minimum Exemple : le mot arabe « متسابقتان » devient «متسابق»
- supprimer les trois lettres « تين » à la fin du mot, si le nombre de lettres restantes est 3 lettres au minimum Exemple : le mot arabe « اميرتين » devient «امير »
- supprimer les deux lettres « ات » à la fin du mot Exemple : le mot arabe « معلمات » devient « معلم »
- supprimer la lettre « ا » a 3ème position de début, si le nombre de lettres restantes 2 lettres au minimum + la même lettre existe dans la 1ère position Exemple : le mot arabe « آفاق » devient « أفق »
- supprimer les deux lettres « ا », Une dans la 1ère position et une autre dans l'avant la dernière position, si le nombre de lettres restantes est 3 lettres au minimum Exemple : le mot arabe « اعناق » devient « عنق »

Etape 5 : Transformation du féminin au masculin

En arabe, le féminin singulier nécessite d'ajouter le suffixe « ة », cette lettre peut être supprimée de la fin d'un mot, uniquement dans le cas où le reste du mot comporte au moins trois lettres [Adabneh et al., 2012].

Etape 6 : Lemmatisation des verbes

Le mot est probablement un verbe si aucune des étapes 2, 4 et 5 ne s'achève pas, et cela veut dire qu'il n'existe aucun des préfixes et suffixes attachés aux noms. Alors, on doit vérifier les préfixes des verbes, suffixes ou bien préfixes et suffixes ensembles pour avoir la lemme des verbes. Il existe trois cas :

- suppression des préfixes des verbes (tableau 3.2). exemple : « ساعمل » après la suppression des préfixes « سا » il devient « عمل »
- suppression des suffixes des verbes (tableau 3.2). exemple : «عملوا » après la suppression des suffixes «وا» il devient « عمل »
- suppression des co-occurrent préfixes et suffixes (tableau 3.3).exemple : « ستعملون » après la suppression des préfixes «ست» et des suffixes «ون» il devient « عمل »

Préfixes	Suffixes
ا ن ت ي س ا سي ست سن	تما تنّ نا تم تا وا ت ا ن

Tableau 3.2 : Liste des préfixes et suffixes des verbes

Préfixes	ست سد سي سد ست ست ت ي ا ا ا ي ي ت ت ت ي ي
Suffixes	ن ن ون ان ون ان ين ن ن ن ا ي وا ون ان ون ان ين

Tableau 3.3: Liste des co-occurrent préfixes et suffixes utilisés par l'algorithme ARLSTem

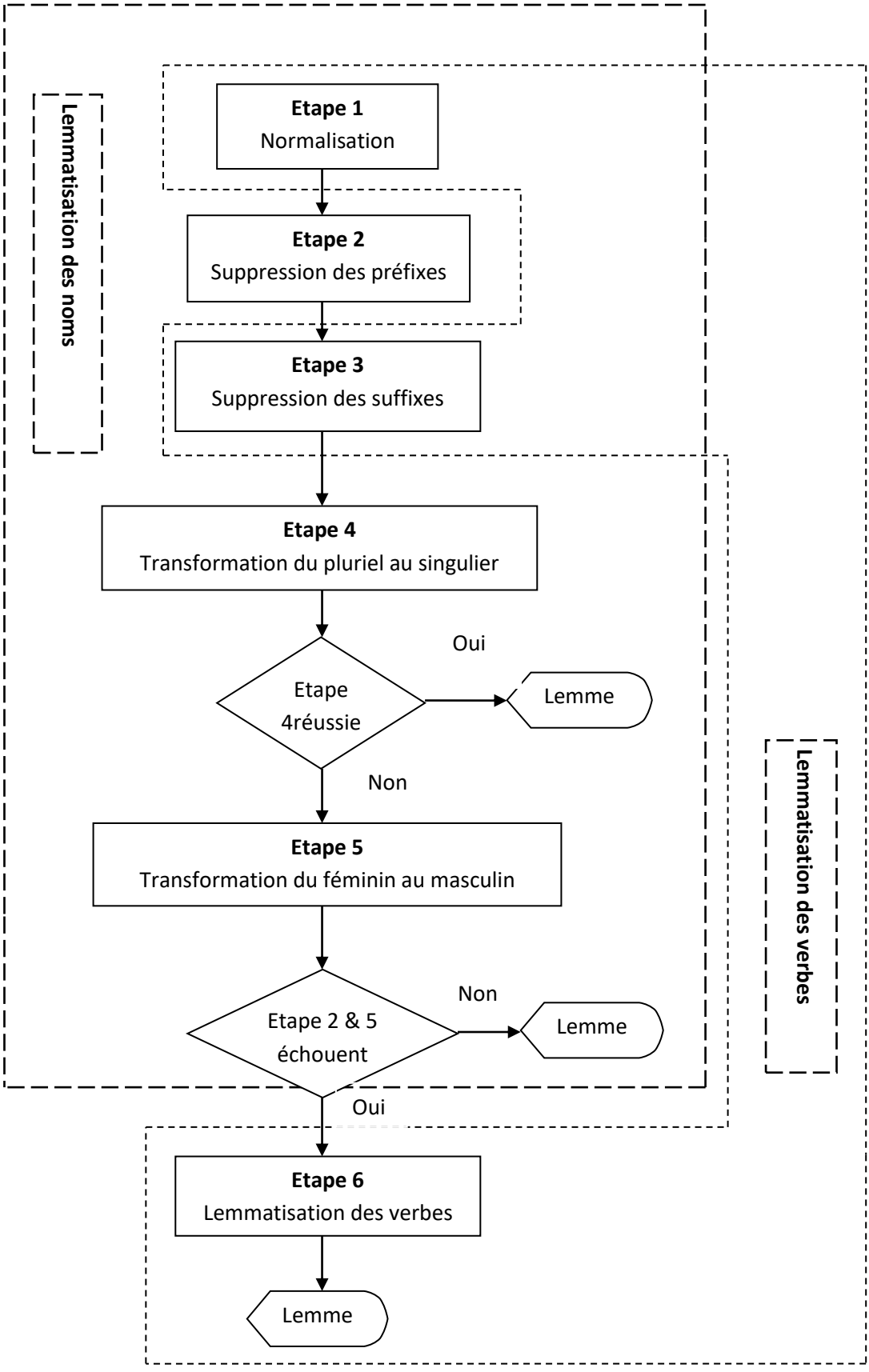

Figure 3.1: Organigramme de déroulement de l'algorithme ARLTSem.

3.3. ARLStem amélioré

Pour améliorer la précision de lemmatisation de l'algorithme « ARLStem » des modifications ont été apportées à son programme par addition de certaines règles. Ces règles ont été déduites de remarques faites sur la langue arabe et sont comme suit :

- Dans la troisième étape, on a ajouté deux instructions :

- si un mot qui se termine par les deux lettres « **ية** », on supprime ces deux lettres dans le cas où le mot comporte au minimum cinq lettres. Par exemple, le mot « **راسمالية** » devient « **راسمال** »
- si un mot qui se termine par les trois lettres «**اية** », on supprime les deux lettres « **ية**» dans le cas où le mot comporte au minimum six lettres.

- Dans la quatrième étape, une seule instruction a été introduite :

- si un mot débute par « **م** » et se termine par les deux lettres «**ون** », on supprime les trois lettres dans le cas où le mot comporte au minimum six lettres. Par exemple, le mot « **مفكرون** » devient « **فكر** »

- Dans la cinquième étape, on a introduit trois instructions :

- si un mot débute par « **ت** » et dont la quatrième lettre est « **ي** » et se termine par « **ية** », ces lettres peuvent être supprimées dans le cas où le mot comporte au minimum sept lettres. Par exemple, le mot « **تنظيمية** » devient « **نظم** »
- si un mot débute par « **ا** » et dont la quatrième lettre est « **ا** » et se termine par « **ية** », le deuxième « **ا** » et le « **ية** » peuvent être supprimées dans le cas où le mot comporte au minimum sept lettres. Par exemple, le mot « **اقتصادية** » devient « **اقتصد** »
- si la deuxième lettre d'un mot est « **ا** » et se termine par « **ة** » ces lettres peuvent être supprimées dans le cas où le mot comporte au minimum cinq lettres. Par exemple, le mot « **لاحقة** » devient « **لحق** »

Nos modifications aussi incluent l'ajout d'autres étapes à l'algorithme initial qui n'en avait que six. La première consiste à transformer l'adjectif en racine. A titre d'exemple, dans un mot qui débute la lettre par « **ا** » et dont la deuxième lettre avant la dernière est aussi « **ا** » et qui se termine par « **ي** », le deuxième « **ا** » et le « **ي** » peuvent être supprimées dans le cas où le mot comporte au minimum six lettres, par exemple le mot « **اشتراكي** » devient « **اشترك** ».

La deuxième fonction introduite, consiste à appliquer au résultat obtenu par toutes ces étapes d'autres instructions et qui sont :

- si un mot débute par « **ت** » et l'avant dernière lettre est «**ي** », on supprime les deux lettres dans le cas où le mot comporte au minimum cinq lettres. Par exemple, le mot « **تمويل** » devient « **مول** »
- si un mot débute par « **م** » et l'avant dernière lettre est «**و** », on supprime les deux lettres dans le cas où le mot comporte au minimum cinq lettres. Par exemple, le mot « **مشروع** » devient « **شرع** »
- si un mot se termine par « **ي** », on supprime cette lettre dans le cas ou le mot comporte au minimum quatre lettres. Par exemple, le mot « **شرعي**» devient « **شرع** »
- si un mot débute par « **ل** »», on supprime cette lettre dans le cas ou le mot comporte au minimum quatre lettres.

3.4. Evaluation et comparaison

Les algorithmes de lemmatisation arabe sont généralement évalués en fonction de la précision du lemmatiseur dans l'extraction d'information, qui est déterminée par le calcul du taux de réussite dans la lemmatisation d'un ensemble de mots. Cependant aucun corpus d'évaluation des lemmatiseurs n'est disponible publiquement.

3.4.1. La base de données ARASTEM

La base de données ou corpus a été créé et nommé ARASTEM (ARAbic STEMming of noisy texts). Il exploite des textes collectés de différents forums de discussion tels que : *efpedia.com, isegs.com, forum.elwlid.com, islamdi.com, hams-3.com, forum.stop55.com, quran.maktoob.com, djelfa.info, vb.alhilal.com, tunisia-sat.com, forum.educ40.net, alsayra.com, hassan-peche.com, syrianhunter.ahlamontada.net et alrahalat.com*. Il contient 60 textes de différentes tailles encodés en UTF-8 [**Abainia, 2016**].

Concernant la construction du corpus ARASTEM, tous les textes sont passés par une étape de prétraitement qui consiste à :

- supprimer les caractères insignifiants.
- supprimer les caractères français et anglais.
- supprimer les diacritiques arabes.

اكتشف فريق من الباحثين برئاسة راسل فوستر البروفيسور في مختبر نافيلد لطب العيون بجامعة أكسفورد، خلايا في شبكية العين لم تعرف سابقا، يطلق عليها اسم cells ganglion retinal (خلايا الشبكية العقدية) تستجيب للضوء الأزرق الذي يزيح عادة الساعة البيولوجية للانسان نحو مناطق زمنية جديدة

Figure 3.2: Exemple d'une partie d'un texte collecté pour la construction du corpus ARASTEM [Abainia, 2016]

La figure ci-dessus (Figure 3.3) montre que dans les textes arabes provenant des forums de discussion, il peut y avoir des citations dans des langues latines dues à l'utilisation, dans le monde arabe, de l'anglais ou du français comme langue secondaire employée pour citer quelques circonstances.

Après le prétraitement, les mots fonctionnels sont supprimés de tous les textes pour ne pas être pris en considération dans la lemmatisation. Cette suppression est suivie par l'élimination des répétitions des mots dans chacun des textes. Une liste de mots de chaque texte du corpus est extraite, et contient les mots uniques.

Enfin, dans l'étape finale du corpus, on regroupe les mots reliés sémantiquement et morphologiquement dans différents fichiers (Figure 3.4), où chaque ligne de chaque fichier contient un groupe de mots ayant la même racine. Cette étape étant exigeante en temps et en concentration, elle nécessitait l'aide de quelques locuteurs arabes natifs.

- ازمته الازمة فالازمة للازمة
- اسقطت السقطة بسقوط
- اسوا سيئ
- اشهر
- اعتبار باعتبار
- العالم العالمي العالمية
- افضل الافضل
- اقتصاد اقتصادي اقتصاديا الاقتصاديات الاقتصادية الاقتصاد والاقتصادي

Figure 3.3: Exemple de quelques groupes de mots d'un fichier d'ARASTEM.

3.4.2. Lemmatiseur de comparaison

Nous allons comparer les résultats obtenus avec ceux de la version précédente, et ceux d'autres méthodes. Les lemmatiseurs qu'on va les utilisés sont :

- **Lemmatiseur de Soori**

Les règles ont été divisées en 5 catégories [Soori et al., 2013]. Les deux premières catégories sont concentrées sur n'importe quel type de mot. La première étape nécessaire consiste à normaliser les mots en convertissant tous les types *d'alif* (أ) et (إ) en *alif bar* (ا), puis on normalise *alif maqSura* (ى) en *ya* (ي).

Après la normalisation, on supprime tous les préfixes possibles (préposition, articles définis et conjonctions) en utilisant le processus suivant:

- Supprimer *baa* (ب) du début de chaque mot;
- Supprimer *waaw* (و) du début de chaque mot, si le reste est constitué de 3 lettres ou plus.

On supprime ensuite les lettres liées au début d'un mot, si le nombre de lettres restantes est supérieur à trois lettres.

- Supprimer: *alif* (ا) et *laam* (ل);
- Supprimer: *alif* (ا), *waaw* (و) et *laam* (ل);
- Supprimer: *alif* (ا), *baa* (ب) et *laam* (ل);
- Supprimer: *alif* (ا), *kaaf* (ك) et *laam* (ل);
- Supprimer: *alif* (ا), *faa* (ف) et *laam* (ل).

La règle suivante est conçue pour les suffixes. La règle consiste à supprimer les lettres liées s'il reste deux lettres ou plus:

- Supprimer *ha* (ه) et *alif* (ا);
- Supprimer *alif* (ا) et *noon* (ن);
- Supprimer *alif* (ا) et *taa* (ت);
- Supprimer *waaw* (و) et *noon* (ن);
- Supprimer *ya* (ي) et *noon* (ن);
- Supprimer *ya* (ي) et *ha* (ه);
- Supprimer *ya* (ي) et *Ta'MarbuTa* (ة)

La troisième règle est conçue pour les adjectifs et les adverbes, et plus précisément pour changer les adjectifs et les adverbes de forme féminine au masculin :

- Enlever *ta 'marbuTa* (ة) s'il se trouve comme dernière lettre d'un mot.

- Supprimer *ha* (ه) s'il se trouve comme dernière lettre d'un mot.

La quatrième règle est conçue pour les noms; son objectif est de changer les noms du pluriel en forme singulière. Dans cette règle, les lettres liées sont supprimées si le reste est supérieur à 2 lettres. Si le mot ne comporte que trois lettres, on supprime les lettres suivantes si les conditions supplémentaires sont remplies.

- Supprimer *alif* (ا) du début de chaque mot;
- Supprimer *alif* (ا) s'il se trouve comme la lettre précédant la dernière;
- Supprimer *waaw* (و) s'il se trouve comme la lettre précédant la dernière;
- Supprimer *alif* (ا) du début de chaque mot, uniquement si la lettre précédant la dernière est également *alif* (ا).

Si le mot comporte quatre lettres, on supprime les lettres suivantes si les conditions supplémentaires sont remplies.

- Supprimer *alif* (ا) la lettre précédant le dernier
- Enlever *alif* (ا) s'il est trouvé comme troisième lettre

La cinquième règle est conçue pour les verbes. Elle convertit les formes flexionnelles du verbe en radical. Les lettres sont supprimées à la fin du mot, s'il en reste au moins 2.

- Supprimer *taa* (ت)
- Supprimer *yaa* (ي)
- Supprimer *alif* (ا)
- Supprimer *alif* (ا) si trouvée comme deuxième lettre d'un mot, uniquement si *alif* (ا)est trouvée en tant que dernière lettre d'un mot.
- Supprimer *waaw* (و)
- Supprimer *miim* (م)
- Supprimer *noon* (ن) et *alif* (ا)
- Supprimer *taa* (ت) et *alif* (ا)
- Supprimer *taa* (ت) et *miim* (م)
- Supprimer *taa* (ت) et *ya* (ي)
- Supprimer *miim* (م) et *alif* (ا)
- Supprimer *miim* (م), *taa* (ت) et *alif* (ا)
- Supprimer *noon* (ن), *taa* (ت) et *alif* (ا)

- Enlever *alif* (ا) s'il s'agit de la seconde lettre d'un mot. Après cela, supprimer *taa* (ت) à la fin du mot
- Supprimer *taa* (ت) et *alif* (ا)

- **Lemmatiseur ISRI**

L'ISRI (The Information Science Research Institute) qui est aussi un lemmatiseur assoupli arabe, n'utilise pas le dictionnaire des racines mais un ensemble des schèmes (voir Tableau 3.5). Pour chaque mot, son algorithme général, basé sur un ensemble de marques diacritiques et un ensemble d'antéfixes à enlever [Kaz et al., 03] , comporte les étapes suivantes:

- Suppression des diacritiques
- Normalisation des lettres (ء , ئ , ؤ) par la lettre (أ).
- Suppression des préfixes de longueur 3 et de longueur 2 dans cet ordre.
- Suppression du connecteur initial و , dans le cas où les deux lettres initiales du mot W sont وو .
- Normalisation, si besoin, des lettres initial (إ , أ , آ) par la lettre ا.
- Retournement de la racine si la longueur du mot est inferieure ou égale à trois lettres.

Dans les cas où on n'a aucun résultat, et en fonction des cas suivants, on essayera de déduire la racine du mot par :

a- Extraction de la racine appropriée si la longueur du mot trouvé est égale à 4 et que sa forme correspondant aux schèmes de forme PR4 (voir tableau 3.4), sinon par suppression des suffixes et des préfixes de longueur 1 de S1 et P1 dans cet ordre et retournement de la racine et ce uniquement si le mot a au moins trois lettres.

b- Extractionde la racine trilitère du mot s'il a 5 lettres et que sa forme correspondant aux schèmes de forme PR5. Dans le cas où aucune forme n'est adaptée, on retire les suffixes et les préfixes et on retourne la racine trilittère. Si la longueur du mot est encore de cinq caractères, le comparer avec les schèmes de PR54 et retourner la racine dans le cas où elle a une longueur de 4 lettres.

c- Extraction de la racine trilitère du mot s'il a 6 lettres et que sa forme correspondant aux schèmes de forme PR63, sinon suppression des suffixes. Si la suppression d'un suffixe résulte en un terme avec cinq caractères, alors renvoyer ce terme à l'étape b. Dans le cas où ce n'est toujours pas satisfaisant, on supprime alors les préfixes de

longueur 1, et en cas de succès, on renvoie le terme à l'étape b. Si le mot a toujours six caractères, on le comparera avec les schèmes de PR64 et on retournera sa racine dans le cas où elle a une longueur de 4 lettres.

d- Si le mot est de 7 lettres, suppression des suffixes de longueur 1 et envoie du terme résultant à l'étape précédente. Dans le cas où cela ne marchera pas, alors on supprimera les préfixes de longueur 1 et on renverra le terme obtenu à l'étape précédente (c).

Le type de l'ensemble		Description	Leur contenu proposé
Diacritiques		Les diacritiques de vocalisation	ّ, َ, ُ, ْ, ٌ, ـ, ٰ
Préfixes	P3	Les préfixes de longueur 3	وللل والل كال بال
	P2	Les préfixes de longueur 2	لل ال
	P1	Les préfixes de longueur 1	ا, ن, ت, ي, ف, و, س, ب, ل
Suffixes	S1	Les suffixes de longueur 3	كمل, تين, تان, همل, تمل
	S2	Les suffixes de longueur 2	نا, هن, كم, تن, ين, ان, ات, ون, هم, ما, وا, ني, تن, تم, ها, يا
	S3	Les suffixes de longueur 1	ن, ا, ت, ك, ي, ه, ة

Tableau 3.4: Les ensembles des antéfixes proposées par ISRI

Le type de l'ensemble	Description	Leur contenu proposé
PR4	Les schèmes de longueur 4	فاعل فعول فعلة فعال فعيل مفعل
PR53	Les schèmes de longueur 5 et racine de longueur 3	تفاعل افتعل افعال افاعل فعالة فعلان فعولة تفعلة تفعيل مفعلة مفعول فاعول فواعل مفعال مفعيلل افعلة فعائل منفعل مفتعل فاعلة مفاعل فعلاع يفتعل تفتعل فعالي انفعل
PR54	Les schèmes de longueur 5 et racine de longueur 4	تفعلل افعلل مفعلل فعللة فعلان فعالل
PR63	Les schèmes de longueur 6 et racine de longueur 3	استعفل مفعالة افتعال افعوعل انفعال مستفعل
PR64	Les schèmes de longueur 6 et racine de longueur 4	افتعلل افعلال متفعلل

Tableau 3.5 :les schèmes et leurs racines proposé par ISRI

- **Lemmatiseurs de Larkey**

Les Light-stemmers de Larkey étaient basés sur la suppression d'un petit nombre de préfixes et de suffixes sans toucher les infixes et ce afin de reconnaitre la forme et trouver le stem [Lar et al., 06]. En fonction des listes des préfixes et des suffixes à supprimer (voir Tableau 3.6), Larkey et ses collaborateurs ont développé plusieurs versions de light-stemmers, sauf qu'avant toute suppression et pour faciliter le traitement des mots, on doit normaliser les corpus et les requêtes.

	Suppression des préfixes	Suppression des suffixes
Light 1	ال ,وال,بال ,كال , فال	Aucun suffixe à supprimer
Light 2	ال ,وال,بال ,كال , فال, و	Aucun suffixe à supprimer
Light 3	ال ,وال,بال ,كال , فال, و	ه, ة
Light 8	ال ,وال,بال ,كال , فال, و	ه, ة, ها, ان, ات, ون, ين, يه, ية, ي
Light 10	ال ,وال,بال ,كال , فال, ولل	ه, ة, ها, ان, ات, ون, ين, يه, ية, ي

Tableau 3.6 : Les chaines enlevées par light stemming en arabe

Cette approche est facile, rapide avec minimisation d'espace mémoire. Pour notre implémentation de test, nous avons retenu la version Light-10.

- Normaliser le mot en entrée comme suit :
  - Supprimer les ponctuations
  - Supprimer les diacritiques courtes
  - Supprimer les caractères qui ne sont pas des lettres arabes
  - Remplacer les lettres (إ , أ , آ) par la lettre (ا)
  - Remplacer la dernière lettre ى par la lettre ي.
  - Remplacer la dernière lettre ة par la lettre ي.
- Supprimer la lettre *waw* si la longueur de mot dépasse trois lettres
- Supprimer l'article de définition 'ال' si la longueur de mot dépasse deux lettres
- Supprimer les suffixes indiqués dans le tableau 1 si la longueur de mot dépasse deux lettres
- Supprimer les préfixes montrés dans le tableau 1 si la longueur de mot dépasse deux lettres

- **Lemmatiseur d'Assem**

Les règles de ce lemmatiseur ont été divisées en plusieurs catégories. Premièrement, la phase de normalisation, elle consiste a normalisée *Hamza* :

- Normaliser *hamza* à la fin du mot
  - Remplacer les lettres (إ , أ , آ) par la lettre (ء)
  - Remplacer les lettres (ئ , ؤ) par la lettre (ء)
- Normaliser les autres *hamza*

- Remplacer les lettres (إ , أ , آ) par la lettre (ا)
- Remplacer les lettres (ؤ) par la lettre (و)
- Remplacer les lettres (ئ) par la lettre (ي)

La deuxième phase consiste à classifier le mot si c'est un verbe ou un nom.

- S'il commence par (بال), (كال) et le mot contient plus que 4 lettres, il est un nom défini
- S'il commence par (لل), (ال) et le mot contient plus que 3 lettres, il est un nom défini
- S'il se termine par (ة) et le mot contient plus que 2 lettres, il est un nom
- S'il se termine par (ات) et le mot contient plus que 3 lettres, il est un nom
- S'il se termine par (تان), (تين) et le mot contient 5 lettres ou plus, il est un nom

La phase suivante consiste à traiter les préfixes et elle est divisée en plusieurs étapes:

- Etapes 1
  - Remplacer les lettres (أأ) par la lettre (أ)
  - Remplacer les lettres (أئ) par la lettre (ئ)
  - Remplacer les lettres (أؤ) par la lettre (ؤ)
  - Remplacer les lettres (أإ) par la lettre (إ)
- Etape 2
  - Supprimer (فال), (وال), si le mot contient plus de 5 lettres
  - Supprimer (ف), (و), si le mot contient plus de 3 lettres
- Etape 3 : si le mot est un nom défini
  - Supprimer (بال), (كال), si le mot contient plus de 5 lettres
  - Supprimer (لل), (ال), si le mot contient plus de 4 lettres
- Etapes 4 : si le mot est probablement un nom et défini
  - Supprimer (ب), si le mot contient plus de 3 lettres
  - Supprimer (ك), (ل), si le mot contient plus de 4 lettres
  - Remplacer (ببب), (ككك) par (ب), (ك), si le mot contient plus de 3 lettres
- Etape 5 : si le mot est probablement un nom et non défini
  - Supprimer (ب), si le mot contient plus de 4 lettres
- Etape 6 : si le mot est un verbe
  - Supprimer (س), si le mot contient plus de 4 lettres

- Remplacer les lettres (سي) par la lettre (ي), si le mot contient plus de 4 lettres
- Remplacer les lettres (ست) par la lettre (ت), si le mot contient plus de 4 lettres
- Remplacer les lettres (سن) par la lettre (ن), si le mot contient plus de 4 lettres
- Remplacer les lettres (سا) par la lettre (ا), si le mot contient plus de 4 lettres
- Remplacer les lettres (يست), (نست), (تست) par les lettres (است), si le mot contient plus de 4 lettres

La dernière phase traite les suffixes:

- Pour les noms
  - Supprimer (ي), (ك), (ه) si le mot contient 4 lettres ouplus
  - Supprimer (نا), (كم), (ها), (هن), (هم), (ني), (كن), (تان), (تين), si le mot contient 5 lettres ouplus
  - Supprimer (كما), (هما), (كمو), si le mot contient 6 lettres ouplus
  - Supprimer (ن), (ات), si le mot contient plus de 5 lettres
  - Supprimer (ا), (ي), (و), (ت) si le mot contient plus de 4 lettres
  - Supprimer (ة), si le mot contient plus de 3 lettres
- Pour les verbes
  - Supprimer (ت), (ا), (ن), (ي), si le mot contient plus de 4 lettres
  - Supprimer (نا), (تا), (تن), (ان), (ون), (ين), (وا), (تم) si le mot contient plus de 5lettres
  - Supprimer (تما),(تمو) si le mot contient plus de 6 lettres
- Pour le suffixe « ئ »
  - Remplacer (ئ) par (ي)
  - Supprimer (ئ) si le mot est un nom qui contient plus de 3 lettres
  - Remplacer (ئ) par (ا) si le mot est un verbe

3.4.3. Expérimentation et résultats

Afin d'évaluer les performances de notre version d'ARLStem, on mesure les erreurs de lemmatisation en utilisant la méthode de Paice. Pour réaliser cette évaluation, nous avons utilisé le corpus ARASTEM comme entrée d'analyse pour les méthodes suivantes :

ARLStem v1, ISRI, Light10, Soori et Assem. Cette évaluation se fait sur la base de la comparaison des valeurs des erreurs de sur-lemmatisation (OI) et de sous- lemmatisation (UI) obtenues par ces méthodes et par la nôtre.

Le lemmatiseur idéal doit être capable d'associer (groupe) les mots liés au même stem à des valeurs de UI et OI faibles. À ce stade, de petites augmentations de rappel sont obtenues au détriment d'une perte importante de précision [Paice, 1994].

- **Comparaison avec light10**

D'après les valeurs obtenues sur tout le corpus, on constate que par rapport au light10, ARLStem v1.1 produit moins d'erreurs de sous-lemmatisation (figure 3.5), d'où la création de moins de lemmes (stems) pour le même groupe de mots. En plus, excepté pour cinq textes pour lesquels la valeur de OI est petite, le lemmatiseur ARLStem v1.1 donne, pour presque tous les textes du corpus, une valeur de OI nulle alors que le Light10 a 13 valeurs non nulle et par conséquence a plus d'erreurs de sur-lemmatisation (figure 3.6).

A titre d'exemple, pour light10, bien que les mots arabes « قران » (le coran) et « قرية » (un village) soient distincts, après lemmatisation, on a le même lemme pour les deux mots. Il y a donc confusion de production d'erreur de sur-lemmatisation.

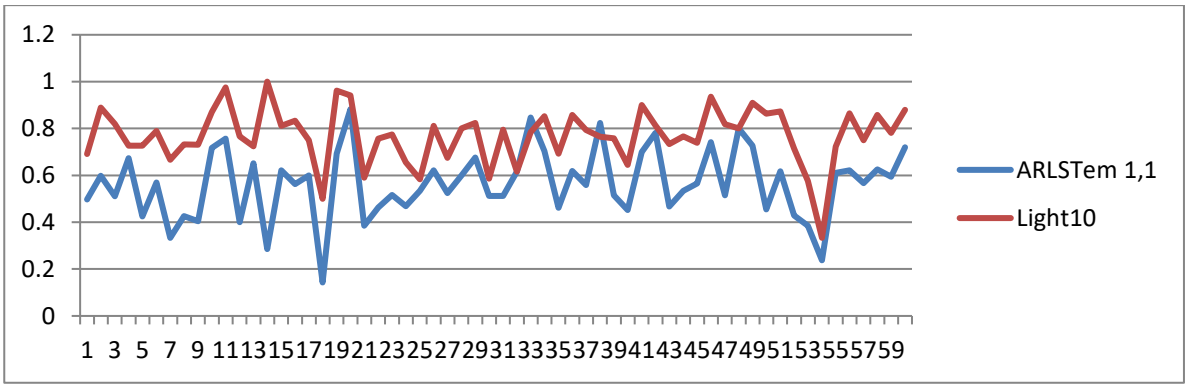

Figure 3.4 : Comparaison en termes d'UI entre ARLStem v1.1 et light10 sur le corpus ARASTEM

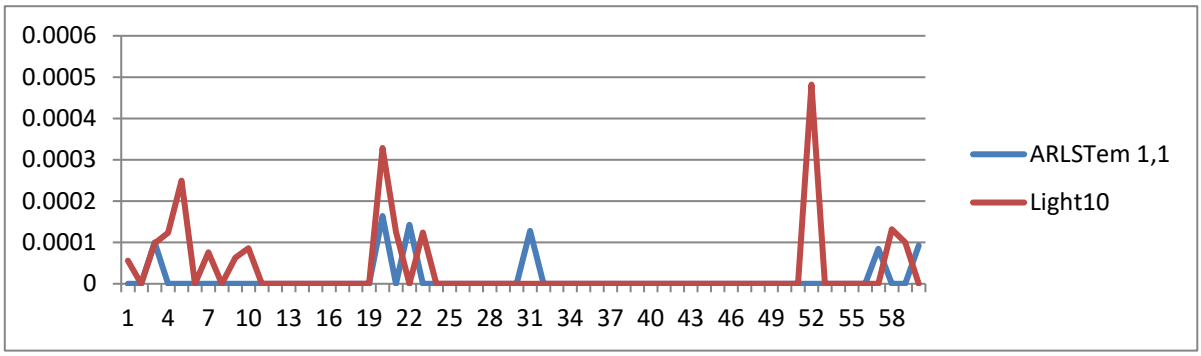

Figure 3.5 : Comparaison en termes d'OI entre ARLStem v1.1 et light10 sur le corpus ARASTEM

- **Comparaison avec ISRI**

Là aussi, on voit que notre programme donne des résultats bien meilleurs que ceux du lemmatiseur ISRI. La figure 3.7 expose une comparaison entre les deux programmes en termes d'indexe de sous-lemmatisation (UI). Quant aux erreurs de sur-lemmatisation (OI), il y en a plus pour l'ISRI que pour notre lemmatiseur (figure 3.8).

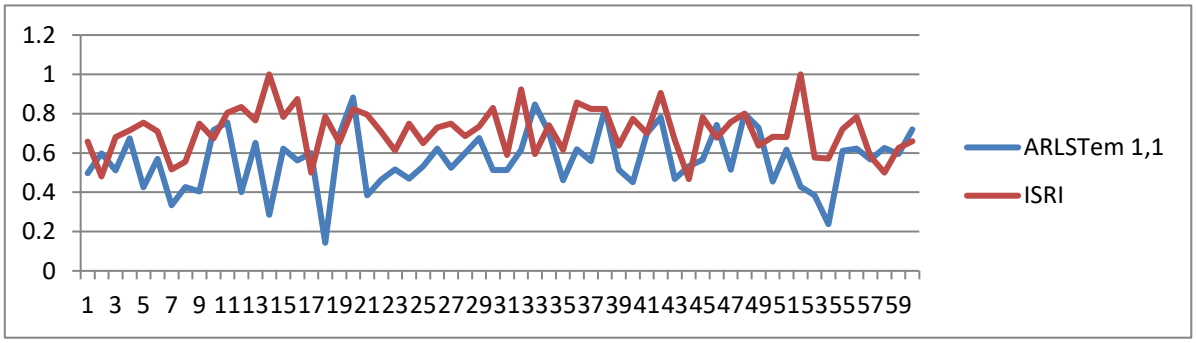

Figure 3.6: Comparaison en termes d'UI entre ARLStem v1.1 et ISRI sur le corpus ARASTEM

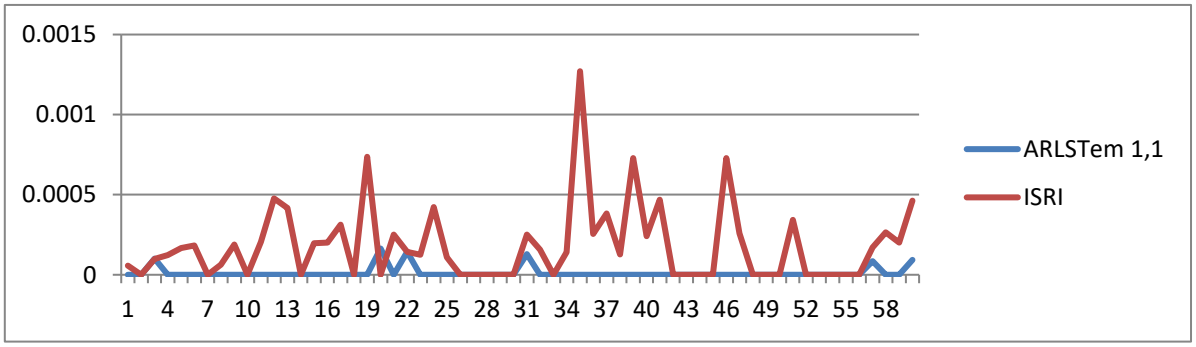

Figure 3.7: Comparaison en termes d'OI entre ARLStem v1.1 et ISRI sur le corpus ARASTEM

- **Comparaison avec lemmatiseur d'Assem**

En termes d'erreurs de sous-lemmatisation (UI), la comparaison des résultats présentée dans la figure 3.9 montrent que notre méthode surpasse celle d'Assem. En d'autres termes ARLStem ne crée pas plusieurs lemmes (stems) pour le même groupe de mots. En comparant les valeurs d'OI des deux lemmatiseurs présentée par la figure 3.10, on constate aussi qu'il n'y a pas une grande différence entre les deux lemmatiseurs.

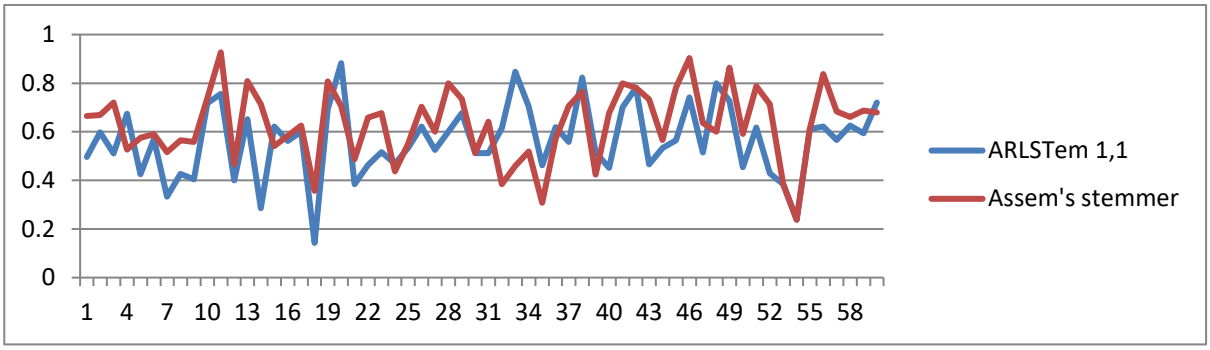

Figure 3.8: Comparaison en termes d'UI entre ARLStem v1.1 et lemmatiseur d'Assem sur le corpus ARASTEM

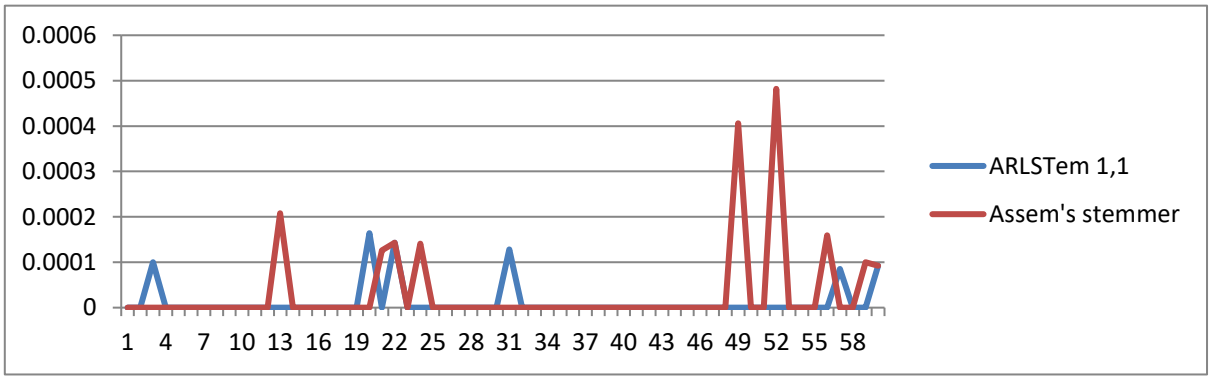

Figure 3.9: Comparaison en termes d'OI entre ARLStem v1.1 et lemmatiseur d'Assem sur le corpus ARASTEM

- **Comparaison avec lemmatiseur de Soori**

La comparaison des résultats présentée dans les figures 3.11 et 3.12 montre que le lemmatiseur ARLStem v1.1 est le plus approprié que celui de Soori, car ce dernier donne plus d'erreurs de sous-lemmatisation (UI) et de sur-lemmatisation (OI). Par exemple, dans le vingtième texte, bien que les mots arabes « احد » et « الحادة » soient distincts, le lemmatiseur de Soori produit le même lemme pour les deux mots (erreur de sur-lemmatisation).

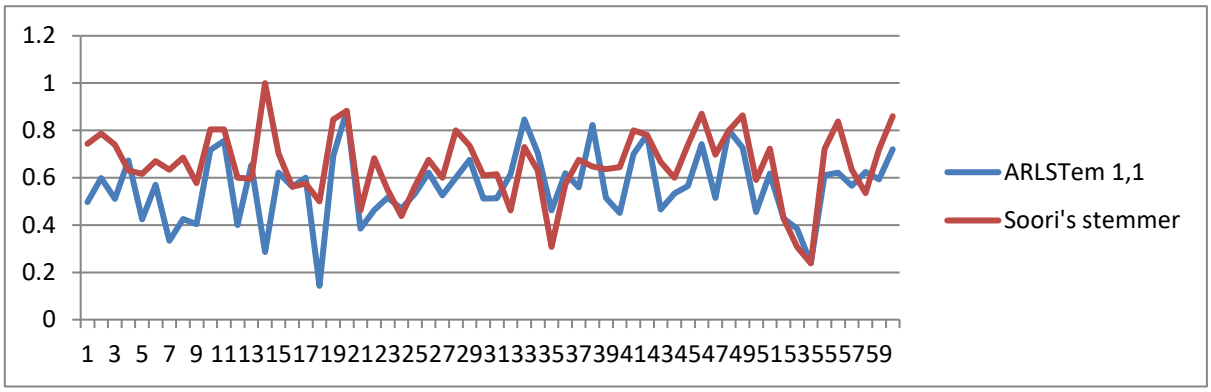

Figure 3.10: Comparaison en termes d'UI entre ARLStem v1.1 et lemmatiseur de Soori sur le corpus ARASTEM

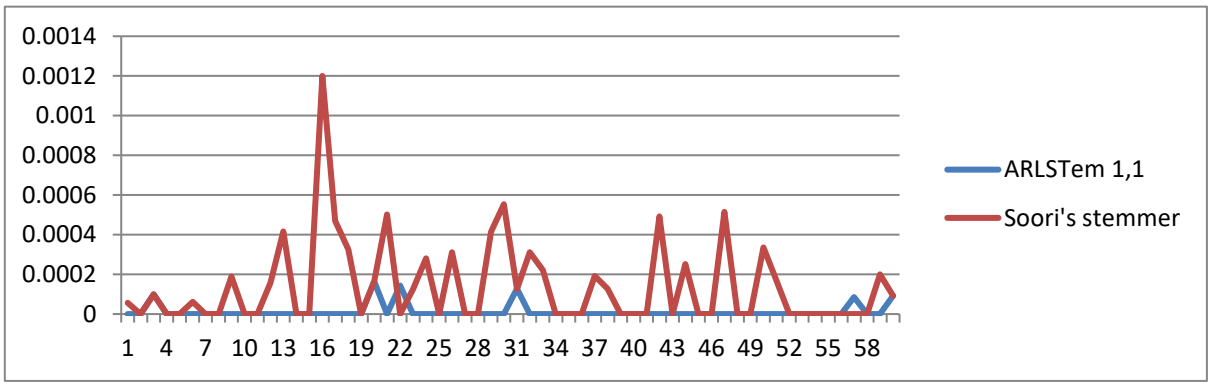

Figure 3.11: Comparaison en termes d'OI entre ARLStem v1.1 et lemmatiseur de Soori sur le corpus ARASTEM

- **Comparaison avec ARLStem 1.0**

Concernant les erreurs de sous-lemmatisation, la comparaison des résultats d'ARLStem v1.0 et v1.1 (figure 3.13) montre que la nouvelle version a donnée des améliorations. Par contre, pour les erreurs de sur-lemmatisation (figure 3.14), on constate une légère dégradation dans les résultats (une erreur dans un fichier de plus). Dans le 22ème fichier, par exemple, ARLStem v1.1 produit le lemme sur-lemmatisé «كاف » en confondant les mots arabes « كافيين » et « كافة », où le premier mot est un nom propre ce qui permet de conclure que c'est un cas exceptionnel.

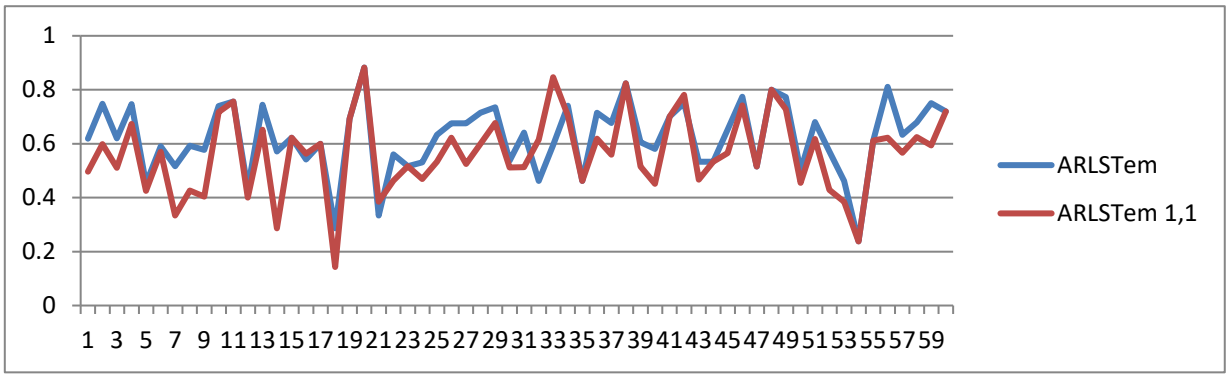

Figure 3.12: Comparaison en termes d'UI entre ARLStem v1.1 et ARLStem v1.0 sur le corpus ARASTEM

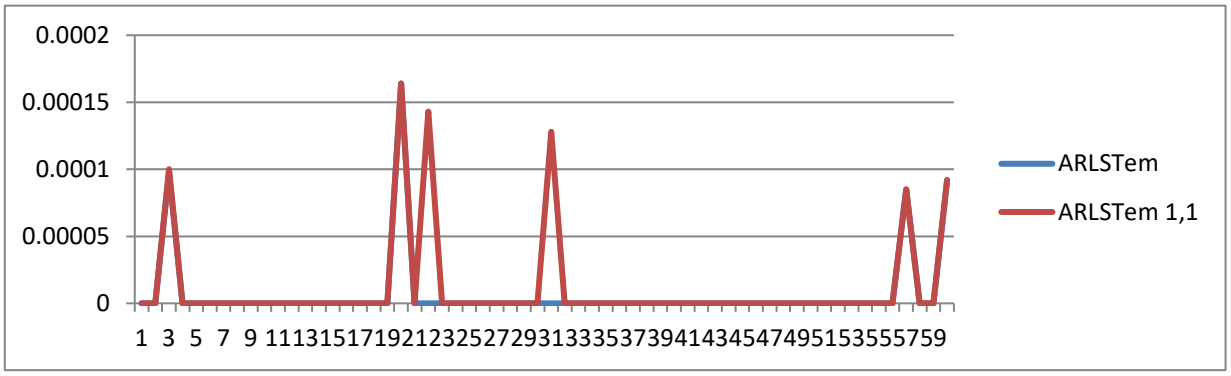

Figure 03.13: Comparaison en termes d'OI entre ARLStem v1.1 et ARLStem v1.0 sur le corpus ARASTEM

3.5. Conclusion

Dans la première partie de ce chapitre, nous avons présenté la version précédente d'ARLStem, puis nous avons abordé les améliorations faites sur cette version. Par la suite, les différents lemmatiseurs de comparaison et le corpus utilisé ont été présentés avec l'exposition des résultats de cette comparaison qui a été réalisée en utilisant les paramètres de Paice (UI, OI).

En termes d'erreurs de sous-lemmatisation et sur-lemmatisation, cette comparaison a montré qu'ARLStem v1.1 est doté de performances très intéressantes.

Conclusion générale

La recherche dans le domaine du traitement de la langue arabe est devenue cruciale pour deux raisons principales: le nombre croissant d'utilisateurs d'Internet dans le monde arabe et le fait que la langue arabe est la sixième langue la plus utilisée dans le monde. Contrairement aux autres langues, la langue arabe possède un système dérivationnel très riche et c'est dans cette caractéristique que réside la difficulté de son traitement. Ces caractéristiques constituent en effet les problèmes majeurs face aux travaux sur la langue arabe dans le domaine de la recherche d'information.

Dans cette optique, nous avons défini une méthodologie pour améliorer la performance de la lemmatisation des mots arabes. Ceci par détermination du lemme le plus précis en intégrant quelques règles, dans le but d'améliorer la performance globale du processus de lemmatisation.

La comparaison de l'efficacité de notre lemmatiseur par rapport à d'autres lemmatiseurs, sur la base du même corpus ARASTEM, en termes de taux d'erreurs de sous-lemmatisation (UI ou Under-stemming Index) et de sur-lemmatisation (OI ou Over-stemming Index), nous permet de conclure que notre lemmatiseur est plus performant.

REFERENCES BIBLIOGRAPHIQUES

[Abainia 2016] Abainia Kheireddine. ‹‹ Catégorisation automatique des conversations textuelles - Application d'aide à l'archivage des forums ››.Thèse de Doctorat. Université des Sciences et Technologie Houairi Boumediène.2016

[Bourane & Berrekbia 2017] Bourane Zohra & Berrekbia Ahlam. ‹‹ Un système de classification et de recherche de documents textuels de la langue Anglaise ››.Mémoire Master. Université Kasdi Merbah Ouargla.2017

[Hammache 2013] HAMMACHE, Arezki. Récherche d'information: un modéle de langue combinant mots simples et mot composés. Thèse de doctorat. Université Mouloud Mammeri. 2013.

[Bouabdellah and Benmansour 2012] BOUABDALLAH, Lamia et BENMANSOUR, Asma. ‹‹ Expansion de requête pour un système de recherche d'information par croisement de langues ››.Mémoire Master. Université de Tlemcen. 2012.

[Jalam 2003] Radwan Jalam. ‹‹ Apprentissage automatique et catégorisation de textes multilingues ››. Thèse de Doctorat. Université Lumière Lyon2.2003.

[Matallah 2011] Matallah Hocine. ‹‹ Classification Automatique de textes Approche Orientée agent››.Memoire Magister. Université Aboubekr Belkaid- Tlemcen. 2011.

[Federalist 2012] MOSTELLER, Frederick et WALLACE, David L. " Springer Science & Business Media ".‹‹Applied Bayesian and classical inference: the case of the Federalist papers››, vol. 110; p 688-696, 2012.

[Sanderson & Croft, 2012] SANDERSON, Mark et CROFT, W. Bruce. The history of information retrieval research. Proceedings of the IEEE, vol. 100, no Special Centennial Issue, p. 1444-1451. , 2012.

[Stamatatos, 2009] STAMATATOS, Efstathios. A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology, vol. 60, no 3, p. 538-556 , 2009.

[Mendenhall, 1887] MENDENHALL, Thomas Corwin. The characteristic curves of composition. Science, vol. 9, no 214, p. 237-249, 1887.

[Mosteller et al., 1963] MOSTELLER, Frederick et WALLACE, David L. Inference in an authorship problem: A comparative study of discrimination methods applied to the

authorship of the disputed Federalist Papers. Journal of the American Statistical Association, vol. 58, no 302, p. 275-309, 1963.

[Mosteller and Wallace, 1964] MOSTELLER, Frederick et WALLACE, David. Inference and disputed authorship: The Federalist. 1964.

[Xia et al., 2010] XIA, Jun-Feng, ZHAO, Xing-Ming, et HUANG, De-Shuang. Predicting protein–protein interactions from protein sequences using meta predictor. Amino acids, vol. 39, no 5, p. 1595-1599, 2010.

[McNamee, 2005] MCNAMEE, Paul. Language identification: a solved problem suitable for undergraduate instruction. Journal of Computing Sciences in Colleges, 2005, vol. 20, no 3, p. 94-101.

[Baldwin et al., 2010]. Newman, D., Lau, J. H., Grieser, K., & Baldwin, T. ‹‹ Automatic evaluation of topic coherence. In Human Language Technologies ››: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics.vol 7, (pp. 100-108). June 2010.

[Hayes and Weinstein, 1990], HAYES, Philip J. et WEINSTEIN, Steven P. CONSTRUE/TIS: ‹‹A System for Content-Based Indexing of a Database of News Stories››. p. 49-64. 1990.

[Lang, 1995] LANG, Ken. Newsweeder: Learning to filter netnews. In : Machine Learning Proceedings 1995. Morgan Kaufmann, 1995. p. 331-339.

[Armstrong et al, 1995] ARMSTRONG, Robert, FREITAG, Dayne, JOACHIMS, Thorsten, et al. Webwatcher: A learning apprentice for the world wide web. In : AAAI Spring symposium on Information gathering from Heterogeneous, distributed environments. 1995. p. 107.

[Martins et al., 2005] MARTINS, Bruno et SILVA, Mário J. A graph-ranking algorithm for geo-referencing documents. In : Fifth IEEE International Conference on Data Mining (ICDM'05). IEEE, 2005. p. 4 pp.

[Abainia et al., 2014] ABAINIA, Kheireddine, OUAMOUR, Siham, et SAYOUD, Halim. Robust language identification of noisy texts: proposal of hybrid approaches. In : 2014 25th

International Workshop on Database and Expert Systems Applications. IEEE, 2014. p. 228-232.

[Liddy et al., 1994] LIDDY, Elizabeth D., PAIK, Woojin, YU, Edmund S., et al. Document retrieval using linguistic knowledge. In : Intelligent Multimedia Information Retrieval Systems and Management-Volume 1. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, 1994. p. 106-114

[Kadri 2008] Youssef Kadri. ‹‹ Recherche d'information Translinguistique sur les documents en Arabe ››. Thèse de Doctorat. Université de Montréal.2008.

[Abbasi & Meftah 2013] ABBASSI WALID et MEFTAH BELAL. Un modèle de reformulation des requêtes pour la recherche d'information sur le Web. Thèse de doctorat. Université Kasdi Merbah Ouargla.2013.

[Benblal & Belouafi 2015] Zoulikha Benblal & Fatima Belouafi. ‹‹ Intégration d'un lemmatiseur arabe dans le cadre d'un système de recherche d'informaion ››.Mémoire Master.Ahmed Draia-Adrar.2015.

[Boulaknadel 2008] Siham Boulaknadel. ‹‹ Traitement Automatique des Langues et Recherche d'Information en langue arabe dans un domaine de spécialité : Apport des connaissances morphologiques et syntaxiques pour l'indexation ››. Thèse de Doctorat. Université de Nantes.2008

[Fayet and Scribe, 1997] SYLVIE, FAYET-SCRIBE. ‹‹ Chronologie des supports, des dispositifs spatiaux, des outils de repérage de l'information›› .Thèse de Doctorat, 1997.

[Zeroul & Lakhouaja, 2017] Zeroual, Imad, and Abdelhak Lakhouaja. "Arabic information retrieval: Stemming or lemmatization?." 2017 Intelligent Systems and Computer Vision (ISCV). IEEE, 2017.

[Chouchoui & Brahimia, 2016] Chouchoui Maissa & Brahimia Yamna Affaf. ‹‹ Détection Automatique de la Cohésion Lexicale entre phrases dans les textes Arabes ››. Mémoire Master. Université de Djilali Bounaama Khemis Miliana.2016.

[Cheragui et all ;2015] Cheragui Mohamed Amine & Chougueur Djilali. ‹‹ Conception et réalisation d'un lemmatiseur hybride de texte arabe›› . Mémoire Master. Université Ahmed Draya-Adrar, 2015.

[Khemakhem, 2006] Aïda KHEMAKHEM, "ArabicLDB : une base lexicale normalisée pour la langue arabe", mémoire de MASTER en Systèmes d'Information et Nouvelles technologies, Université de Sfax, Tunisie, 2006.

[Bourezg 2017] Bourezg Aissa. « Implementation d'une methode d'analyse morpho-lexicale pour la langue arabe basée sur la position des lettres ».Memoire Master. Université Mouhamed Boudiaf-M'SILA.2017.

[Boubekeur 2016] Boubekeur Yassamina. ‹‹ Identification automatique de motes clés dans les textes Arabes ››. Mémoire Master. Université de Djilali Bounaama Khemis Miliana.2016.

[Chen & Gey, 2002] Chen, Aitao, and Fredric Gey. "Building an Arabic stemmer for information retrieval." TREC. Vol. 2002. 2002.

[AL-ABWEENY et all, 2018] AL-ABWEENY, Waed Waleed et ZAID, Nahed Abu. Arabic Stemmer System based on Rules of Roots. International Journal of Information Technology and Language Studies, vol. 2, no 1, 2018.

[Saadane, 2015] Houda Saadane. ‹‹ Le traitement automatique de l'arabe dialectalisé : aspects méthodologiques et algorithmiques ››.Thèse Doctorat. Université Grenoble Alpes. 2015.

[Benhalima, 2017] Benhalima Maissa. . ‹‹ Implémentation d'une méthode hybride (Morphologique & statistique) pour l'analyse des mots arabes ››.Mémoire Master. Université Mohamed Boudiaf-M'SILA.2017.

[Mesfar, 2008] Slim Mesfar. ‹‹ Analyse Morpho-Syntaxiqe Automatique et reconnaissance des entités nommées en Arabe Standard ››.Thèse Doctorat. Université de Franche-Compte.2008.

[Dariouache 2016] dariouache Adnane "Etude et réalisation d'un analyseur morphologique de la langue arabe" Projet de Fin d'Etudes Master Sciences et Techniques, université de Maroc Sidi Mohamed Ben Abdellah, 2015.

[Ed-Dariouache.2015] Ed-Dariouache Adnane. ‹‹ Etude et réalisation d'un analyseur morphologique de la langue Arabe ››. Mémoire Master. Université Sidi Mohamed Ben Abdellah.2015.

[Benzater, 2015] Benzater Nebia**,** " analyse morphologique de texte arabe pour son indexation sémantique", Mémoire de Magistère en Informatique, Université des Sciences et de la Technologie - Mohamed Boudiaf – Oran, 2014-2015.

[M. Al-Badrashiny et al., 2009]. M. Rashwan, M. Al-Badrashiny, M. Attia, S.M. Abdou, "A hybrid system for automatic arabic diacritization», The 2nd International Conference on Arabic Language Resources and Tools, Egypt 2009

[zeroual 2017**]** Zeroual, Imad, et al. "Developing and performance evaluation of a new Arabic heavy/light stemmer." Proceedings of the 2nd international Conference on Big Data, Cloud and Applications. ACM, 2017.

[AL Hajjar, 2010]Abd El Salam AL HAJJAR,"extraction et gestion de l'information a partir des documents arabes", thèse de Doctorat, UNIVERSITE PARIS VIII, SAINT DENIS ,2010.

[M. Sanan, 2008] M. Sanan," Etude Des Méthodes De La Recherche D'information Et De L'indexation Sur Les Documents Electroniques : Cas De La Langue Arabe", thèse de Doctorat, UNIVERSITE PARIS VIII - SAINT DENIS, 2008.

[Baloul 2003] Baloul Sofiane, "Développement d'un système automatique de synthèse de la parole à partir de texte arabe standard voyelle", mémoire de doctorat d'informatique, Université du MarieFrance, 2003.

[LAMRANI et al., 2014] LAMRANI, El Khadir, MARZAK, Abdelaziz, EL GUEMMAT, Kamal, et al. MÉTHODES DE CLUSTERING DES DOCUMENTS TEXTES ARABES: ÉTUDE COMPARATIVE. In : 5th International Conference on Arabic Language Processing (CITALA 2014). 2014.

[AL-OMARI et ABUATA., 2014] AL-OMARI, Asma et ABUATA, Belal. Arabic light stemmer (ARS). Journal of Engineering Science and Technology, 2014, vol. 9, no 6, p. 702-717.

[Larkey L. and M. E. Connell, 2001] Larkey L. and M. E. Connell. "Arabic information retrieval at UMass in TREC-10". Proceedings of TREC 2001, Gaithersburg: NIST. 2001.

[Benblal & Belouafi, 2015] Zoulikha Benblal & Fatima Belouafi. ‹‹ Intégration d'un lemmatiseur arabe dans le cadre d'un système de recherche d'informaion ››.Mémoire Master.Ahmed Draia-Adrar.2015.

[M.Al-Kabi, 2013]. M.Al-Kabi,"Towards improving Khojarule-based Arabic stemmer." Proceedings of Applied Electrical Engineering and Computing Technologies(AEECT),IEEE Jordan Conference on, 2013.

[R. Al-Shalabi and M. Evens., 1998] R.Al-Shalabiand and M.Evens,"Acomputational morphology system for Arabic," presented at the Proceedings of the Workshopon Computational Approaches to Semitic Languages,Montreal, Quebec, Canada, 1998.

[Bahassine et all., 2014] Bahassine, Said, Mohamed Kissi, and Abdellah Madani. "New stemming for Arabic text classification using feature selection and decision trees." Proceedings of the 5th International Conference on Arabic Language Processing. 2014.

[K. Darwish and D. W. Oard., 2002] K. Darwish and D. W. Oard. CLIR experiments at Maryland for TREC-2002:Evidence combination for Arabic-English retrieval. In Proceedings of the Text REtrieval Conference (TREC-11), pages 703710, 2002.

[A. Chen and F. Gey., 2002] A. Chen and F. Gey. Building an Arabic stemmer for information retrieval. In Proceedings of the Text REtrieval Conference (TREC-11), pages 631639, 2002.

[Alhanini 2011]. ALHANINI, Yasir et AB AZIZ, Mohd Juzaiddin. The enhancement of Arabic stemming by using light stemming and dictionary-based stemming. Journal of Software Engineering and Applications, 2011, vol. 4, no 09, p. 522.

[Hadni 2013]. HADNI, Meryeme, OUATIK, Said Alaoui, et LACHKAR, Abdelmonaime. Effective Arabic stemmer based hybrid approach for Arabic text categorization. International Journal of Data Mining & Knowledge Management Process, 2013, vol. 3, no 4, p. 1.

[Al-Nashashibi., 2010]. Al-Nashashibi, May Y., D. Neagu, and Ali A. Yaghi. "Stemming techniques for Arabic words: A comparative study." 2010 2nd International Conference on Computer Technology and Development. IEEE, 2010.

[Khedr., 2005] KHEDR, Eman M., KOTB, H., KAMEL, N. F., et al. Longlasting antalgic effects of daily sessions of repetitive transcranial magnetic stimulation in central and peripheral neuropathic pain. Journal of Neurology, Neurosurgery & Psychiatry, 2005, vol. 76, no 6, p. 833-838.

[Beltagy., 2009] EL-BELTAGY, Samhaa R. et RAFEA, Ahmed. KP-Miner: A keyphrase extraction system for English and Arabic documents. Information Systems, 2009, vol. 34, no 1, p. 132-144.

[De Roeck and Al-Fares, 2000] DE ROECK, Anne N. et AL-FARES, Waleed. A morphologically sensitive clustering algorithm for identifying Arabic roots. In : Proceedings of the 38th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2000. p. 199-206.

[Paice, 1994] Paice, Chris D. "An evaluation method for stemming algorithms." SIGIR'94. Springer, London, 1994.

[Paice, 1996] PAICE, Chris D. ‹‹ Method for evaluation of stemming algorithms based on error counting ››. "Journal of the American Society for Information Science", vol. 47, no 8, p. 632-649. 1996.

[Al-Shammari et al. 2008] Al-Shammari, Eiman, and Jessica Lin. "A novel Arabic lemmatization algorithm." Proceedings of the second workshop on Analytics for noisy unstructured text data. ACM, 2008.

[Larkey et al., 2002] Larkey, Leah S., Lisa Ballesteros, and Margaret E. Connell. "Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis." Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2002.

[Zaghloul et al., 2009] ZAGHLOUL, Kareem A., BLANCO, Justin A., WEIDEMANN, Christoph T., et al. Human substantia nigra neurons encode unexpected financial rewards. Science, 2009, vol. 323, no 5920, p. 1496-1499.

[He et al., 2000] AMARAL, Luıs A. Nunes, SCALA, Antonio, BARTHELEMY, Marc, et al. Classes of small-world networks. Proceedings of the national academy of sciences, 2000, vol. 97, no 21, p. 11149-11152.

[Al-Shargabi et al., 2011] AL-SHARGABI, Bassam, OLAYAH, Fekry, et ROMIMAH, Waseem AL. An experimental study for the effect of stop words elimination for arabic text classification algorithms. International Journal of Information Technology and Web Engineering (IJITWE), 2011, vol. 6, no 2, p. 68-75.

[Kanaan et al., 2009]. KANAAN, Ghassan, AL-SHALABI, Riyad, GHWANMEH, Sameh, et al. A comparison of text-classification techniques applied to Arabic text. Journal of the American society for information science and technology, 2009, vol. 60, no 9, p. 1836-1844.

[NAMLY et al., 2017]. JAAFAR, Younes, NAMLY, Driss, BOUZOUBAA, Karim, et al. Enhancing Arabic stemming process using resources and benchmarking tools. Journal of King Saud University-Computer and Information Sciences, 2017, vol. 29, no 2, p. 164-170.

[Bahassine 2014]. Bahassine, Said, Mohamed Kissi, and Abdellah Madani. "New stemming for Arabic text classification using feature selection and decision trees." Proceedings of the 5th International Conference on Arabic Language Processing. 2014.

[Ababneh et al., 2012] M. Ababneh, R. Al-Shalabi, G. Kanaan et A. Al-Nobani, Building an Effective RuleBased Light Stemmer for Arabic Language to Improve Search Effectiveness, International Arab Journal of Information Technology (IAJIT), Vol. 9, No. 4, 2012.

[Lar et al, 06]: Larkey L. S., Ballesteros, L, & Connell, M. E « Light Stemming for Arabic Information Retrieval Center for Intelligent Information Retrieval and in part by SPAWARSYSCENSD ».