REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE RECHERCHE SCIENTIFIQUE

Université d'Adrar

Faculté des Sciences et de la Technologie

Département des Mathématiques et Informatique

Mémoire Préparé en Vue de l'Obtention Du diplôme de Master en Informatique

Option: Réseaux et Systèmes Intelligents

Thème:

Acquisition de Connaissances à partir d'un texte Arabe non vocalisé (JEEM BOX)

Présenté et soutenu publiquement par

Mr.DAHOU Abdelghani

Le 16 juin 2014, devant le jury ci-dessous

Président : Mr.CHOGUEUR Djilali

Examinateur : Mr.KOHILI Mohamed

Encadreur : Mr.CHERAGUI Mohamed Amine

Juin 2014

Remerciements

Il est de coutume de dire qu'une thèse n'est pas le fruit du seul travail de son auteur, mais le résultat de nombreuses et étroites collaborations; celle-ci ne déroge pas à la règle.

Nous remercions avant tout le Bon Dieu de nous avoir donné la volonté de finir ce mémoire.

Ce travail a pu voir le jour avec énormément d'aide et encouragement des personnes autour de nous. Ce court remerciement ne sera pas suffisant pour récompenser leurs efforts mais tout de même ...

A l'issue de deux agréables années au sein de département d'informatique de l'université d'ADRAR nous tenons à remercier l'ensemble des enseignants pour leur dévouement sans oublier d'adresser des remerciements particuliers à Monsieur KHALLADI, le chef du département, pour le dynamisme de ce département d'études, à Monsieur CHOGUEUR et à Monsieur KOHILI.

Ensuite, toutes nos pensées de gratitude se dirige vers notre encadreur Mr.CHERAGUI Mohamed Amine, qui en tant qu'un encadreur de mémoire, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu nous consacrer, et qui sans son concours ce mémoire n'aurait jamais vu le jour.

Nous tenant aussi à remercier les membres du jury qui ont accepté d'examiner notre mémoire.

Nous adressons nos plus sincères remerciements à tous nos collègues et nos amis qui partagent avec nous les bons moments de l'étude pendant les deux années.

Nous exprimons nos gratitude à tous nos proches qui nous ont toujours soutenue et encouragée au cours de la réalisation de ce mémoire.

Enfin, tous ceux qui ont Contribués, de près ou de loin à la réalisation de cette thèse et que nous ne pouvons malheureusement citer, trouvent ici l'expression de notre profonde gratitude.

Dédicace

Je dédie ce travail à

Mes très chers parents.

Mes très chers frères.

À toute ma famille

Département d'Informatique d'ADRAR

Mr.KHALADI Mohamed Taha

Mr.MAZOUZI Hadj

Mr.CHERAGUI Mohamed Amine

Mr.OMARI Mohamed

Mr.KOHILI Mohamed

Mr.CHOGUEUR Djilali

Tous mes amis, ainsi que toute la promotion du Master 2 en

Informatique 2013/2014 de l'université d'ADRAR.

Table des matières

Liste des figures

Liste des tables

Résumé

Le travail que nous avons effectué consiste à développer une boite à outils, pour l'acquisition de connaissance à partir d'un texte arabe non vocalisé, basée sur les différentes approches et techniques issues du traitement automatique du langage naturel. La boite à outil que nous avons élaboré s'appelle JEEM BOX.

Ce travail se compose de trois (03) parties principales : la première présente un état de l'art et fournit le cadre théorique et méthodologique de notre travail et la deuxième partie est consacrée à la conception et la réalisation des outils informatique constituant JEEM BOX (Les outils sont: JStem pour la lemmatisation, JTrans pour la translitération, JClass pour la classification des mots ayant la même racine, JConcord pour la classification des mots par fréquence, JDiac pour la vocalisation des mots arabes et enfin JExtract pour l'extraction des informations se basant sur les différentes catégories grammaticales du mot arabe). Et la troisième partie est l'analyse expérimentale des outils de JEEM BOX.

Mots clés

JEEM BOX, traitement automatique du langage naturel, lemmatisation, classification, extraction, vocalisation, fréquence, catégories grammaticales.

ملخص

العمل المُقدم يتمحور حول تطوير و يرمجمة علبة من الأدوات لإكتساب المعرفة من النص العربي الغير مشكل، بالإعتماد على تقنيات لغوية للمعالجة اللآلية للغة الطبيعية. علبة الأدوات المطورة من طرفنا تدعى JEEM BOX.

العمل مقسم إلى ثلاثة أقسام : القسم الأول يعنى بطرح خلفية عن الأعمال السابقة و يوفر الإطار النظري و المنهجي لعملنا، اما القسم الثاني يعنى بتصميم و برمجة الأدوات الخاصة بـ JEEM BOX (الأدوات هي : JStem و هي أداة لإستخراج جذور الكلمات، JTrans و هي أداة لتحويل حروف اللغة العربية إلى حروف باللغة اللاتينية، JClass و هي أداة لتصنيف الكلمات في النص في أقسام حسب نفس الجذر المكون لها، JConcord و هي أداة لتصنيف الكلمات في قائمة حسب ترددها في النص، JDiac و هي أداة لتشكيل النص تلقائيا، JExtract و هي أداة لإستخراج المعلومات بالإعتماد على الفئات النحوية للنص العربي). أما القسم الثالث و الأخير فيعنى بتجريب و تحليل الأدوات المبرمجة في JEEM BOX.

الكلمات المفتاحية

الجذر، المعالجة الآلية للغة الطبيعية، المرفولوجية، تشكيل، تحويل، تصنيف، تردد، إستخراج، الفئات النحوية.

Abstract

The work we have done is the development of a toolbox for the acquisition of knowledge from an unvowelled Arabic text based on linguistic techniques for the automatic processing of natural language. The toolbox we developed called JEEM BOX.

This work consists of three main parts: the first presents a state of art and provides the theoretical and methodological framework of our work and the second is devoted to the design and the implementation of JEEM BOX tools (the tools are: JStem for lemmatization, JTrans for transliteration, JClass for the classification of words based on their root, JConcord for the classification of words by their frequency, JDiac for vocalization and JExtract for the information extraction based on the different grammatical categories of the Arabic word). And the third part is for the experimental analysis of JEEM BOX tools.

Keywords

Toolbox, JEEM BOX, automatic processing of natural language, lemmatization, transliteration, classification, root, frequency, vocalization, extraction, grammatical categories.

INTRODUCTION GÉNÉRALE

1. Introduction

L'information joue aujourd'hui un rôle de plus en plus important dans toutes les activités du monde contemporain. En effet, de nombreux facteurs, agissant de manière synergique, ont contribué au cours de ces trente dernières années, à accroître son influence dans tous les domaines. Simultanément, l'internationalisation des marchés et le développement des Nouvelles Technologies de l'Information et de la Communication (NTIC) ont favorisé le multilinguisme et l'augmentation du volume d'information.

Dans ce contexte, les aspects d'acquisition, de gestion, d'analyse, d'exploitation, ...etc des informations, multilingues ou non, pour la plupart sous formes textuelles ou orales, sont au centre des grands débats du monde de la recherche et de l'économie. La société de l'information donne par ailleurs au support de l'information, la langue elle-même, une importance nouvelle sur le plan de son traitement informatique.

Le domaine des technologies de la langue est ainsi devenu un des domaines-clés qui propose des voies de recherche susceptibles d'apporter des solutions à ces problèmes et répond ainsi aux besoins actuels de ce que les experts dénomment la « société de l'information ».

Le présent travail entre dans le cadre d'acquisition des connaissances de texte arabe non vocalisé un sous domaine du Traitement Automatique du Langage Naturel (TALN).

Le TALN a connu des évolutions très rapides ces dernières années, et spécialement le traitement de la langue arabe, c'est pourquoi les demandes en matière d'applications fiables augmentent sans cesse. De ce fait, nous nous sommes intéressés à ce domaine afin de concevoir une boite à outils d'acquisition des connaissances pour la langue arabe, Plusieurs principaux axes sont étudiés : lemmatisation, translitération, classification, concordance, vocalisation et étiquetage afin de réaliser ce travail.

2. Objective de l'étude

2.1 Objectif général

Le travail que nous avons effectué consiste en le développement d'une boite à outils avec base de données pour le traitement automatique de la langue arabe (acquisition des connaissances). La boite à outils que nous avons élaborés s'appelle JEEM BOX. JEEM BOX est une boite à outils programmé pour Microsoft Windows et bénéficie d'une interface utilisateur puissante, agréable et interactive.

2.2 Objectifs spécifiques

Les objectifs spécifiques de ce travail sont les suivants:

- Étudier les différents algorithmes de lemmatisation qui ont été développés pour la langue Arabe;
- Sélectionnez le corpus (texte Arabe) de test et le préparer pour le traitement ;
- Construire un lemmatiseur hybride pour la langue arabe.
- Construire un outil de translitération pour la langue arabe.
- Construire deux outils : un pour la calcule de la concordance des mots dans un texte arabe et l'autre pour la classification des mots d'un texte arabe dans des familles de même racine.
- Construire un outil de vocalisation de texte arabe.
- Construire un outil pour l'extraction des informations d'après un texte arabe étiqueté.
- Évaluer les outils afin de mesurer leurs efficacité;

3. Organisation du mémoire

Ce mémoire s'articule autour de quatre (04) chapitres, comme suit :

- Dans le premier chapitre, nous présentons une introduction au traitement automatique du langage naturel, nous exposons son objectif et nous donnons un bref aperçu historique du domaine. Ensuite, nous présentons les différents niveaux de base nécessaires pour le traitement du langage naturel. Enfin, nous présentons quelque application du domaine et ces différentes tâches.
- Le second chapitre explique la morphologie de la langue arabe avec sa complexité qui nous intéresse pour développer notre boite à outils et cité quelque exemple d'outil de traitement automatique de la langue arabe.
- Dans le troisième chapitre, qui peut être considéré comme le cœur de notre travail on a essayé de donner une : Conception générale de l'outil proposé : nous décrivons l'architecture et la conception des outils proposées.
- Le quatrième chapitre explique les différentes étapes d'implémentation de notre boite à outils ainsi que les résultats obtenus.
- Finalement nous conclurons notre travail avec la présentation des perspectives essentielles qui sont susceptibles d'enrichi d'avantage et affiner nos contributions.

CHAPITRE I

Traitement automatique des langues naturelles

Principes et concepts de base

1. Introduction

Le traitement automatique des langues (T.A.L.) ou NLP (Natural Language Processing en Anglais) est un domaine de recherche pluridisciplinaire, qui fait collaborer des linguistes, informaticiens, logiciens, psychologues, documentalistes, lexicographes ou traducteurs, et qui appartient au domaine de l'Intelligence artificielle (I.A).

Le domaine du traitement automatique des langues s'organise autour des questions de conception et modélisation des actes de production et de perception (reconnaissance, compréhension) des énoncés de la langue naturelle, en vue de leur accomplissement ou de leur simulation par des machines. Il constitue un point de rencontre entre différents disciplines de connaissances :

Linguistique, sciences cognitives, psychologie expérimentale ; il mobilise, adapte et contribue à enrichir, en plus des modèles propres à ces disciplines, des outils empruntés à de multiples domaines :

Traitement du signal, mathématiques et informatique théorique (logiques, langages formels), intelligence artificielle (représentation des connaissances, apprentissage), ...etc [1].

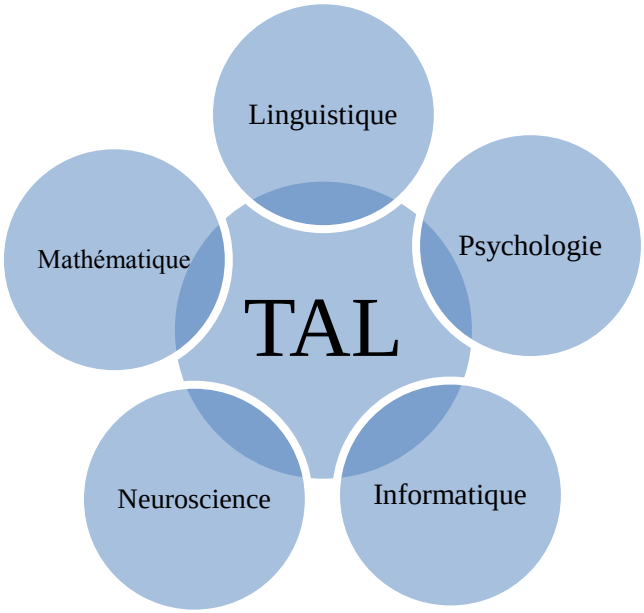

Figure 1: *Domaines de recherche du TAL*

Le but de ce chapitre est de donner un état des lieux concernant le traitement automatique du langage naturel à travers son historique, Objectifs, les différents niveaux de traitements, mais aussi les domaines d'applications.

2. Objectif du TALN

L'objectif des traitements automatiques des langues est la conception de logiciels ou programmes, capables de traiter de façon automatique des données linguistiques, c'est-à-dire des données exprimées dans une langue (dite "naturelle").Ces données linguistiques peuvent être des textes écrits, ou bien des dialogues écrits ou oraux, ou encore des unités linguistiques de taille inférieure à ce que l'on appelle habituellement des textes (par exemple : des phrases, des énoncés, des groupes de mots ou simplement des mots isolés) [2]. Ce traitement nécessite l'élaboration d'outils et de méthodes automatiques qui sont de trois ordres: linguistiques, formels et informatiques [2]. Parmi les logiciels d'analyse linguistique, citons les logiciels d'étiquetage morphosyntaxique et de parsage, qui sont à la base de la plupart des applications en TAL (traduction automatique, traitement de la parole, etc…).

3. Histoire du Traitement automatique des langages Naturelles

Historiquement, Le traitement automatique du langage naturel (TALN) est né à la fin des années quarante dans un contexte scientifique imprimé par les premiers travaux sur la traduction mais aussi dans un contexte politique qui s'explique par la seconde guerre mondiale. Le but de ce point est de donner quelques dates marquantes dans le développement du traitement automatique du langage naturel à travers le monde [1] [3] [4] [5]:

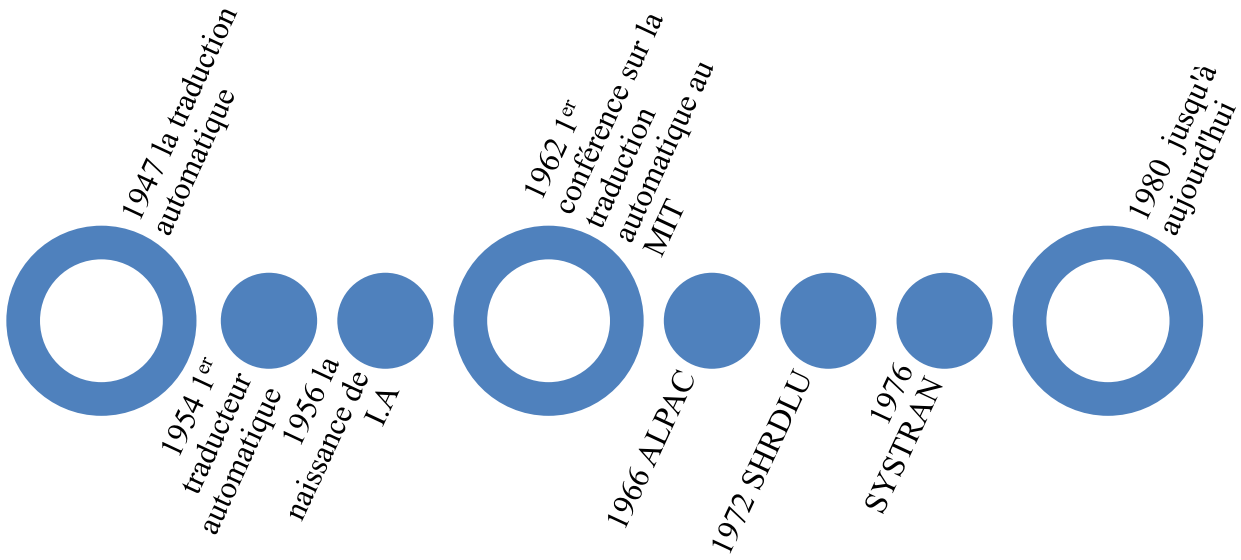

Figure 2: *Les dates Marquantes dans l'histoire du TAL N*

- 1947 : Début des travaux sur la traduction automatique;
- Entre 1951 et 1954 : Zelling Harris publie ses travaux les plus importants de la linguistique (linguistique distributionnaliste) ;
- 1954 : La mise au point du premier traducteur automatique (très rudimentaire) qui traduit du Russe à l'Anglais ;
- 1956 : L'école de Dartmouth (au Etats-Unis) et la naissance de l'Intelligence Artificielle (I.A) sous l'influence de plusieurs figures marquantes de cette époque : J. McCarthy, Marvin Minsky, Allan Newell et Herbert Simon qui discutent sur les possibilités de créer des programmes d'ordinateurs qui se comportent intelligemment et en particulier qui soient capables d'utiliser le langage naturel ;
- 1957 : N. Chomsky publie ses premiers travaux sur la syntaxe des langues naturelles, et sur les relations entre grammaire formelles et grammaire naturelles ;
- 1962 : la première conférence sur la traduction automatique est organisée au MIT (Institut Technologique du Massachussetts) par Y. Bar-Hillel ;
- Entre 1961 et 1966 : beaucoup d'applications ont été mis en place tel que : BASBEL, SIR, STUDENT, ELIZA, …etc. Mettant en œuvre des mécanismes de traitement simple, à base de mots clés ;
- 1966 : L'histoire du TAL fait souvent celle des rendez-vous manqués et des désillusions cruelles Parmi ces faits marquants, on peut citer le rapport de la commission ALPAC (Automatic Language Processing Advisory Committee) en Anglais qui s'interroge sur l'utilité de poursuivre les recherches dans ce domaine. Dès lors, les crédits sont considérablement réduits et la recherche stagne jusqu'au début des années 70 ;
- Depuis 1970, la plupart des recherches visent surtout la sémantique dans le cadre de la compréhension, mais aussi en parallèle les modèles syntaxiques connaissent en informatique des développements et des raffinements continus, et des algorithmes de plus en plus performants sont proposées pour analyser les grammaires les plus simples.
- 1972 : Terry Winogard, réalise le premier logiciel appelé SHRDLU capable de dialoguer en anglais avec un robot ;
- 1976 : L'installation d'un système de traduction automatique commercial nommé SYSTRAN, la traduction automatique se fait connaître du grand public et suscite à nouveau l'intérêt des firmes privés que ce soit au Etats Unis ou au Japon ;
- Entre 1980 jusqu'à aujourd'hui : La recherche en traitement automatique du langage naturel a connu depuis les années 80 jusqu'à nos jours une véritable progression, en termes de performance[1] qui se traduit

[1]Le rendement des solutions proposées atteint aujourd'hui un certain seuil de fiabilité, qui se traduit par des pourcentages élevées de bon traitement.

d'un côté par la diversification des applications industrielles[2], mais aussi d'un autre côté par la création de plusieurs conférences internationales de renommées[3] et de laboratoires[4] de recherches à travers le monde.

4. Les Niveaux de Traitement Automatique des langages naturels

Pour traiter le langage naturel, on a besoin d'informations coordonnées et pertinentes sur la langue à des niveaux divers. Nous introduisons dans cette section les différents niveaux de traitement nécessaires pour parvenir à une compréhension (analyse linguistique) complète d'un énoncé en langage naturel. Du point de vue des TALIST's[5], ces niveaux correspondent à des modules qu'il faudrait développer et faire coopérer dans le cadre d'une application complète de traitement de langage naturel [4]. Pour cela, on distingue quatre (04) modules (respectivement niveaux) de traitement où chaque module (respectivement niveau) a une tâche bien précise :

- Un module de reconnaissance (morphologique).
- Un module de structuration (niveau syntaxique).
- Un module de compréhension (niveau sémantique).
- Un module de conceptualisation (niveau pragmatique).

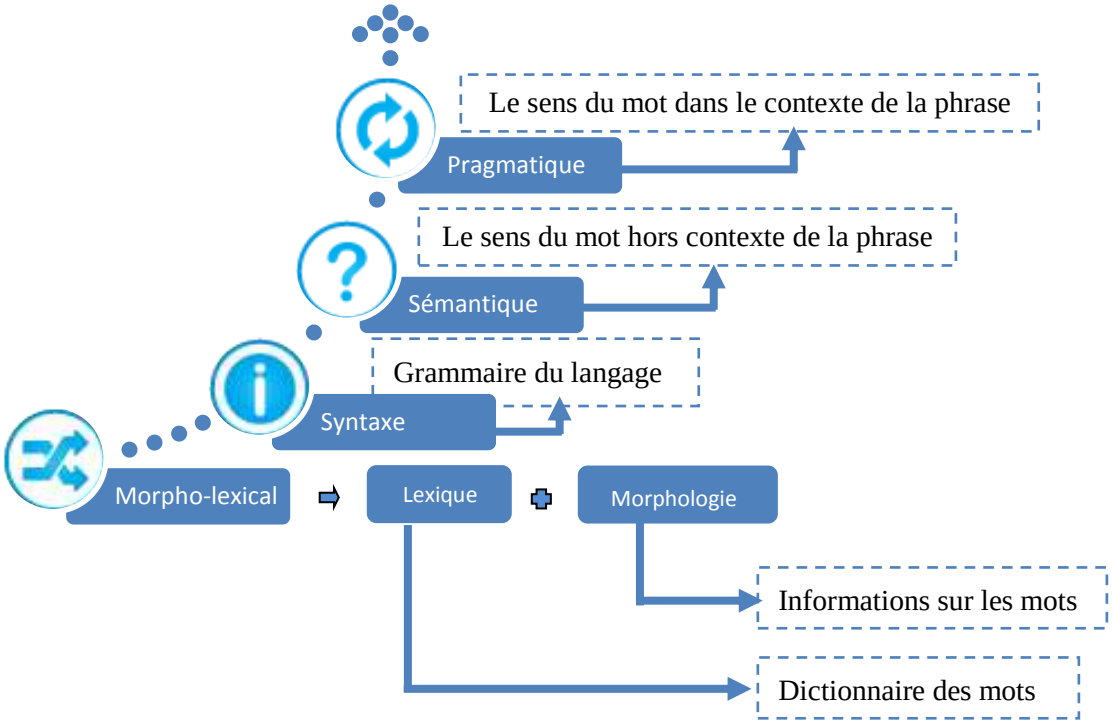

Figure 3: *Représentation des niveaux de traitement du langage naturel*

[2] Beaucoup de grandes entreprises, se sont alliées à ce domaine tel que : IBM, XEROX, …etc.
[3] Exemple de conférences : ATALA, ACL-EACL, ANLP, ICASSP, …etc.
[4] Exemple de laboratoire : CERTAL (Centre d'Etude et de Recherche en Traitement automatique des Langues) en France. CRSTDLA (Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe) en Algérie.
[5] TALIST : c'est la personne qui travail sur l'automatisation du langage naturel.

4.1 Niveau morphologique

Le niveau de traitement morpho-lexical [6] est l'étude de la forme des mots (de leur flexion : indications de cas, genre, nombre, mode, temps, etc.de leur dérivation (proclitiques, préfixes, base, suffixes, enclitique) et de leur compositions). Sous l'appellation de morphosyntaxe, elle représente également l'étude des règles de combinaison des morphèmes[6] (unités minimales de sens) selon la configuration syntaxique de l'énoncé [7].

En pratique, dans le cadre du traitement automatique de la langue, l'analyse morphologique consiste à segmenter le texte en unité élémentaires (tokenisation) et de de vérifier l'appartenance d'un mot donné au domaine linguistique choisi (ou bien à la langue étudié) et à déterminer les différentes caractéristiques de ces unités.

4.2 Niveau syntaxique

Le niveau de traitement syntaxique permet d'associer à un énoncé (mot) sa ou ses structures syntaxiques possibles, en identifiant ses différents constituants et les rôles que ces derniers entretiennent entre eux. Cette phase reçoit au fur et à mesure de la phase 'Morphologie' les résultats de traitement des mots de la phrase indépendamment du contexte, commence à faire l'analyse du premier mot reçu de la phrase, et entre en communication avec les autres phases d'analyse, si nécessaire [7].

4.3 Niveau sémantique

L'analyse sémantique joue un rôle important dans l'étude du langage naturel, dans le sens où elle consiste à extraire la signification des structures de surface (l'étude du sens du mot hors contexte de la phrase ou du texte), et vise aussi à enlever les ambiguïtés qui restent après le traitement syntaxique et ainsi traiter les problèmes relatifs à la correspondance structure sens. [8].

4.4 Niveau pragmatique

Ce type de traitement permet de lever les ambiguïtés qui ne peuvent pas être éliminées par le traitement sémantique, à cause de certains problèmes ayant un lien avec le contexte dans lequel la phrase est prononcé (donner un sens au mot par rapport au contexte dans lequel il se trouve), c'est-à-dire, il se charge de placer le mot dans le contexte de l'ensemble des connaissances en faisant recours à des informations hors-contexte (géographie, sport, travail, …etc.) [4].

[6] Le morphème est la plus petite unité de première articulation au sein du mot (Exemple le mot nageur et composé de nag +eur) [9].

5. Les applications du TALN

Il est classique de présenter le domaine en l'organisant en grandes tâches, aux entrées/sorties bien identifiées : la traduction automatique, la production de résumés, la génération d'énoncés ou de textes, l'interrogation en langage naturel de bases de données, la synthèse de la parole à partir du texte sont ainsi quelques exemples de ces tâches que l'on appelle tâches finalisées. En schématisant grossièrement, ces tâches finales s'organisent en trois types principaux :

5.1 Les tâches de production ou d'aide à la production de documents

- Correction orthographique ou de syntaxe.
  - ✓ Intégrée à toute application informatique impliquant la rédaction
  - ✓ Correction basée sur des lexiques
  - ✓ Ex : traitement de texte, courrier électronique, navigateur Internet (zone de saisie)
- Génération de texte à partir d'une description formelle.
- Atelier d'aide à la rédaction.
- Correction grammaticale.
- Reconnaissance de caractères (OCR pour Optical Character Recognition en Anglais).
- l'apprentissage assisté par ordinateur des langues naturelles.

5.2 Les tâches liées à la gestion de documents ou de bases documentaires

- Traduction automatique (ou l'aide à la traduction automatique).
- Résumé.
- Recherche et extraction d'information.
- Le routage, classement ou l'indexation automatique de documents électroniques sont des variantes applicatives du paradigme de la recherche documentaire.
- La recherche de documents « intéressants » dans des bases documentaires.
- L'analyse d'un corpus de documents relatifs à un thème donné (histoire, veille technologique, etc.).

5.3 Les tâches liées à la conception d'interfaces homme-machine

- Agents dialoguant par téléphone.
- Assistants virtuels.
- Reconnaissance de la parole.
- Reconnaissance de la parole ou commande vocale (Reconnaissance vocale de Windows, Systèmes de navigation routière GPS, Smartphone…).
- Synthèse de la parole (Créer de la parole artificielle à partir d'un texte quelconque).

❖ Interfaces vocales (reconnaissance, synthèse, génération de dialogue, gestion du dialogue, accès aux bases de connaissance, etc.). [10]

6. Conclusion

Dans ce chapitre nous avons défini le TALN comme une discipline dont l'objet est la création de programmes informatiques capables de traiter de façon automatique les langues naturelles, et nous avons présenté les différents niveaux de traitement d'une langue naturelle et les domaines d'application existent dans cette discipline.

Dans le chapitre qui suit, nous allons centrer notre intérêt sur les propriétés morphologiques de la langue l'arabe et les différents outils de traitement automatique de la langue arabe.

CHAPITRE II

La langue arabe vs TALN

L’arabe se prêt à l’automatisation

1. Introduction :

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

Le développement de la langue arabe a été associé à la naissance et la diffusion de l'islam. L'arabe s'est imposée, depuis l'époque arabo-musulmane, comme langue religieuse mais plus encore comme langue de l'administration, de la culture et de la pensée, des dictionnaires, des traités des sciences et des techniques. Ce développement s'est accompagné d'une rapide et profonde évolution (en particulier dans la syntaxe et l'enrichissement lexical).

Dans ce chapitre, nous commencerons par présenter les caractéristiques de la langue arabe. Ensuite, nous décrirons le mécanisme de dérivation et la structure d'un mot arabe. Puis, nous présenterons les différentes catégories grammaticales. Enfin, nous donnerons un aperçu sur les problèmes, les outils et les ressources linguistique utilisé dans le domaine du traitement automatique de la langue arabe.

2. Caractéristiques de la langue arabe

L'arabe s'écrit et se lit semi-cursivement (écriture dont les lettres sont reliées les unes avec les autres) de droite à gauche, en utilisant un alphabet de 28 lettres. La plupart des mots arabes sont extrait à partir d'une racine de 3 caractères en ajoutant ou pénétrant des lettres ce qui engendre de nouveau mots en utilisant des schèmes. Ce qui le rend difficile à maitriser dans le domaine du traitement automatique de la langue [11].

2.1 Types de caractères

2.1.1 Les Consonnes

L'alphabet de la langue arabe comprend vingt-huit consonnes fondamentales, mais il y a des auteurs qui traitent la lettre alif « ا » comme la vingt-neuvième consonne. L'alif se comporte comme une voyelle longue qu'on ne trouve jamais en tant que consonne dans la racine [12].

Du point de vue linguistique l'alphabet arabe se divise en deux types de caractères:

a. Les caractères greffés

Ils ont nommés accessoire **«زوائد»**, parce qu'ils servent à former les différentes inflexions grammaticales des verbes et des noms, ainsi que les mots dérivent des racines (radicales). Les caractères greffes identifies: **« أ ، ب ، و ، ت ، س ، ن ، م ».**

b. Les autres caractères de l'alphabet

Forment l'ensemble des **caractères radicaux**. Ils ne servent à aucune fonction grammaticale, et constituent seulement des verbes racines. Il faut remarquer qu'un caractère greffe peut jouer le rôle d'un caractère radical alors qu'un radical ne peut jamais être greffe.

La représentation morphologique de l'arabe est assez complexe en raison de la variation morphologique et du phénomène d'agglutinement; les lettres changent de formes selon leur position dans le mot (isolée, initiale, médiane et finale). Le tableau 1 montre un exemple des différentes formes de la lettre « qaf » dans différentes positions [12]. Nous pouvons observer ainsi plusieurs caractéristiques générales de cette langue suivant le détail ci-après :

Isolée	Initiale	Médiane	Finale
ق	قـ	ـقـ	ـق
ق	قـرآن	الـقـرآن	غسـق

Table 1 : *Différentes écritures de la lettre « **qaf - ق** » selon sa positions dans le mot*

Toutes les consonnes se lient entre elles sauf « ذ , د , ز , ر , و » celles qui ne se joignent jamais à gauche. En plus, on peut trouver d'autres représentations qui sont le résultat de concaténation de deux consonnes par exemple, lorsque une « ل » lâm est suivie d'une « أ » hamza, les deux lettres sont remplacées par la ligature « لأ » [12].

2.1.2 Les voyelles

Un mot arabe s'écrit avec des consonnes et des voyelles. Les voyelles sont ajoutées au-dessus ou au-dessous des lettres. Elles sont nécessaires à la lecture et à la compréhension correcte d'un texte, elles permettent de différencier des mots ayant la même représentation. Les voyelles sont de deux types : les voyelles brèves et les voyelles longues [13].

a. Les voyelles brèves

Les voyelles brèves **(ـَ ، ـِ ، ـُ)** sont ajoutées au-dessus ou au-dessous des consonnes. Lorsque la consonne n'a aucune voyelle, on marquera une absence de voyelle représentée en arabe par une voyelle muette **(ـْ)**

Unité lexicale	1er interprétation		2ème interprétation		3ème interprétation	
كتب	كَتَبَ	Il a écrit	كُتِبَ	Il a été écrit	كُتُب	Livres
مدرسة	مَدرسَة	Ecole	مُدَرِسَة	Enseignante	مُدَرَسَة	Enseignée

Table 2 : *Ambiguïté causée par l'absence de voyelles pour les unités lexicales « **مدرسة** » et « **كتب** »*

Malgré l'importance de ces voyelles brèves, elles sont absentes dans la majorité des textes arabes ce qui peut engendrer des ambiguïtés de prononciation et de compréhension.

b. Les voyelles longues

Les voyelles longues sont des lettres prolongées, elles sont formées par une des voyelles brèves et une des lettres suivantes « ا، و، ي ».

Voyelle longue	Transcription
ـَـا	« â »
ـِـي	« î »
ـُـو	« û »

Table 3 : *Les voyelles longues*

2.2 Grammaire de la langue arabe

La grammaire traditionnelle se divise en deux branches [14] :

2.2.1 La morphologie (الصَرف)

Qui comprend :

a) Morphologie dérivationnelle

Elle étudie la construction des unités lexicales et leur transformation selon le sens voulu. Ainsi, la dérivation morphologique est décrite sur une base morphosémantique : d'une même racine, se dérivent différentes unités lexicales selon des schèmes qui sont des adjonctions et des manipulations de la racine [14].

كتب KTB (K : C1, T : C2, B : C3)			
Accompli	**Inaccompli**	**Nom d'agent**	**Nom de patient**
(C1 a C2 a C3 a)	(y a C1 C2 u C3 u)	(C1 a : C2 i C3 u n)	(m a C1 C2 u : C3 u n)
[kataba]	[yaktubu]	[ka:tibun]	[maktu:bun]
كَتَبَ	يَكْتُبُ	كـاتِبٌ	مَكْتُـوبٌ

Table 4 : *Exemples de dérivation de la racine « ktb كتب».*

b) Morphologie flexionnelle

Concerne le marquage casuel pour le nom et l'adjectif ou la conjugaison du verbe, appelé « الإعراب » [14].

2.2.2 La syntaxe (النحو)

Qui étudie la formation correcte des phrases garantit la grammaticalité de la phrase en analysent :

a) La position des unités lexicales les unes par rapport aux autres, déterminant ainsi l'ordre des unités lexicales.

b) Le marquage casuel des unités lexicales de la phrase. Ainsi, la fonction syntaxique de l'unité lexicale est déterminée en s'appuyant sur la morphophonologie.

3. Conception de la morphologie arabe

En arabe, la majorité des mots sont construits sur la base d'une racine tout en respectant un schème : ceci concernant notamment les verbes, les noms et quelques particules.

3.1 Notion de racine

Une famille de mots peut ainsi être générée d'un même concept sémantique à partir d'une seule racine à l'aide de différents schèmes. Ce phénomène est une caractéristique de la morphologie arabe. On dit donc que l'arabe est une langue à racines réelles à partir desquelles on déduit le lexique arabe selon des schèmes qui sont des adjonctions et des manipulations de la racine.

Une racine est purement consonantique, elle est formée par une suite de trois ou quatre consonnes formant la base du mot [15].

3.2 Notion de schème

Le schème est un mot composé de trois consonnes le 'FA', « **ف** », le 'AYN', « **ع** », et le 'LAM', « **ل** », qui sont vocalisées et qui peuvent être augmentées par d'autres lettres (préfixe, suffixe et infixe). Le schème joue un rôle très important dans le processus de génération des formes dérivées à partir d'une racine [16].

Figure 4 : *La représentation d'un schème*

Autrement dit le schème peut être considéré comme une moule sur laquelle coule la racine (voir Figure 5).

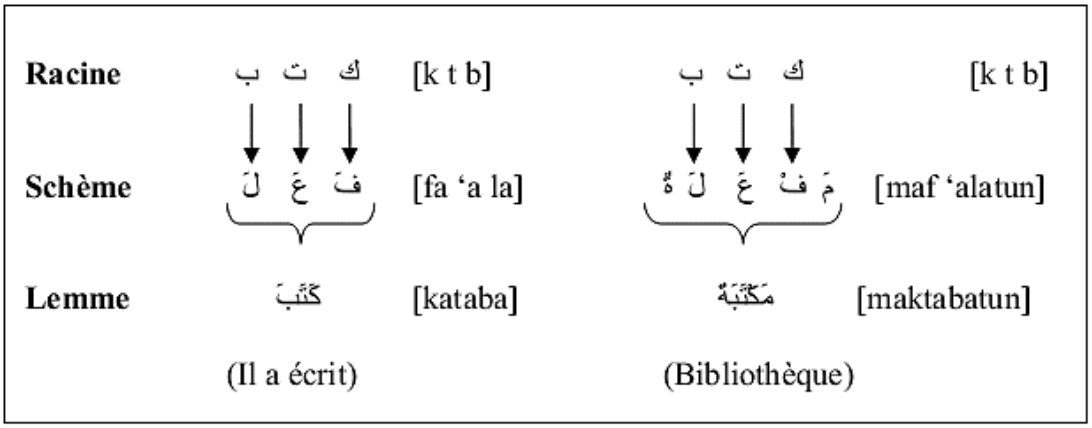

Figure 5 : *Exemple de dérivation de la racine « كتب » (écrire)*

3.3 Notion de lemme

Le lemme est l'entrée lexicale dans un lexique ou dans un dictionnaire. Il s'agit d'une forme entièrement vocalisée. Chaque mot est rapporté à son lemme qui est sa forme canonique qui dépend toujours de la catégorie grammaticale de ce mot : par exemple dans le tableau 5, si c'est un nom il doit être au singulier et si c'est un verbe il doit être à l'accompli avec la troisième personne du singulier etc. Un lemme peut être formé par un mot simple ou un mot composé [16].

Catégorie grammaticale	Lemme	Mot
Nom	كِتَابٌ	كُتُبٌ
Verbe	كَتَبَ	كَتَبْتُ
Particule	عَلَى	عَلَى

Table 5 : *Exemple de lemmes de catégories grammaticales différentes*

4. Structure d'un mot arabe

Le lexique arabe comprend trois catégories de mots : verbes, noms et particules. En arabe un mot peut signifier toute une phrase grâce à sa structure composée qui est une agglutination d'éléments de la grammaire, la représentation suivante schématisé une structure possible d'un mot (de droite vers la gauche).

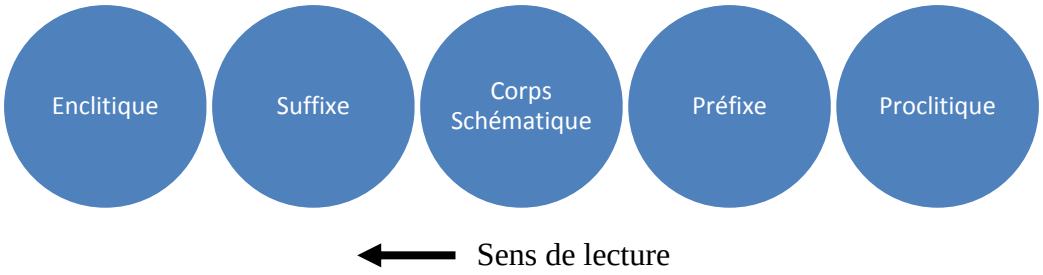

Figure 6 : *Structure du mot arabe*

4.1 Proclitiques

Au contraire des préfixes et des suffixes, les proclitiques se combinent entre eux pour donner plus d'informations sur le mot arabe (traits sémantiques, coordination, détermination...etc.). Comme le note [17] : « Dans le cas des verbes, les proclitiques dépendent exclusivement de l'aspect verbal. Ils prennent donc tous les pronoms et par conséquent ils sont compatibles avec tous les préfixes pris par l'aspect. Dans le cas des noms et des déverbaux, le proclitique dépend du mode et du cas de déclinaison. » Voici une liste non exhaustive des proclitiques simples :

Description	Proclitiques
La coordination par les coordonnants	« wa » وَ - فَ « fa »
L'interrogation par le morphème	« أ » « a »
La marque du futur	« Sa » « سَ »
L'article	« Al » « اَلْ »
Les prépositions par les lettres	« Li » لِ - « bi » بِ
Les particules des subjonctifs	« wa » وَ – « li » لِ – « fa » « فَ »
Le marqueur de comparaison par les lettres	« Ka » « كَ »
Le marqueur de corroboration	« La » « لَ »
La particule du jussif (الجزم) par la lettre	« Li » « لِ »

Table 6 : *Liste des proclitiques arabe.*

4.2 Préfixes

Habituellement représentés par une seule lettre, ils indiquent la personne de conjugaison des verbes au présent. Ils ne se combinent pas entre eux [17].

Voici la liste exhaustive de tous les suffixes:

Description	Préfixes
Indique la première personne au singulier (je)	أَ
Indique la première personne au pluriel (nous)	نَـ
Indique la deuxième personne féminine, masculine, singulière et duelle	تَـ
Indique la troisième personne masculine au singulier, duel, pluriel, masculin et féminin pluriel	يَـ

Table 7 : *Liste des préfixes arabe.*

4.3 Suffixes

Sont les terminaisons de conjugaison des verbes et de marques duelles/plurielles/singuliers pour les noms y compris les adverbaux. Ils ne se combinent pas entre eux. Le tableau suivant présente quelques suffixes an arabe avec leurs longueurs [17]:

Longueur	Suffixes
1	ـة ه ك ي ا ن و ـ
2	ته تي تك هو كو نه نك اه اك نو تا ات
3	تها تهم تهن تكم تون تنا وها وهم وهن وكن وكم وون
4	تهما وهما نهما اهما يهما ونهم ونهن ونكم ونكن وننو وننا تاها
5	ونهما تاهما اتهما ينهما ناهما تنهما انهما تموها تموهم تمونا تماها تماهم تماهمن

Table 8 : *Liste des suffixes arabe.*

4.4 Enclitiques

Comme les proclitiques, les enclitiques se combinent entre eux pour donner une post-base composée. Ils s'attachent toujours à la fin du mot pour produire des pronoms suffixes qui s'attachent au verbe(CD), au nom et au préposition (complément du nom) [17].

Voici dans le tableau suivant une liste des enclitiques:

Description	Enclitiques
1er Personne, Masculin/Féminin, Singulier	ي
1er Personne, Masculin/Féminin, Duel/Pluriel	نَا
3eme Personne, Féminin, Singulier	هَا
3eme Personne, Féminin, Pluriel	هُنَ

Table 9 : *Liste des exemples des enclitiques arabe.*

L'exemple dans la (Figure 7) montre bien la richesse morphologique de la langue arabe. Pour identifier les différentes formes soudées par ces phénomènes d'agglutination, et envisager un traitement automatique, il va donc falloir mettre en œuvre une phase spécifique de segmentation.

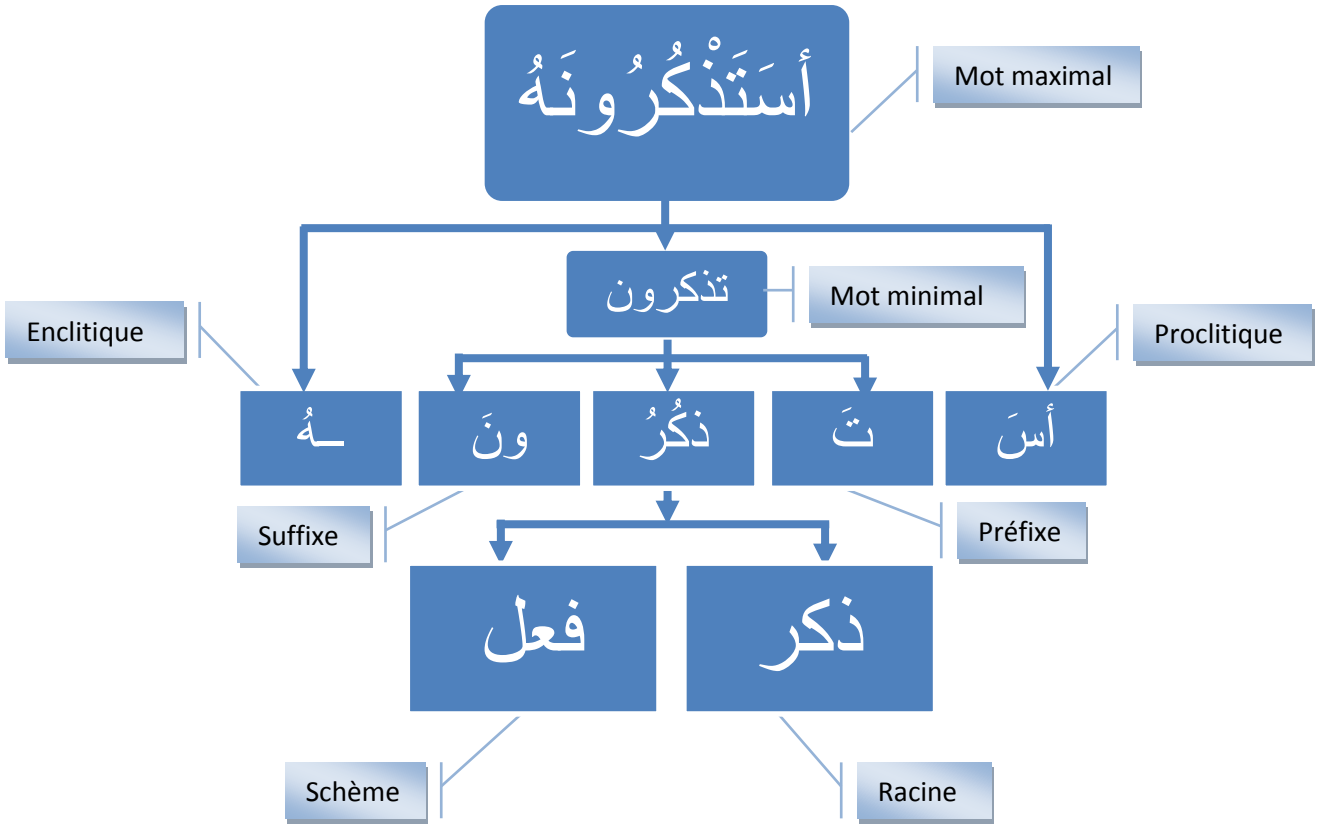

Figure 7 : *Segmentation du mot en arabe « أستذكرونه» (Est-ce que vous allez parler de lui)*

5. Les catégories grammaticales

Selon la théorie grammaticale arabe ancienne, le lexique de la langue arabe comprend trois catégories de mots : verbe, nom et particule [18].

5.1 Les verbes

Entité exprimant un sens dépendant du temps, c'est un élément Fondamental auquel se rattachent directement ou indirectement les divers mots qui constituent l'ensemble.

La plupart des mots en arabe, dérivent d'un verbe de trois lettres. Chaque verbe est donc la racine d'une famille de mots. Comme en français, le mot en arabe se déduit de la racine en rajoutant des suffixes ou des préfixes [19].

L'arabe possède un système complet de dérives verbaux, les formes augmentées pour exprimer l'intensité, le but, la réciprocité, etc. Ils sont créés par modification, par redoublement de la deuxième consonne, par adjonction et même par intercalation d'affixe.

La conjugaison des verbes dépend de plusieurs traits morphologiques :

- L'aspect (accompli, inaccompli).
- Le mode (indicatif, subjonctif, apocopé).
- La voix (actif, passif).

- ❖ Le nombre du verbe (singulier, duel, pluriel).
- ❖ Le genre du verbe (masculin, féminin).
- ❖ La personne (première, deuxième et troisième)

5.1.1 L'aspect (الصيغة)

La conjugaison du verbe arabe est réduite par rapport aux langues indo-européennes. La notion de temps n'y a point de position solide, mais il y a la notion d'aspect du verbe [20]. On en dénombre trois aspects :

a. L'accompli (الماضي)

Présente l'action passée et se distingue par des suffixes.

شَرِبَ	
شَرِبَ+ تُ = شَرِبْتُ	*j'ai bu*
شَرِبَ+ تْ = شَرِبَتْ	*elle a bu*
شَرِبَ+ نَا= شَرِبْنَا	*nous avons bu*

Table 10 : *L'aspect accompli*

b. L'inaccompli (المضارع)

Présente l'action en cours d'accomplissement, ses éléments sont des préfixes et/ou des suffixes.

شَرِبَ	
أَشْرَبُ = شَرِبَ + أ	*je bois*
يَشْرَب = شَرِبَ + يـ	*il boit*
تَشْرَبِينَ = ـين + شَرِبَ +تـ	*tu bois au féminin singulier*

Table 11 : *L'aspect inaccompli*

c. L'impératif (الأمر)

Indique l'ordre ou la demande. Il peut être conjugué seulement avec les deuxièmes personnes. Généralement, il faut ajouter un hamza au début du verbe et terminer celui-ci par la voyelle muette **(سكون ـْ)**.

Mode impératif	Lemme
إشربْ	شَرَبَ
أكتُبْ	كَتَبَ

Table 12 : *L'aspect impératif*

5.1.2 Le mode (الوضع)

On parle de mode quand il s'agit de l'inaccompli, mais l'accompli et l'impératif, chacun d'eux a une seule modalité. L'inaccompli a trois modes qui diffèrent par leurs désinences [20]:

a. L'indicatif (المرفوع)

Employé dans une proposition principale ou isolée. Il se caractérise par une désinence **(الضمة ـُـ) [dammat]** et par des flexions longues.

b. Le subjonctif (المنصوب)

Utilisé en proposition subordonnée. Il se caractérise par une désinence **(الفتحة ـَـ) [fathat]** et par des flexions courtes.

c. L'apocopé (المجزوم)

Employé dans le conditionnel. Il se caractérise par l'absence de désinence **(سكون ـْـ) [sucûn]** et par des flexions courtes.

5.1.3 La voix

a. l'actif (المعلوم)

La voix active et la voix où le sujet du verbe est connu. Le sujet est l'agent de l'action ; il est actif.

L'accompli connu	L'inaccompli connu
رَمَـى الكُرَةَ الصَبـيُ	يَرْمـي الكُرَةَ الصَبـيُ
Le garçon a jeté le ballon	**Le garçon jette le ballon**

Table 13 : *Exemple de la voix active*

b. le passif (المجهول)

La voix passive et la voix où le sujet subira l'action du verbe, action faite par un agent l'inconnu.

L'accompli de l'inconnu	L'inaccompli de l'inconnu
رُمـيَتْ الكُرَةُ	يُرْمَـى الكُرَةُ
Le ballon a été jeté	**Le ballon est jeté**

Table 14 : *Exemple de la voix passive*

5.1.4 La personne

Comme les autres langues, on en distingue trois :

Les personnes	Transcription
Première personne	[nahnu] نحن, [’anâ] أنـا
Deuxième personne	[’anta] أنتَ ، [’nti] أنتِ ، [’antumâ] أنتمـا ، [’ntum] أنتم ، [’atunna] أنتن
Troisième personne	[huwa] هو ، [hiya] هي ، [humâ]همـا، [hum] هم ، [hunna] هن

Table 15 : *La personne 1ère, 2ème et 3ème*

5.1.5 Le genre du verbe

Dans la langue arabe, il existe deux genres : masculin et féminin.

5.1.6 Le nombre du verbe

Un verbe arabe est pour vue de nombres à savoir singulier, pluriel, et duel [20].

Personne	Genre	Pluriel	Duel	Singulier
1° personne		fa3alnâ - فَعَلْنا		fa3altu - فَعَلْتُ
2° personne	masculin	fa3altum - فَعَلْتُمْ	fa3altumâ - فَعَلْتُما	fa3alta - فَعَلْتَ
	féminin	fa3altunna - فَعَلْتُنَّ		fa3alti - فَعَلْتِ
3° personne	masculin	fa3alû* - فَعَلُوا	fa3alâ - فَعَلَا	fa3ala - فَعَلَ
	féminin	fa3alna - فَعَلْنَ	fa3alatâ - فَعَلَتا	fa3alat - فَعَلَتْ

Table 16 : *Exemple de nombre de verbes*

5.2 Les noms

Les noms arabes regroupent les substantifs, les adjectifs et les pronoms, ainsi que d'autres noms invariables. La langue arabe est pourvue de nombres : singulier, pluriel, et duel. Les grammairiens distinguent deux sortes de pluriels : le pluriel externe ou sain et le pluriel interne ou brisé [20].

5.2.1 Le nombre d'un nom

a. Le féminin singulier

On ajoute le « ة », **exemple** : « **صغير** » *petit* devient « **صغيرة** » *petite.*

b. Le féminin pluriel

De la même manière, on rajoute pour le pluriel les deux lettres **« ات »**.

Exemple : « **صغير** » *petit* devient « **صغيرات** » *petites.*

c. Le masculin pluriel

Pour le pluriel masculin on rajoute les deux lettres « **ينَ** » ou « **ون** » dépendamment de la position du mot dans la phrase (sujet ou complément d'objet) [20].

Exemple : « **الراجع** »*revenant* devient « **الراجعون** » *revenants.*

d. Le Pluriel irrégulier

Il suit une diversité de règles complexes et dépend du nom.

Exemple : « **طفل** » *un enfant* devient « **أطفال** » *des enfants.*

Le phénomène du pluriel irrégulier dans l'arabe pose un défi à la morphologie, non seulement à cause de sa nature non concatenative, mais aussi parce que son analyse dépend fortement de la structure comme pour les verbes irréguliers [20].

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

e. Le duel

Sa désinence dépend de sa flexion casuelle et de la catégorie grammaticale. Pour le pluriel masculin on rajoute les deux lettres « يْنِ » ou « ان ».

Exemple : « **طفل** » *un enfant* devient « **طِفْلَيْنِ** » *deux enfants.*

5.2.2 La définition (التعريف)

C'est une information de type booléen qui peut prendre les deux valeurs suivantes [21]:

- **Oui**: si le nom est définissable par « **ال** » comme « **بَابُ** » « **البَابُ** ».
- **Non**: si le nom n'est pas définissable par « **ال** » comme « **بَابٌ** ».

5.2.3 Adjectif (الصفة)

Les adjectifs sont des mots qui décrivent où modifier une autre personne ou une chose dans la phrase. Les exemples suivants utilisent les adjectifs dans différentes façons et endroits afin de démontrer comment ils se comportent dans une phrase [21].

Règles de grammaire	Arabe	Adjectif
ma maison est blanche **[nom + adjectif]**	**بيتي أبيض** [bayti abyad]	**أبيض** - blanche abyad
votre pays est grand **[nom + adjectif]**	**بلدك كبير** [baladuka kabir]	**كبير** - grand kabir
les nouveaux livres sont chers **[pluriel + adjectif]**	**الكتب الجديدة غالية** [alkutub aljadida ghalia]	**غالية** - chers ghalia
cette langue est très facile **[adverbe + adjectif]**	**هذه اللغة سهلة جدا** [hadehe allughah sahlatun jidan]	**سهلة** - facile sahlatun

Table 17 : *Exemple de des adjectifs arabe*

5.2.4 Conditionnel (إسم الشرط)

Ces mots sont considérés comme des noms par les grammairiens arabes est qu'ont une influence sur les verbes, en les mettant à l'inaccompli 'madjzome'.

Exemple : من،ما،مهما،متى …etc.

5.2.5 Interrogatif (اسم استفهام)

Ces mots ce placent au début des phrases et peuvent être précédés d'une particule.

Exemple : لماذا،كيف،ماذا …etc.

5.2.6 Allusif (اسم الكناية)

Ces sont quelque noms qui sont employés pour éviter de citer des expressions.

Exemple : كذا،كم …etc.

5.2.7 Déclination (الإعراب)

En arabe, il existe trois cas de déclinaisons pour les noms. Leur dernière consonne porte généralement une voyelle brève qui varie selon leur fonction [21].

Fonction du mot dans la phrase	voyelle		mot déterminé		mot indéterminé
cas nominatif (sujet, attribut)	Damma	ـُ	البَيتُ - al-baytu	ـٌ	بَيتٌ - baytu^n
cas indirect (complément de nom, de préposition)	kasRa	ـِ	البَيتِ - al-bayti	ـٍ	بَيتٍ - bayti^n
cas direct (complément d'objet direct)	fatHa	ـَ	البَيتَ - al-bayta	ـاً	بَيتاً - bayta^n

Table 18 : *Exemple des trois cas de déclinations pour les noms*

5.2.8 Les pronoms personnels (الضمائر)

Les pronoms sont les suivantes: les pronoms personnels (ils se réfèrent à des personnes parlant, des personnes de qui on parle, ou les personnes ou les choses dont on parle), pronoms indéfinis, les pronoms relatifs (ils relient les parties de la phrase) et pronoms réciproques ou réflexive (dans lequel l'objet d'un verbe est influencé par l'objet du verbe) [21].

Pronom	Genre	Pluriel	Duel	Singulier
Pronom personnel isolé	masculin	أَنْتُمْ - 'antum	أَنْتُما - 'antumâ	أَنْتَ - 'anta
	féminin	أَنْتُنَّ - 'antunna		أَنْتِ - 'anti
Pronom personnel affixe	masculin	هُمْ - hum	كُما - kumâ	ه - hu
	féminin	هُنَّ - hunna	هُما - humâ	ها - hâ

Table 19 : *Exemple des pronoms personnels (isolé et affixe)*

5.2.9 Les démonstratifs (اسماء الإشارة)

Les pronoms démonstratifs, permettent de parler des choses sans les nommer, et en cela leur rôle est comparable à celui des pronoms personnels. Le pronom personnel est un sujet désignant une personne (je, tu, il, elle, …etc.), et de même le pronom démonstratif est un sujet désignant une chose (celui-ci, celle-là, …etc.).

En Arabe, ils sont plus nombreux qu'en français, car en Arabe on peut exprimer les choses plus clairement et avec plus de précision [21].

	Singulier	Pluriel	Duel
démonstratifs proches	هَذا - Celui-ci	هؤلاء - Ceux-ci, celles-ci	هذان - Ces-deux-ci
démonstratifs éloignés	ذاكَ - Celui-ci là-bas	أولئكَ - Ceux-ci là-bas, celles-ci là-bas	ذَانَكَ - Ces-deux-ci là-bas

Table 20 : *Exemple des pronoms démonstratifs (proches et éloignés)*

5.2.10 Les conjoints (الأسماء الموصولة)

Les conjoints désignent une ou plusieurs personnes, animaux ou choses (voir le tableau 21).

	Singulier	Duel	Pluriel
Masculin	الذي	اللّذان	الّذين، الألى،اللّاؤون،الألاءِ
féminin	التي،التِ،التْ	اللّتانِ	اللّواتي،اللّاتي،اللّوى

Table 21 : *Conjoints « الأسماء الموصولة »*

5.3 Les particules

Ce sont principalement les mots outils comme les conjonctions de coordination et de subordination. Les particules sont classées selon leur sémantique et leur fonction dans la phrase, on en distingue plusieurs types (Introduction, explication, conséquence, …). Elles servent à situer des faits ou des objets par rapport au temps ou au lieu [20].

Description	Exemples
Préposition	حتى ، عن
Particules de coordination	و ، ف ، ثم
Particules d'affirmation	نعم ، بلى ، أجل
Particules relatives	مـا
Particules d'interrogatives	هل ، أ
Particules de négation	لا ، لن ، لم
Particules distinctive	أيْ
Particules conditionnelles	إنْ ، لَوْ

Table 22 : *Liste des exemples des particules arabe.*

Ces particules seront très utiles pour notre traitement, elles font partie du dictionnaire qui regroupe les mots outils.

6. Traitement automatique de la langue arabe

6.1 Etat des lieux

Les recherches sur le traitement automatique de l'arabe ont débuté vers les années 1970, et les premiers travaux concernaient notamment les lexiques et la morphologie. Le domaine TALA[1] est devenu un centre de la recherche et du développement commercial dû à la nécessité essentielle de tels outils pour des personnes dans l'ère électronique. Mais, de l'autre côté de la réalité, peu d'outils de traitement automatique de la langue arabe sont mis à la disposition des utilisateurs arabophones, bien que les efforts soient en marche pour servir le nombre croissant d'utilisateurs.

6.2 Problèmes de traitement automatique de la langue arabe

Outre les phénomènes classiques comme l'ambiguïté, la coordination, la référence, l'anaphore et l'ellipse, phénomènes existants dans les langues latines (espagnol, français, italien, etc.), il y a d'autres problèmes spécifiques à la langue arabe et à certaines autres langues sémitiques, à savoir l'absence de voyelles, l'absence d'une ponctuation régulière et les problèmes de flexion et d'agglutination [22].

6.2.1 L'absence de voyelles

La plupart des documents arabes sont non voyellés. En effet, les voyelles ne sont utilisées que dans certains ouvrages scolaires pour débutants et dans le Coran.

Un texte arabe non voyellé est fortement ambigu. En effet, 74% des mots qui le composent acceptent potentiellement plus d'une voyellation lexicale et 89,9% des noms qui le constituent acceptent potentiellement plus d'une voyelle casuelle. La proportion des mots ambigus passe à plus de 90% si les comptages portent sur les voyellations globales (lexicales et casuelles) [23].

6.2.2 L'irrégularité de l'ordre des mots dans la phrase

L'ordre des mots en arabe est relativement libre. D'une manière générale, on met au début de la phrase le mot sur lequel on veut attirer l'attention et l'on termine sur le terme le plus long ou le plus riche en sens ou en sonorité. Cet ordre provoque des ambiguïtés syntaxiques artificielles dans la mesure où il faut prévoir dans la grammaire toutes les règles de combinaisons possibles d'inversion de l'ordre des mots dans la phrase [22].

Ainsi par exemple, on peut changer l'ordre des mots dans la phrase (1) pour obtenir des phrases ayant le même sens.

(1) ولد العالم و الباحث في الجزائر

(2) في الجزائر ولد العالم و الباحث

Figure 8 : *Exemple de problème d'irrégularité de l'ordre des mots dans une phrase*

[1] Traitement Automatique de la Langue Arabe

6.2.3 Problèmes de segmentation de textes

Pour traiter un texte, nous devons procéder à sa segmentation en paragraphes, phrases, propositions et mots. Cette segmentation est source d'ambiguïtés, vu que d'une part la ponctuation est rarement utilisée dans les textes arabes et d'autre part cette ponctuation, lorsqu'elle existe, n'est pas toujours déterminante pour guider la segmentation. De plus, certains mots outils peuvent marquer le début d'une nouvelle phrase (ou proposition), ce qui nécessite des analyses de surface afin de pouvoir segmenter le texte [22].

6.2.4 Problèmes d'agglutination

L'arabe montre une forte tendance à l'agglutination : l'ensemble des morphèmes collés les unes aux autres et constituant une unité lexicale véhiculent plusieurs informations morphosyntaxiques. Ces unités lexicales sont souvent traduisibles par l'équivalent d'une phrase en français. La structure d'une unité lexicale arabe est donc décomposable en cinq éléments : proclitique, préfixe, base, suffixe et enclitique.

La base est une combinaison de lettres radicales (le plus souvent trois) et d'un schème. La base – avec préfixe et suffixe - forme le noyau lexical, éventuellement entouré d'extensions [24]. Comme le montre l'exemple suivant : **(وَلِيَضرِبُهَا).** Les éléments clitiques sont séparés par le symbole "+" :

وَلِيَضرِبُهَا			
Wa +	**li +**	**ya + Dribu**	**+haA**
Et	**pour**	**frappent**	**elle**
« Et	**pour**	**la**	**frapper »**

Figure 9 : *Exemple d'agglutination dans le mot* « **وَلِيَضرِبُهَا** »

Cet exemple révèle la complexité morphologique de l'arabe. Il s'agit du verbe **يضرب** employé au présent du subjonctif, 3ème personne du masculin pluriel, la base verbale est « **ضَرَبَ** » et la racine « **ضرب** ». Le pronom sujet n'est pas réalisé. En position proclitique, on utilise la conjonction de coordination « **wa** » « **و** » et la conjonction « li » « **ل** ». En position enclitique, on utilise le pronom complément d'objet 3ième personne du féminin singulier « haA » « **هَا** » « elle ».

6.3 Outils de traitement automatique de la langue arabe

Concernant les outils, la demande de TAL provient, pour dire vite, de deux tendances « lourdes » :d'une part la nécessité de concevoir des interfaces de plus en plus ergonomiques, d'autre part la nécessité de pouvoir traiter (produire, lire, rechercher, classer, analyser, traduire) de manière de plus en plus « intelligente » les informations disponibles sous forme textuelle. Les outils des techniques de TAL sont donc nombreux et variées. Notre objectif dans cette section est de recenser les principaux outils de TAL en langue arabe.

6.3.1 Lemmatiseurs

Les lemmatiseurs se veulent d'abord un outil utile au TAL, ce type d'analyse « simpliste », traite de façon identique affixes flexionnels et dérivationnels. Les algorithmes de lemmatisation en arabe les plus connus sont ceux de [25] et [26]. Ci-dessous une description succincte de ces lemmatiseurs.

a. Khoja

Le lemmatiseur de Shereen khoja développé au sein de l'université de Lancaster, a été utilisé dans le cadre d'un système de recherche d'information développé à l'Université du Massachusetts. L'approche de Khoja consiste à détecter la racine d'une unité lexicale, d'une part, il faut connaître le schème par lequel elle a été dérivée et supprimer les éléments flexionnels (préfixes et suffixes) qui ont été ajoutés, d'autre part comparer la racine extraite avec une liste des racines préalablement conçue [26].

b. ISRI

Le lemmatiseur ISRI (The Information Science Research Institute) en Anglais partage de nombreuses caractéristiques avec le lemmatiseur khoja. Cependant, la principale différence est que l'ISRI n'utilise pas un dictionnaire de racine. En outre, si une racine n'est pas trouvée, l'ISRI retourne la forme normalisée, plutôt que retourne le mot d'origine non modifiée [27].

c. Al-Fedaghi

Al-Fedaghi et Al-Anzi proposent un algorithme tente de trouver la racine du mot en faisant correspondre le mot avec des schèmes différents avec tous les affixes possibles attachés à lui, et ne pas retirer les préfixes ou les suffixes [28].

d. Al-Shalabi

En ce contexte et avec d'autres systèmes de lemmatisation Al-Shalabi technique applique plusieurs algorithmes d'extraction des racines et des schèmes. Cet algorithme cherche la racine dans les cinq premières lettres du mot en enlevant le préfixe le plus long [29].

e. Larkey

L'approche de Larkey [25] est une analyse morphologique assouplie. Elle consiste à essayer de déceler les préfixes et les suffixes ajoutés à l'unité lexicale : par exemple le duel « **ان** » dans (**معلمان**, deux professeurs), le pluriel des noms masculins « **ين,ون** » dans (**معلمون**, des professeurs) et féminins « **ات** » dans (**مسلمات**, musulmanes).La forme possessive « **هم ,كم ,تا** » dans (**كتبهم**, ses livres) et les préfixes dans les articles définis (**فال,كال,بال, وال,ال**).

6.3.2 Analyseurs morphologiques

a. Aramorph

L'analyseur morphologique Aramorph [17] segmente les unités lexicales, repère les différents composants et atteste son appartenance à la langue. Par la suite, l'analyseur donne une liste des traits associés à l'unité lexicale en entrée. Il offre deux types d'options. Le premier vise les traits morphosyntaxiques, le second concerne l'analyse des préfixes et suffixes.

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

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

b. Xerox

L'analyseur morphologique de Xerox [30] est basé sur l'approche de transducteur à états finis. Ce transducteur découpe la chaîne d'entrée en une séquence d'unités lexicales qui peuvent correspondre à une forme fléchie, une marque de ponctuation, etc. La deuxième étape est l'analyse morphologique des unités lexicales produites par la segmentation de la phrase. Cette étape est aussi réalisée par un transducteur qui relie la forme fléchie à la forme lexicale (et vice-versa). La forme lexicale est une séquence comprenant la représentation canonique de l'unité lexicale (le lemme), un ensemble d'étiquettes représentant le comportement morphologique de l'unité lexicale, et sa catégorie syntaxique.

c. Sarf

Sarf[2] est un système intégré (moteur) qui peut générer des verbes arabes, les noms dérivés, à partir de leurs racines triples et quadruples, en fonction de la grammaire et des règles de la morphologie, et l'utilisation de la base de données du système.

d. ElixirFM

ElixirFM[3] est une mise en œuvre de haut niveau de la morphologie fonctionnelle arabe. Le noyau d'ElixirFM est écrit en Haskell, tandis que les interfaces en Perl supportent l'édition de lexique et d'autres interactions.

e. Stanford parser

La version originale de Stanford parser[4] a été principalement écrite par Dan Klein, avec un développement de grammaire linguistique par Christopher Manning. Les analyseurs probabilistes utilisent les connaissances de la langue pour essayer de produire l'analyse la plus probable de textes.

[2] Sarf : Arabic Morphology System, source :http://sourceforge.net/projects/sarf/ (Accédé le 03/05/2014)
[3] ElixirFM, source : http://sourceforge.net/apps/trac/elixir-fm/wiki (Accédé le 03/05/2014)
[4] Stanford parser , source : http://nlp.stanford.edu/software/lex-parser.shtml (Accédé le 03/05/2014)

6.3.3 Vocalisation

a. Mishkal

Ce projet vise à fournir des logiciels, en particulier dans la langue arabe pour l'utilisateur moyen et le développeur. Mishkal est un système de vocalisation automatique de texte arabe développé par Taha Zerrouki[5].

b. MADA

MADA+TOKAN[6] est un système de tokenization, vocalisation, étiquetage et la lemmatisation des textes arabe.

c. Sakhr

Sakhr[9] est un système commercial de vocalisation automatique pour la langue arabe. Malheureusement, le système est totalement fermé.

d. ArabDiac

ArabDiac[7] est un outil de vocalisation de texte arabe fourni par RDI, il est construit sur l'infrastructure de traitement automatique de langue naturel de RDI (Analyseurs morphologiques, étiqueteurs …etc.), comme le projet Sakhr le système est totalement fermé.

6.3.4 Traduction automatique

a. Google traduction

Google Traduction[8] est un service de traduction automatique fourni par Google. Il s'agit d'un service gratuit qui propose des traductions instantanées dans des dizaines de langues différentes y compris la langue arabe. Il peut traduire des mots, des phrases et des pages Web dans toutes les combinaisons de langues acceptées. L'objectif de Google Traduction est de rendre l'information utile et accessible pour tous, quelle que soit la langue dans laquelle elle est publiée.

b. Tarjim

Un site web de traduction automatique arabe-anglais et anglais-arabe, développé par la compagnie Sakhr[9].

c. Bing translator

Bing Translator[10] est une application gratuite fourni par Microsoft, qui propose un système de traduction très performant. C'est un service qui vous permet de traduire tous types de textes dans plus de 40 langues entre elles où que vous soyez et quand vous le souhaitez.

---

[5] Mishkal: Arabic Text Vocalization, source : http://sourceforge.net/projects/mishkal/ (Accédé le 02/06/2014)
[6] MADA, source : http://www1.cs.columbia.edu/~rambow/software-downloads/MADA_Distribution.html (Accédé le 03/06/2014)
[7] ArabDiac, lien : http://arabdiac.sakhr.com.eg/ (Accédé le 05/06/2014)
[8] Google traduction, lien : https://translate.google.com (Accédé le 05/06/2014)
[9] Tarjim, source : http://translate.sakhr.com/sakhr (Accédé le 05/06/2014)
[10] Bing translator, lien : http://www.bing.com/translator/ (Accédé le 05/06/2014)

6.3.5 Correction automatique

a. Ghalatawi

Le projet Ghalatawi [11] vise à développer une liste de mots faux orthographiquement et les corriger automatiquement, ainsi que des expressions régulières qui reflètent certains cas.

b. Duali

Duali[12], nommé d'après le légendaire fondateur de la grammaire arabe (Abul Aswad al Du'ali -. 688 d), est un correcteur orthographique pour la langue arabe.

c. Baghdad

Le correcteur orthographique arabe Baghdad a été développé sur la base des idées et des données de l'analyseur morphologique Buckwalter arabe.

6.3.6 Etiquetage

a. The Stanford tagger

La version originale de Stanford tagger[13] a été principalement écrite par Kristina Toutanova, l'outil lit le texte dans une langue et affecte les parties du discours de chaque mot (et autre jeton), comme nom, verbe, adjectif, etc.

b. Treetagger

Le TreeTagger [14] est un outil permettant l'étiquetage morphosyntaxique et la lemmatisation. Il a été développé par Helmut Schmid dans le cadre de projet TC dans l'ICLUS (Institut for Computational Linguistics of the University of Stuttgart) en Anglais. Cet outil est gratuit, disponible en ligne, et facile à installer sur les systèmes d'exploitations Linux ou Windows. Il a été utilisé avec succès pour étiqueter des nombreuses langues (arabe, anglais, français, allemand, italien, néerlandais, espagnol, bulgare, russe, grec, portugais, chinois, swahili …) .

c. ASVM

C'est un analyseur gratuit développé en perl par l'équipe de Mona Diab en 2004. Il s'agit d'une adaptation à l'arabe du système anglais « Yamcha » qui a été entrainé sur le corpus annoté Treebank, en utilisant le modèle Support Vector Machineet en se basant sur 24 étiquettes [31].

[11] Ghalatawi : Arabic AutoCorrect, source : http://ghalatawi.sourceforge.net/ (Accédé le 05/06/2014)
[12] Duali, source : http://projects.arabeyes.org/project.php?proj=duali_(Accédé le 05/06/2014)
[13] The Stanford tagger, source :http://nlp.stanford.edu/software/tagger.shtml (Accédé le 05/06/2014)
[14] Treetagger, source : http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/ (Accédé le 05/06/2014)

d. TAGGAR

C'est un analyseur morphosyntaxique spécialement développé pour la synthèse vocale arabe des textes voyellés. Il prend en considération l'ordre de traitement des mots pour minimiser les erreurs d'étiquetage. Le traitement se fait dans l'ordre suivant : analyse des mots outils et des mots spécifiques, analyses des formes verbales et enfin, analyse des formes nominales. Cet analyseur utilise 35 étiquettes grammaticales [32].

6.4 Ressources linguistiques

Les ressources linguistiques (RL) jouent un rôle essentiel dans les applications de la technologie des langues. Ainsi, d'une part elles alimentent les différents processus des systèmes de TAL, d'autre part, elles sont de plus en plus exploitées pour accompagner le travail de modélisation linguistique par des méthodes statistiques [33].

6.4.1 Corpus

Le corpus se définit de fait comme l'objet concret auquel s'applique le traitement, qu'il s'agisse d'une étude qualitative ou quantitative. Le corpus est défini par [35] comme « l'ensemble limité des éléments (énoncés) sur lesquels se base l'étude d'un phénomène linguistique ».

Le corpus de langue générale est consacré à une langue naturelle. Il tend à représenter la diversité des usages de la langue choisie. A ce titre, il est constitué d'un ensemble de données dont les conditions de production et de réception sont représentatives d'une grande variété de situations de communication (orale : monologue, interview, écrite : lettre, roman...), et de types textuels (exposé scientifique, fiction narrative, reportage...).

a. Khaleej 2004

Ce corpus est pour but de la réalisation des expériences sur des thèmes d'identification pour la langue arabe. Il a été extrait de milliers d'articles qui ont été téléchargés à partir d'un journal en ligne [36].

Le corpus contient plus de 5000 articles qui correspondent à près de 3 millions de mots.

Domaine de corpus	Nombre de documents
Economie	909
Articles Sportifs	1430
Internationales news	953
Locales news	2398
Totale	5690

Table 23 : *Composition du corpus Khaleej 2004*

b. Watan 2004

Watan-2004 corpus contient environ 20 000 articles qui parlent des six thèmes suivants "catégories": *Culture, religion, économie, locales news, international news et le sport*. Dans ce corpus, la ponctuation a été omise intentionnellement afin de le rendre utile pour la modélisation de langues [36].

Domaine de corpus	Nombre de documents
Economie	3468
Sport	4550
Internationales news	2035
Locales news	3596
Culture	2782
Religion	3860
Totale	20291

Table 24 : *Composition du corpus Watan 2004*

c. Tashkeela

Tashkeela est un corpus arabe vocalisé contient environ 6, 149,726 mots en forme HTML. Il a été extrait de la bibliothèque arabe Al-Shamela[15] en 2011 par Taha Zerrouki.

d. Quranic Arabic Corpus

Le corpus coranique arabe est une ressource linguistique annoté constitué de 77 430 mots. Le projet vise à fournir des annotations morphologiques et syntaxiques pour les chercheurs qui veulent étudier la langue du Coran.

e. TREC

La collection de TREC, une parmi ces ressources de large échelle connue et disponible pour les utilisateurs, elle représente un volume de 884 MOctets. Ce corpus est constitué de 383 872 documents. Il inclut des articles journalistiques provenant d'Arabic Newswire de l'AFP (Agence France Presse) du 13 mai 1994 au 20 décembre 2000 avec approximativement 76 millions d'unités lexicales [37].

Caractéristiques	TREC2001	TREC2002
Langue du corpus des documents	Arabe	Arabe
Nombre de documents	383 872	383 872
Nombre total de mots (tokens)	76 millions	76 millions
Nombre de mots différents	666 094	666 094
Taille moyenne des documents	150 mots	150 mots

Table 25 : *Caractéristiques de la collection TREC arabe (version 2001 et 2002)*

[15] Tashkeela: Arabic Vocalized text corpus, Source : http://sourceforge.net/projects/tashkeela/ (Accédé le 05/06/2014)

6.4.2 Dictionnaire

D'autres ressources telles que les dictionnaires monolingues et les dictionnaires bilingues sont nécessaires; ces types de ressources peuvent varier des dictionnaires de traduction automatique aux dictionnaires manuels pour un sujet ou une utilisation spécifique. Les dictionnaires Ajeeb et Ectaco sont accessible en ligne. De plus d'autres efforts ont été déployés dans différentes applications [38].

7. Conclusion

La langue arabe se caractérise par sa directionnalité droite à gauche, par sa nature semi-cursive (agglutination des mots), par ses signes de vocalisation qui s'ajoutent au-dessous et au-dessus des caractères, et par son ambiguïté due à l'absence de voyelles (cas de la majorité des textes arabes). Ces caractéristiques constituent en fait les problèmes majeurs face aux travaux effectués sur la langue arabe dans le domaine de TALA.

La langue arabe possède ses propres caractéristiques qui sont différentes par rapport aux langues indo-européennes. Elle se distingue par le lien étroit entre ses différents niveaux linguistiques : phonologique, morphologique, syntaxique et sémantique. Dans ce chapitre, nous avons étudié certaines caractéristiques de la langue arabe et nous avons ensuite présenté la classification traditionnelle tripartite (verbe, nom et particule). Finalement, Nous avons donné un aperçu sur les différents outils utilisés en TAL de la langue arabe et les différentes ressources linguistiques disponibles en arabe.

CHAPITRE III

Conception et Architecture de la boite à outils JEEM BOX

1. Introduction

Le traitement automatique de la langue arabe est devenu de plus en plus importante. Ce domaine de recherche actif connaît un grand progrès ces dernières décennies. A cette fin, nous nous sommes intéressés à l'acquisition des connaissances à partir d'un texte arabe.

Nous avons réalisé un système d'acquisition des connaissances dédié à la langue arabe fondé sur des outils qui combinent plusieurs techniques connues : lemmatisation, classification, translitération, fréquence des mots, vocalisation et étiquetage.

Dans ce chapitre nous allons présenter notre contribution, le but de notre travail est d'étudier les différentes méthodes d'acquisition des connaissances, d'appliquer quelques méthodes de lemmatisation, étiquetage ,translitération, classification, concordance, vocalisation, et d'élaborer les grands axes de réalisation de notre boite à outils JEEM Box à l'aide des techniques connues.

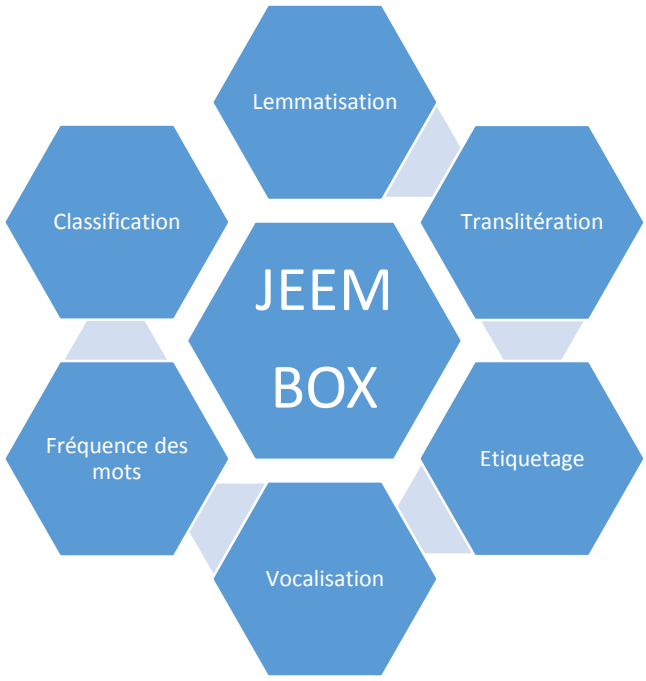

Figure 10 : *Architecture générale notre boite à outils JEEM Box*

2. Prétraitements

Afin de tenir compte de la conception de notre boite à outils JEEM Box et de pallier au problème de variation de représentation des caractères arabes dans les textes, il est nécessaire de définir et d'appliquer quelques prétraitements sur le texte avant l'extraction des informations et la réalisation des outils de l'application.

2.1 Encodage

L'écriture arabe ainsi que les écritures qui sont dérivées de l'arabe de base se caractérisent notamment par leur directionnalité de droite à gauche, par leur nature semi-cursive et par leurs signes de vocalisation qui s'ajoutent au-dessous et au-dessus des caractères. Ces trois caractéristiques constituent en fait les problèmes majeurs que rencontrent les technologies informatiques.

Pour remédier à ces problèmes, le standard Unicode offre tout un ensemble de codes de formatage et des algorithmes permettant, par conséquent, un traitement informatique fiable de l'écriture arabe et des écritures qui en dérivent.

2.1.1 L'Unicode

Unicode est une norme informatique développée par le Consortium Unicode[1] qui vise à donner à tout caractère de n'importe quel système d'écriture de langue un identifiant numérique unique, et ce de manière unifiée, quelle que soit la plate-forme informatique ou le logiciel [39].

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

2.1.2 UTF-8

Généralement en Unicode, un caractère prend 2 octets. Autrement dit, le moindre texte prend deux fois plus de place qu'en ASCII. C'est du gaspillage [39].

Un texte en UTF-8 est simple: il est partout en ASCII[2], et dès qu'on a besoin d'un caractère appartenant à l'Unicode, on utilise un caractère spécial signalant "attention, le caractère suivant est en Unicode".

[1]Le Consortium Unicode est une organisation privée sans but lucratif qui coordonne le développement du standard Unicode.
[2]Le Consortium Unicode est une organisation privée qui coordonne le développement du standard Unicode.

2.1.3 Caractéristiques importantes de l'UTF-8

- Conversion efficace à partir de ou vers un texte codé en UTF-16 ou en UTF- 32.
- Le premier octet indique le nombre d'octets, ceci permet une analyse rapide du texte vers l'avant.
- UTF-8 est un mécanisme de stockage relativement compact en termes d'octets.

L'UTF-8 rassemble le meilleur de deux mondes: l'efficacité de l'ASCII et l'étendue de l'Unicode. D'ailleurs l'UTF-8 a été adopté comme norme pour l'encodage des fichiers XML. La plupart des navigateurs récents supportent également l'UTF-8 et le détectent automatiquement dans les pages HTML. Ainsi, tout a été transformé en format Unicode dans notre application [39].

3. Conception et architecture générale de JEEM Box

Nous décrivons dans cette phase l'architecture générale de notre boite à outils JEEM Box dans le but de mettre on place les différentes conceptions de nos outils en détaillant les mécanismes et les techniques utiliser pour la réalisation de ce travail.

3.1 Architecture du lemmatiseur (JStem)

3.1.1 Principe

Lemmatisation automatique de texte arabe, ce système a pour principale objectif de générer à partir de chaque unité lexicale (verbale ou nominale) reconnu sa forme originale (lemme ou Racine), en passant par une série d'opérations qui rentre dans le domaine du Traitement automatique du langage naturel, comme : la segmentation (Texte, Phrase et mot) et la reconnaissance.

3.1.2 Les techniques de lemmatisation

Chaque langue naturelle a ses propres caractéristiques et dispositifs. Ainsi, il semble difficile de suivre la même configuration de lemmatisation et d'appliquer les mêmes techniques pour toutes les langues. Une technique de lemmatisation pourrait être pertinente à une langue, alors qu'elle ne peut effectivement l'être pour d'autres langues, et par conséquent elle ne peut être appliquée. Il existe plusieurs techniques utilisées pour la lemmatisation de mot. Celles-ci incluent des techniques de dictionnaires et d'analyse morphologique, de suppression des affixes et de statistiques.

a. Technique de dictionnaire

Cette technique basée principalement sur la construction d'un dictionnaire très grand en volume qui enregistre les mots trouvés en textes naturels avec leurs parties morphologiques correspondantes [26]. Ces parties incluent : racines, affixations. Plusieurs algorithmes ont été développés pour cette approche. Khoja contribuer avec un algorithme très important qui supprime les affixes, et vérifié pour chaque fois qu'il n'a pas retiré une partie de la racine. Enfin, trouver l'adéquation entre les modèles et le reste du mot pour en extraire la racine [36].

b. Technique de suppression d'affixe

La technique de suppression des affixes s'appelle généralement la lemmatisation assouplie ou légère « light stemming », quand elle est appliquée à la langue arabe, elle se réfère à un processus de suppression d'un petit ensemble de préfixes et de suffixes, sans essayer de traiter les infixes, ou d'identifier les schèmes (**أوزان**) et de trouver les racines. Cette approche conçue par suppression des chaînes de caractères fréquemment trouvées comme préfixes ou suffixes [11].

c. Technique d'analyse morphologique

La technique d'analyse morphologique est basée sur l'idée de la conformation du mot à un modèle (schème) pour trouver la racine du mot [11]. La racine est extraite après avoir retiré les affixes attachés à un mot donné.

3.1.3 La méthode proposée

Dans ce travail, nous avons proposé une méthode hybride qui incorpore trois techniques différentes. Les trois techniques sont: dictionnaires, suppression d'affixe et analyse morphologique.

Ces techniques ont besoin d'une certaine adaptation pour être pertinentes pour l'utilisation. Chaque technique est adaptée individuellement pour résoudre les problèmes pratiques liés à elle-même. Les sections suivantes décrivent en détails les techniques intégrées dans la méthode proposée.

a. Suppression d'affixe

Cette technique commence par la détermination de tous les affixes possibles (préfixes et suffixes) qui peuvent être attachés aux mots arabes. La liste des préfixes et des suffixes de la langue arabe étant limitée, on peut utiliser celle proposée par [17] pour la lemmatisation (voir annexe C).

Nous avons regroupé tous les affixes dans deux classes : préfixes et suffixes, basées sur les fréquences d'occurrence de ces affixes sur les mots différents de la collection arabe «Al-Khat Alakhdar». Dans ces deux

classes les préfixes et les suffixes sont organisés dans 3 catégories selon la longueur de l'affixe. D'autre part, nous avons donné une argumentation linguistique et statistique pour choisir les préfixes et les suffixes. Nous avons choisi les préfixes qui sont généralement des prépositions attachées aux débuts des mots, les suffixes qui sont des pronoms collés à la fin des mots.

Finalement, et pour la fiabilité, nous avons spécifié une valeur convenable pour la longueur du mot traité selon le schéma suivant (**S** : suffixe, **P** : préfixe) :

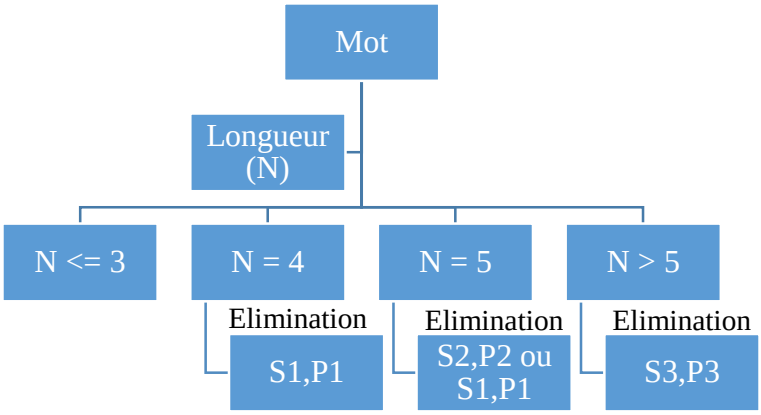

Figure 11 : *Elimination des affixes selon la longueur du mot*

Par exemple, le mot « سهم, flèche » pourrait être incorrectement lemmatisé en retirant les deux dernières lettres « هم » pour produire un lemme sans signification « س ». Dans un autre exemple, le mot « عين, œil », selon la méthode proposée aucune élimination ne serait effectuée. Dans ce cas-ci, la longueur du mot examiné est contrôlée et si elle satisfaite les conditions (supérieur ou égale à N), la lemmatisation serait appliquée.

b. Dictionnaire

Cette technique est adoptée pour atteindre deux objectifs de lemmatisation. Le premier est de résoudre la suppression incorrecte de quelques affixes, tandis que la seconde est de traiter le problème des mots arabisés « المعربة » et les mots spéciaux.

Pour lemmatiser un mot donné, ce dernier traverse une série d'étapes. Ces étapes sont récapitulées comme suit :

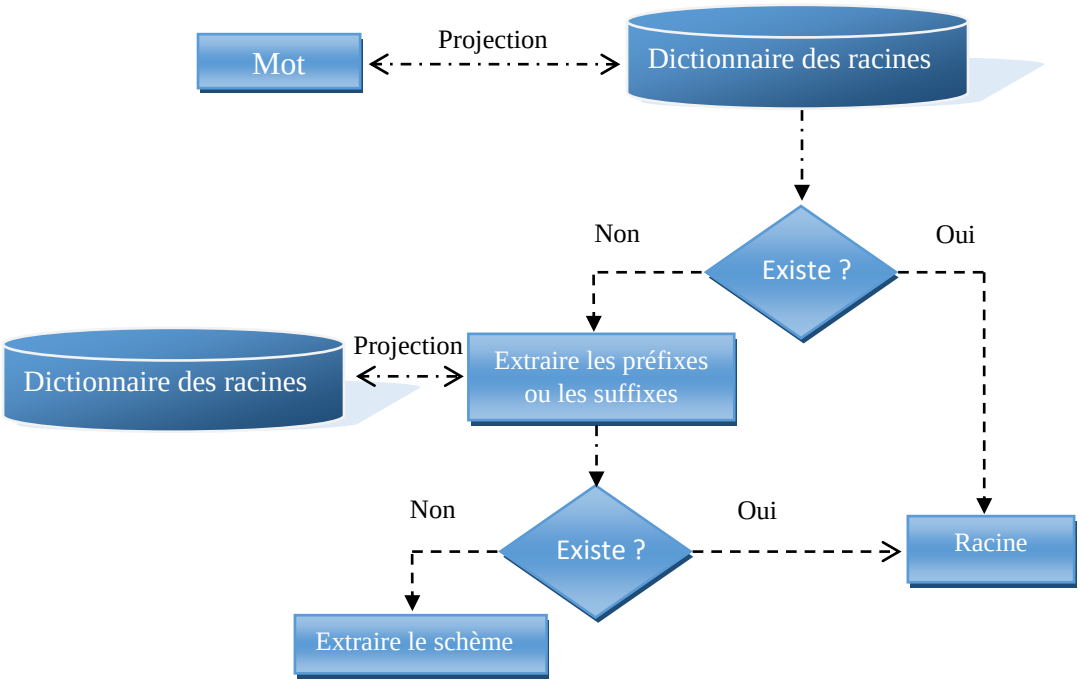

Figure 12 : *Lemmatisation du mot par la technique de dictionnaire*

Pour atteindre le deuxième objectif, on a construit un dictionnaire des mots spéciaux. La lemmatisation des mots spéciaux passera par la vérification du mot donné dans le dictionnaire de mots spéciaux, s'il n'existe pas, supprimer (un par un) les affixes possibles du mot, et vérifier à chaque fois le mot résultant dans le dictionnaire, s'il n'existe pas, alors le mot peut être erroné, autrement le mot est reconnu en tant qu'un mot arabisé ou spécial.

Dans certains cas la racine de trois caractères résultant contient des lettres faibles comme « و،ا،ي » au début de mot ou au centre ou à la fin du mot. Ces lettres doivent être remplacées par la lettre « و ». Pour cela les lettres faibles sont supprimés depuis la racine et les deux autre lettres sont projetées sur un dictionnaire qui contient des mots faibles sont la lettre faible, organise dans des listes selon la lettre faible. Par exemple le mot « يجد » contient une lettre faible « ي » et la racine correcte est « وجد ». Donc on supprime la lettre faible et on fait une projection pour les lettres « جد » sur la liste des mots faibles de la lettre « ي », si les lettres « جد » existent dans la liste donc on remplace la lettre faible « ي » par « و » sinon aucun remplacement ne serait effectué.

c. Technique d'analyse morphologique

La technique d'analyse morphologique est utilisée pour atteindre l'objectif, celui de diminuer le nombre de cas de la suppression incorrecte de quelques affixes (qui ont des lettres principales qui apparaissent en tant qu'affixes).

L'objectif peut être atteint en utilisant un ensemble de schèmes arabes pour améliorer le processus de lemmatisation. Ces schèmes sont organisés dans un dictionnaire selon la longueur du mot associé. On distingue trois (03) catégories principales selon le tableau 26 :

Longueur du mot	Longueur 4	Longueur 5		Longueur 6
Longueur de la racine	3 (PR4)	3 (PR53)	4(PR54)	3 (PR63)
Schèmes	مفعل ، فاعل فعال ، فعول فعيل، فعلة	افتعل، افاعل، مفعول، مفعال، مفعيل، مفعلة، تفعلة، افعلة، مفتعل، يفتعل، تفتعل مفاعل، تفاعل، فعولة، فعالة، انفعل، منفعل افعال، فعلان، تفعيل، فاعول، فواعل، فعائل، فاعلة، فعالي	تفعلل، افعلل ، مفعلل، فعللة، فعالل	مستفعل، استفعل، مفعالة ، افتعال، افعوعل تفاعيل

Table 26 : *un aperçu sur les schèmes de JStem*

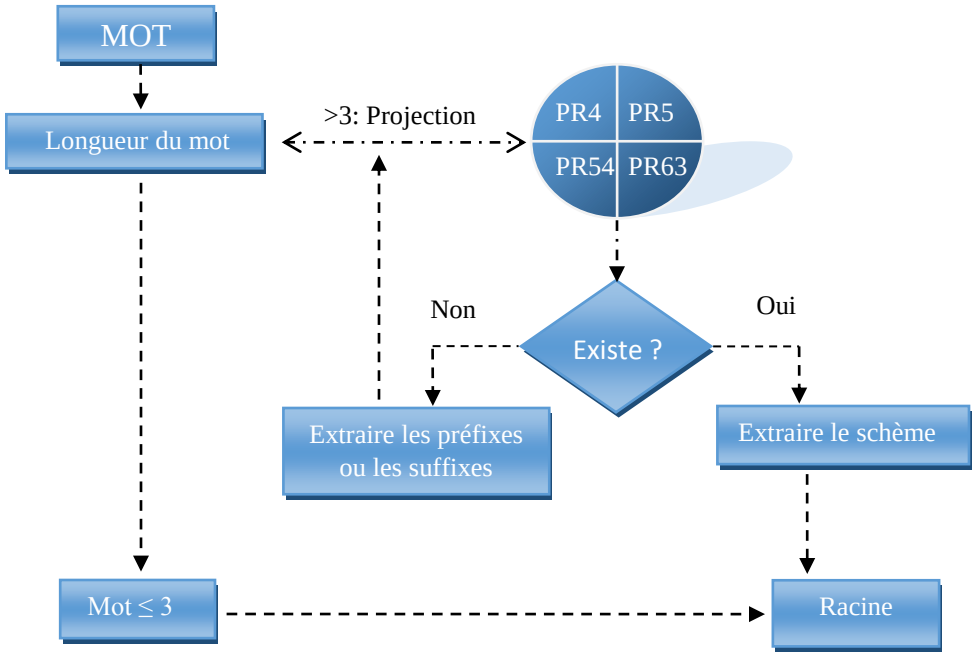

Figure 13 : *Lemmatisation du mot par la technique d'analyse morphologique*

3.1.4 Description de l'architecture générale du lemmatiseur JStem

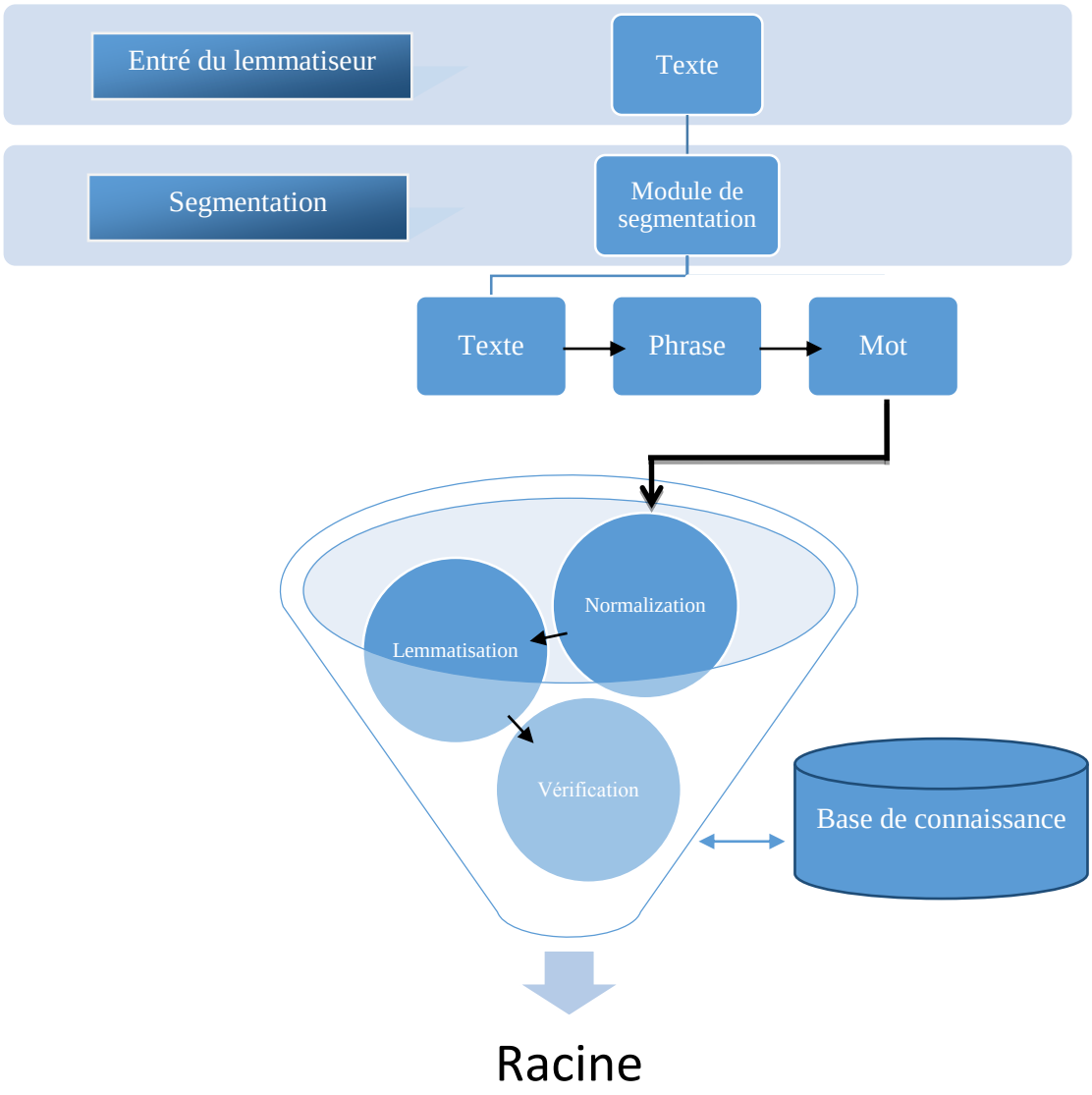

Figure 14 : *Représentation générale de l'architecture de JStem*

Nous avons envisagé de décomposer la réalisation de notre lemmatiseur en trois (03) parties complémentaires, la première partie est la construction d'une base de connaissance lexicale. Alors que la deuxième partie va être orientée vers la réalisation du module de segmentation, qui a pour rôle de fournir comme résultat des unités isolées. Et la troisième partie va inclure un module de reconnaissance et de lemmatisation.

3.1.4.1 Module de Base de connaissances lexicales de JStem

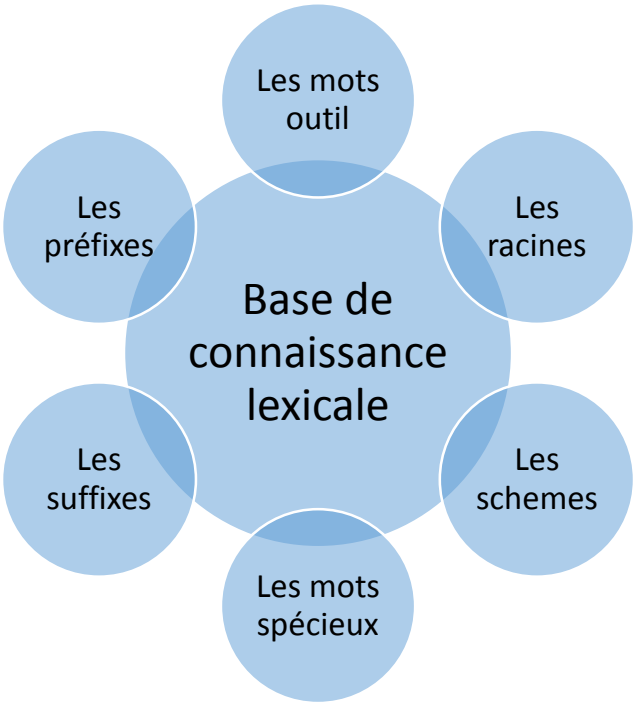

Figure 15 : *L'architecture générale de la base de connaissance de lemmatiseur*

a. Table des racines (الجذر)

Ces deux figures contiennent les racines trilitères et quadrilatères.

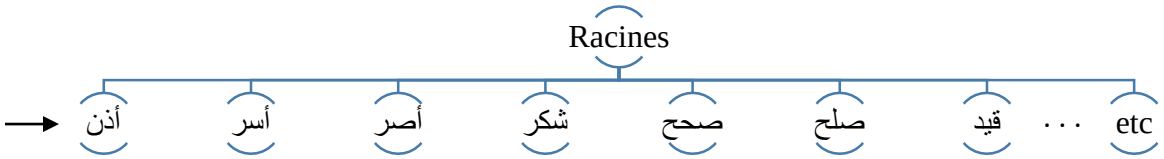

Figure 16 : *Exemple de racines trilitères*

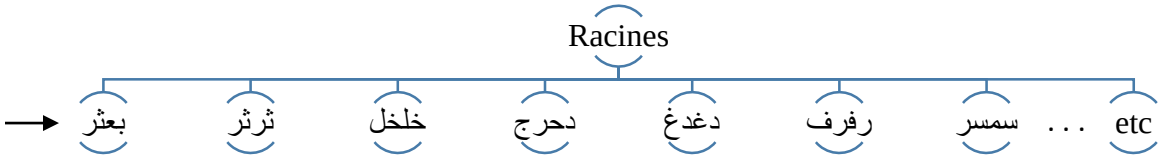

Figure 17 : *Exemple de racines quadrilitères*

b. Les mots spéciaux (الـكـلـمـات الـخـاصـة)

Comme les noms défectifs, c'est-à-dire les noms propres, nom de pays, de continent, de personne et nom commun, …etc.

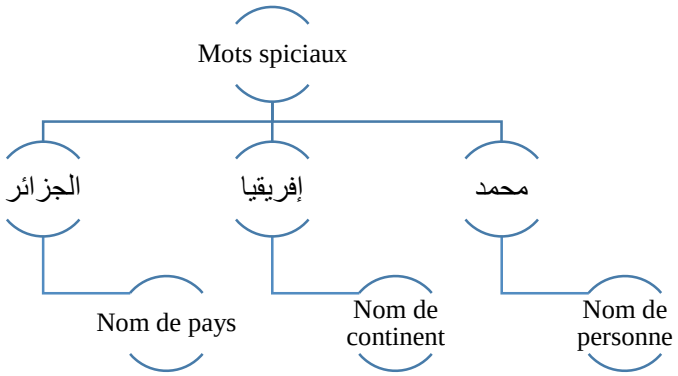

Figure 18 : *Exemple des mots spéciaux*

c. Table des mots outils (الـكـلـمـات الأدواتـيـة)

Les mots outils forment un ensemble de mots qui restent invariable quel que soit le contexte dans lequel ils sont utilisés.

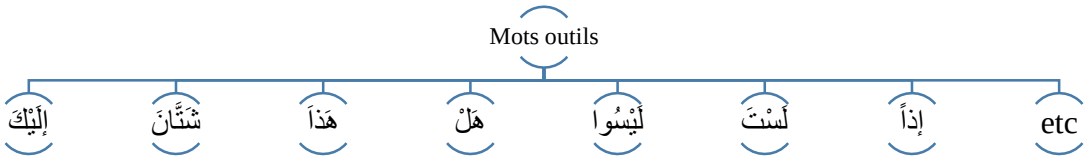

Figure 19 : *Exemple des mots outils*

d. Table des préfixes et suffixes (السوابق و اللواحق)

	Préfixes			Suffixes		
Longueur	1	2	3	1	2	3
Abréviation	P1	P2	P3	S1	S2	S3
	ل ب ف س و ي ت ن ا	ال لل	ولل وال كال بال	ة ه ي ك ت ا ن	ون ات ان ين تن كم هن نا يا ها تم كن ني وا ما هم	تمل همل تان تين كمل

Table 27 : *Table des préfixes et suffixes*

e. Table des schèmes (الأوزان)

Le schème représente une importance majeure dans notre système, il va nous permettre de détecter la racine.

Longueur du schème	Longueur 4	Longueur 5		Longueur 6
Longueur de la racine	3 (PR4)	3 (PR53)	4(PR54)	3 (PR63)
Schèmes	مفعل ، فاعل فعال ، فعول فعيل، فعلة	افتعل، افاعل، مفعول، مفعال، مفعيل، مفعلة، تفعلة، افعلة، مفتعل، يفتعل، تفتعل مفاعل، تفاعل، فعولة، فعالة، انفعل، منفعل افعال، فعلان، تفعيل، فاعول، فواعل، فعائل، فاعلة، فعالي	تفعلل، افعلل ، مفعلل، فعللة، فعالل	مستفعل، استفعل، مفعالة ، افتعال، افعوعل تفاعيل

Table 28 : *Table des schèmes*

PR4 : liste des schèmes de longueur 4 (extraction de racine de longueur 3)

PR53 : liste des schèmes de longueur 5 (extraction de racine de longueur 3)

PR54 : liste des schèmes de longueur 5 (extraction de racine de longueur 4)

PR63 : liste des schèmes de longueur 6 (extraction de racine de longueur 3)

On a utilisé le symbole (-) pour faciliter la procédure d'extraction du racine (projection) à partir d'un mot. Les racines du mot arabe sont basées sur le schème فعل

Schèmes	Formes
فاعل	ـا--
فعال	-ا-
فعول	-و-
استفعل	است---
افعوعل	ا- - و-ل

Table 29 : *Représentation des schèmes*

3.1.4.2 Module de segmentation

Pour notre application nous avons adopté les niveaux de segmentation suivants :

a. Niveau 1 : Une segmentation basée sur les signes de ponctuation majeur, tel que : «. », « ? », etc.

b. Niveau 2 : La segmentation au niveau de la phase peut être vue comme l'opération la plus simple dans le processus de segmentation globale, elle consiste à décomposer la phrase en segment en éliminant les blancs.

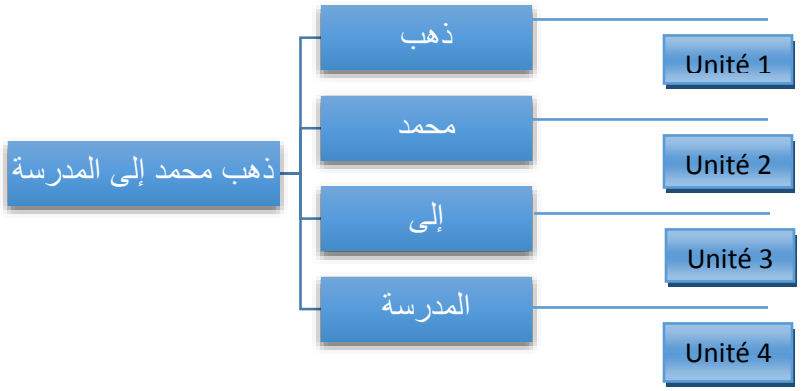

Figure 20 : *exemple de découpage de la phrase*

c. Niveau 3 : Décomposition au sein de mot (Proclitique + Préfixe + Base + Suffixe + Enclitique).

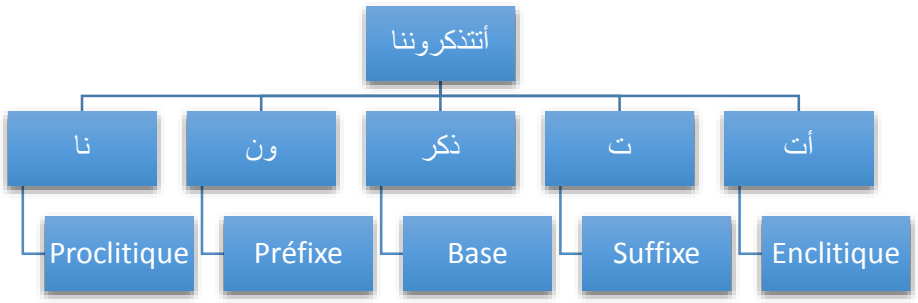

Figure 21 : *Segmentation du mot « أتتذكروننا »*

3.1.4.3 Module de reconnaissance et de lemmatisation

a. Phase de normalisation

Afin de manipuler les variations du texte qui peuvent être représentées en arabe, nous avons appliqué plusieurs genres de normalisation sur le texte. Dans notre démarche, la normalisation a concerné les étapes suivantes :

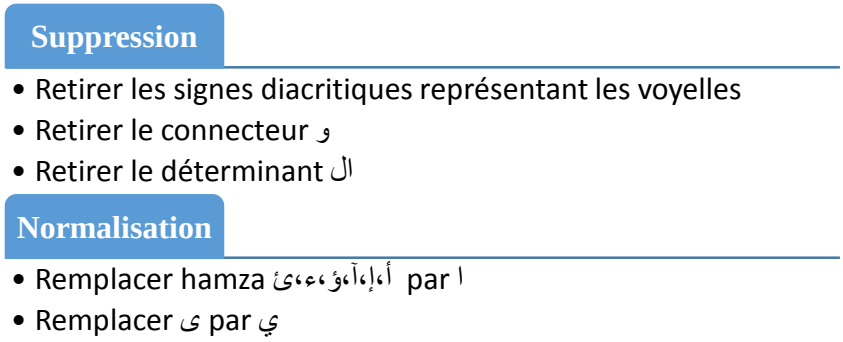

Figure 22 : *Processus de normalisation.*

b. Phase de lemmatisation et vérification

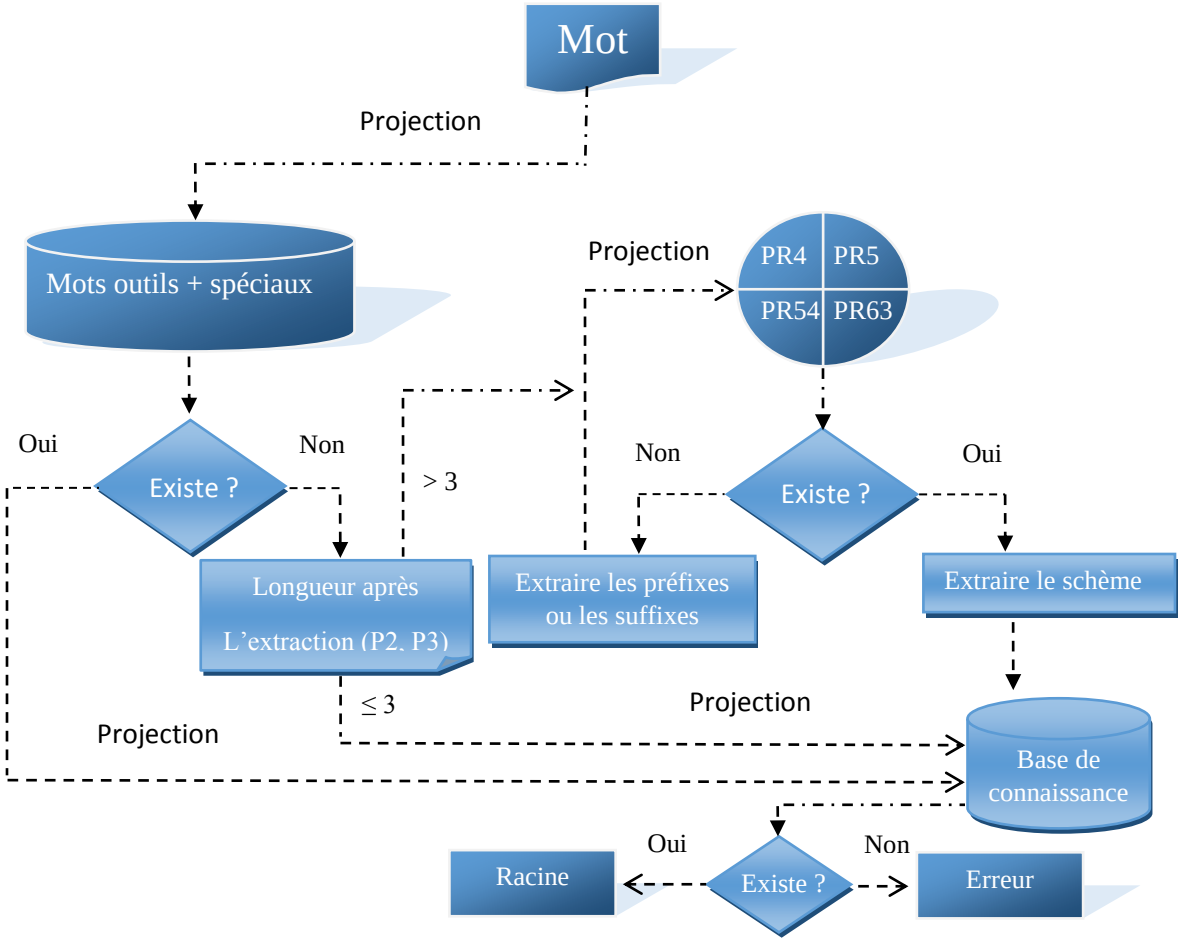

Figure 23 : *Processus de lemmatisation et de vérification*

3.2 Architecture du classificateur JClass

3.2.1 Principe

Le principe de cet outil est de mettre en place un système capable de recenser tous les mots qui se trouve dans un texte arabe et de les classer par racine. L'outil prend en entrée un texte et il fournit comme résultats une liste des classes des mots ayant la même racine.

3.2.2 Description de l'architecture générale du JClass

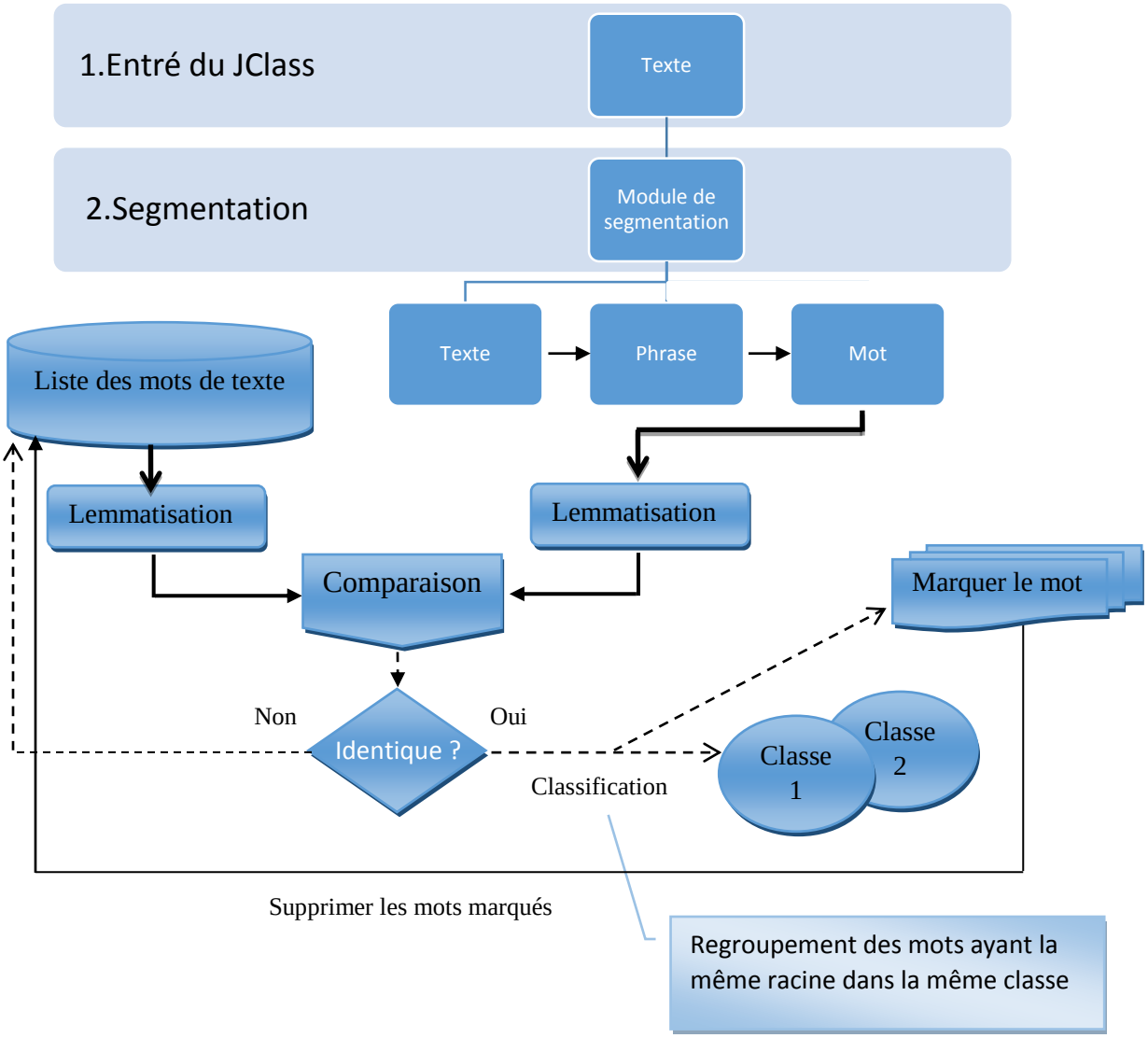

Figure 24 : *Représentation générale de l'architecture de JClass*

3.2.3 Description de la technique de recherche

La technique de recherche utilisée est la fenêtre glissante. Consiste à comparer le premier mot de texte au niveau de la racine avec les autre mots dans le texte un par un, s'il y'a des mots similaire on garde les mots et en marque le mot déjà analysé dans une liste pour éviter l'analyse du même mot plusieurs fois.

Exemple : On cherche les mots de la même famille de la racine « **عرب** » du mot « **العربية** » dans le texte suivant :

اللغة العربية هي أكثر اللغات تحدثا ضمن مجموعة اللغات السامية،و يتوزع متحدقوها في الوطن العربي.اللغة العربية ذات أهمية قصوى لدى العرب.

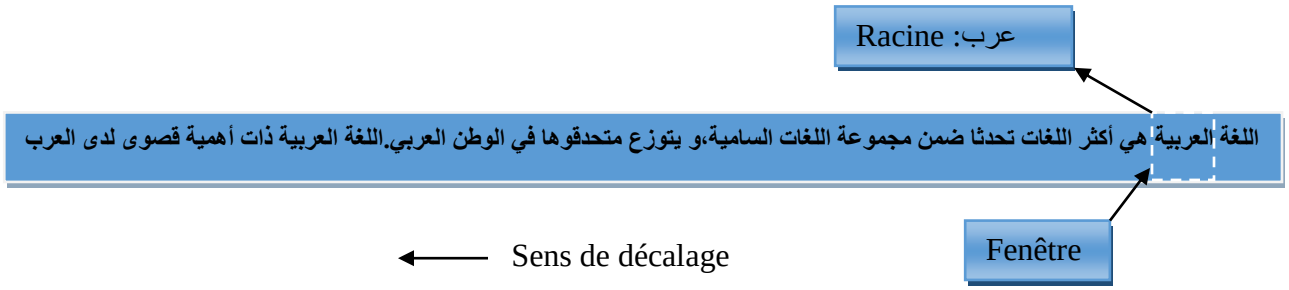

Figure 25 : *La comparaison avec la liste des mots du texte*

a. Comparaison

La comparaison se fait entre la racine du mot « **العربية** » et la racine du mot dans la fenêtre de la liste des mots du texte, si elles sont identique avec un pourcentage de plus 60% on marque le mot identique de la fenêtre dans la liste de la même famille que « **العربية** » et en passe au mot suivant dans la liste jusqu'à la fin.

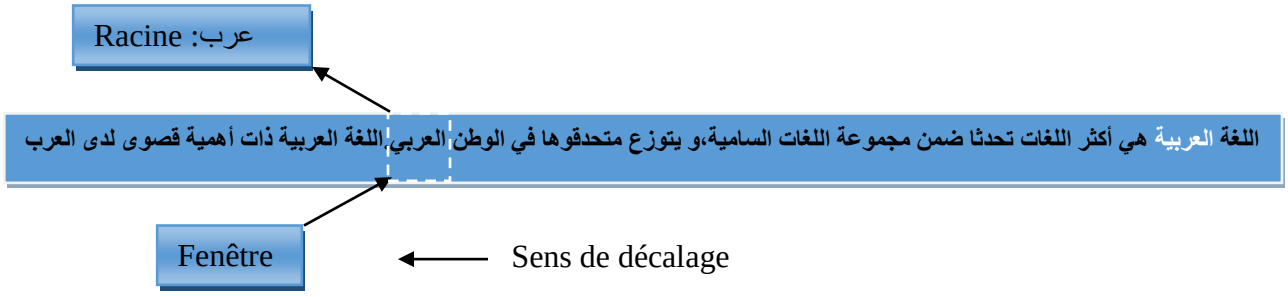

Figure 26 : *Exemple sur la recherche des mots ayant la même racine que « **العربية** »*

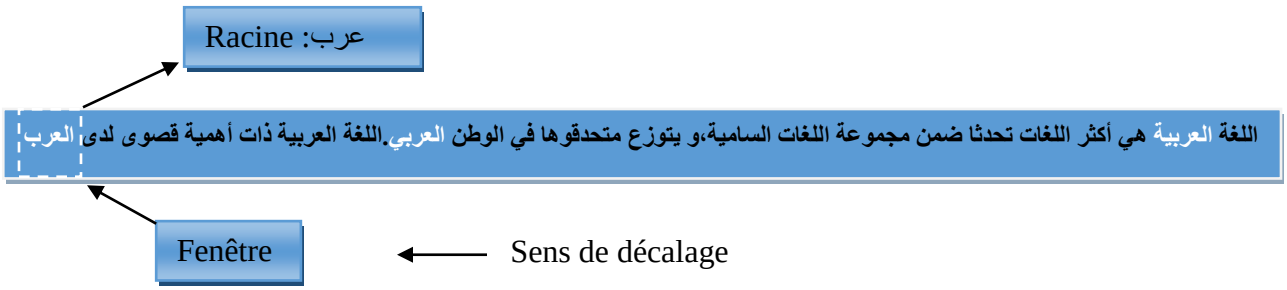

Figure 27 : La fin de la recherche dans le texte

b. Résultats

Classe du la racine « **عرب** » : **العرب،العربي،العربية**

3.2.4 Fonctions de Comparaison

a. Segmenter les deux racines à comparer en caractères et fait la somme de tous les caractères de deux mots (union).

b. Comparer les racines en utilisant la formule mathématique de *coefficient de Jaccard*

3.2.5 Description formelle de coefficient de Jaccard

L'indice de Jaccard (ou coefficient de Jaccard) est le rapport entre le cardinal (la taille) de l'intersection des ensembles considérés et le cardinal de l'union des ensembles. Il permet d'évaluer la similarité entre les ensembles. Soit deux ensembles A et B, l'indice est :

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \qquad (1)$$

Figure 28 : *Formule de coefficient de Jaccard*

3.3 Architecture du JTrans

3.3.1 Principe

JTrans consiste à représenter des caractères de l'alphabet arabe par des caractères latins. Il est utilisé pour le traitement informatique de données textuelles ou pour des indexations bibliographiques.

Par exemple, quand un utilisateur effectue une recherche, la translittération permet de retrouver l'information écrite dans un alphabet différent et de la retourner dans le système d'écriture de l'utilisateur.

Pharases	JTrans translittération
ذهب محمد إلى السوق	*hb mHmd <lY Alswq
ذَهَبَ مُحَمَّدٌ إِلَى السُّوقِ	*ahaba muHam~adN <laY Als~uwqi

Table 30 : *Exemple de l'opération de translitération depuis notre outil*

3.3.2 La translitération dans JTrans

La translitération Buckwalter du texte arabe a été développée à Xerox[3] par Tim Buckwalter[4] dans les années 1990. Il s'agit d'un ASCII système de translittération, représentant orthographe arabe strictement un-à-un, à la différence des systèmes de romanisation plus communs qui ajoutent des informations morphologiques ne sont pas exprimés dans la l'écriture arabe. Ainsi, par exemple, un « و » sera transcrit en « **w** » indépendamment du fait qu'il est réalisé comme une voyelle « وُ » ou une consonne « و ». Ce n'est que lorsque « و » est modifié par un Hamza « وٕ » sera transcrit en « **&** ».

Version utilisé dans notre outil est la Translitération Buckwalter avancée :

14 symboles coraniques ne figurent pas dans le projet initial (voir annexe A). Dans le schéma étendu utilisé par JTrans, ceux-ci ont été affectés à des signes de ponctuation ASCII. Ce n'est pas ambigu, car la ponctuation moderne ne se produit pas dans le Coran.

[3] Beesley, Kenneth. Romanization, Transcription and Transliteration.
Source : http://www.xrce.xerox.com/competencies/contentanalysis/arabic/info/romanization.html
[4] Buckwalter, Timothy. Arabic Transliteration, source : http://www.qamus.org/transliteration.htm

3.4 Architecture du concordancier JConcord

3.4.1 Principe

JConcord prend en entrée un texte. Puis scanne les entées mot par mot, pour les compter, les analyser et les classer. JConcord renvoi une nouvelle liste de texte organisé par fréquence et par ordre alphabétique ou fréquentiel.

3.4.2 Description de l'architecture générale du JConcord

JConcord prend en entrée un texte et il permet :

- La construction d'une concordance
- La construction de listes de fréquences d'items par ordre alphabétique ou par ordre fréquentiel.

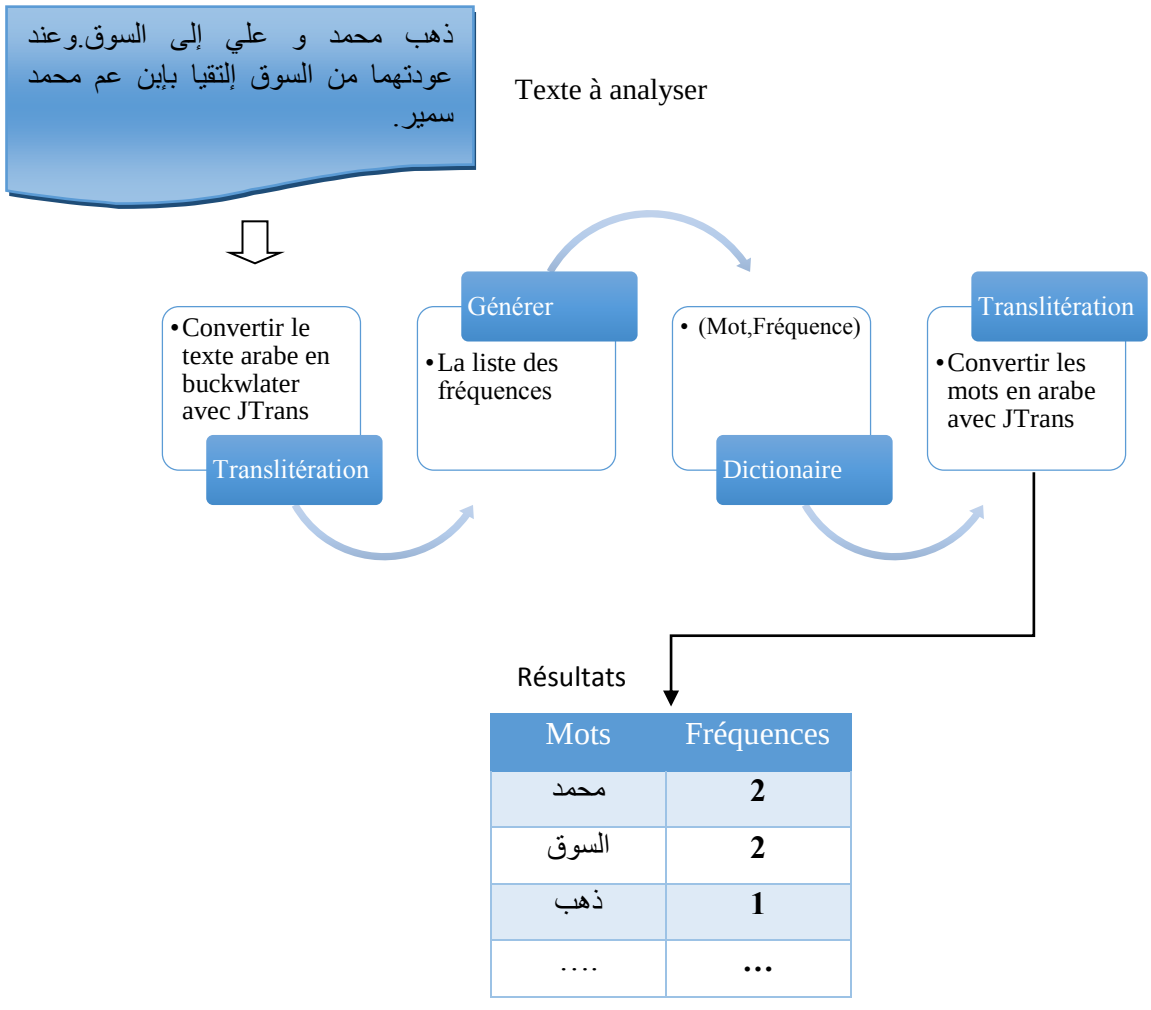

Figure 29 : *Représentation générale de l'architecture de Jconcord*

3.5 Architecture du système de vocalisation JDiac

3.5.1 Principe

Le principe de JDiac est d'étudier le système de vocalisation dans le texte arabe et de construire un système qui serait en mesure de vocalise le texte arabe automatiquement. Il prend en entrée un texte arabe vocalisé ou non, et il produit un texte vocalisé complètement ou partiellement selon le choix des options de vocalisation.

6.5.2 Techniques de vocalisation

a. Concordance

Cette technique utilise un dictionnaire de fréquence (Modèle) générer par un outil de concordance depuis un corpus vocalisé. L'opération s'effectue pour un seul mot à la fois, le mot prend la forme du mot vocalisé similaire dans le dictionnaire, à condition que la fréquence du mot similaire soit la plus élevée fréquence [40].

Fréquence	Mot كتب
18	كَتَبَ
7	كُتِبَ
Résultats	كَتَبَ

Table 31 : *Exemple affectation d'une forme pour le mot « كتب »*

b. Analyse syntaxique

Cette technique utilise un analyseur syntaxique pour reconnaitre le rôle du mot dans le texte, et pour affecter une propre forme de vocalisation au mot [40] :

Forme	Rôle	Mot
عَلِمَ	فعل ماض	علم
عُلِمَ	فعل مبني للمجهول	علم

Table 32 : *Exemple affectation d'une forme pour le mot « علم »*

c. Model de Markov caché (HMM hidden markov model en Anglais)

Utilisation du modèle de Markov caché qui se base sur l'emplacement du mot dans une phrase et générer un tableau des mots qui a suivi et qui l'a précédé dans la phase d'apprentissage. Pour affecter une forme au mot désiré le système cherche la forme dans le texte d'apprentissage par le mot précédant et suivant ignorant la relation directe avec le mot a formé [40].

3.5.3 Description de l'architecture générale du JDiac

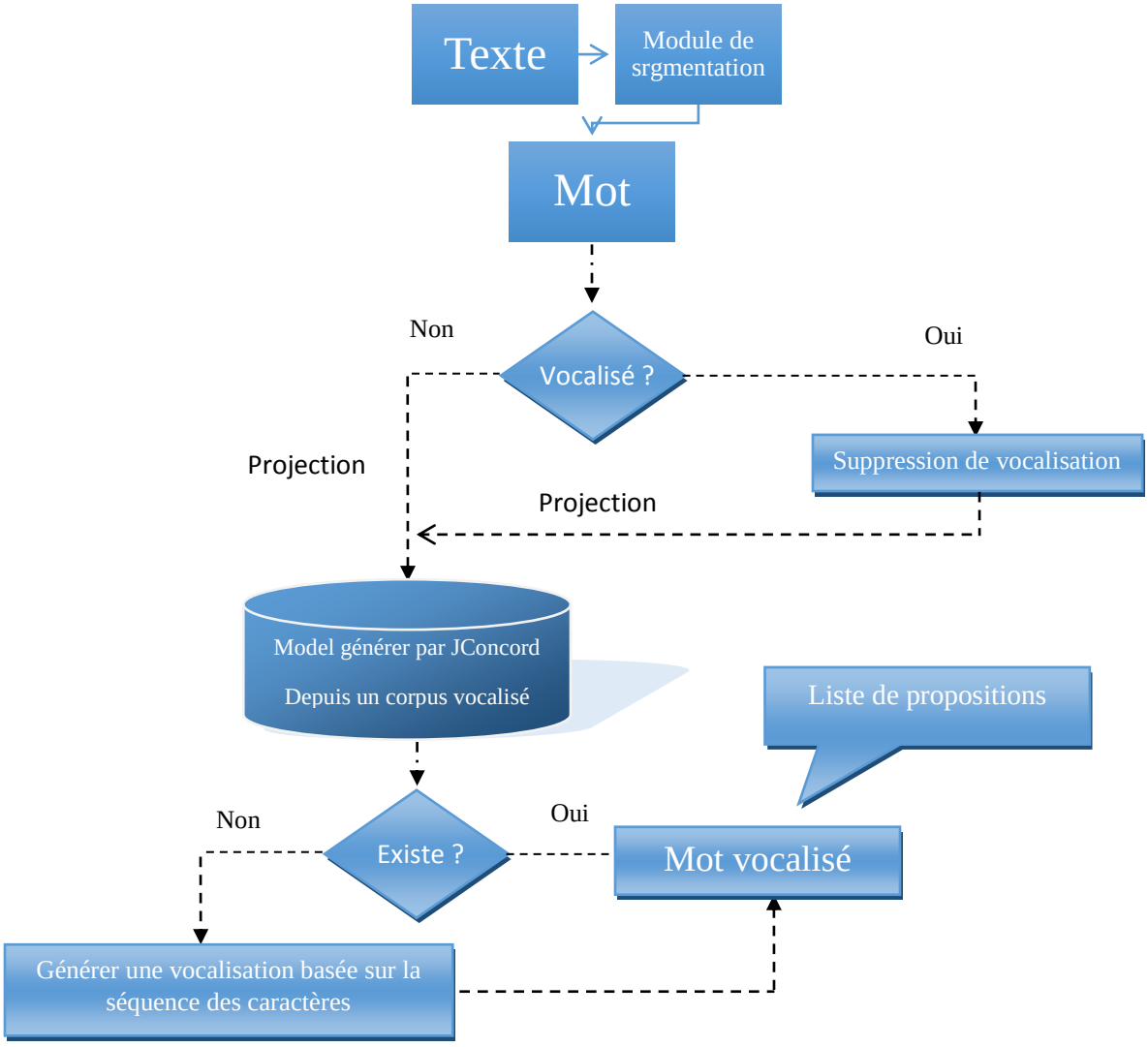

Figure 30 : *Représentation générale de l'architecture de JDiac*

JDiac se base sur 2 modules pour vocaliser les mots d'un texte arabe, l'un est basé sur le modèle de concordance d'un corpus vocalisé et l'autre sur la séquence des caractères d'un mot appliquant des règles de vocalisation de la langue arabe.

3.5.4 Les modules

3.5.4.1 Vocalisation à base dictionnaire ou modèle

Le premier module est généré par notre outil de concordance JConcord appliqué sur un corpus vocalisé qui s'appelle Tashkeela extrait de la bibliothèque arabe Al-Shamela en 2011 par Taha Zerrouki. Le texte du corpus doit être converti en buckwlater par l'outil JTrans afin de réaliser le modèle de vocalisation.

Le modèle de vocalisation est composé de 3 fichiers générer par l'outil JConcord avec l'extension '*.model' :

- Model de mots vocalisés
- Model de mots non vocalisé
- Model de fréquence

a. Représentation des modèles

Les modèles sont représentés avec des vecteurs, un vecteur pour chaque modèle avec un indice. Les vecteurs sont de même taille.

b. La recherche dans les modèles

La recherche de la forme de vocalisation similaire d'un mot se fait par la projection du mot sur le vecteur du modèle non vocalisé. On vérifie que le mot existe dans le modèle non vocalisé. Si le mot existe on vérifie que ce dernier a la plus haute fréquence dans le modèle de fréquence gardant leur indice pour récupérer la forme de vocalisation depuis le vecteur de mots vocalisés.

Remarque : les modèles sont ordonnés par fréquence de la plus élevée à basse fréquence.

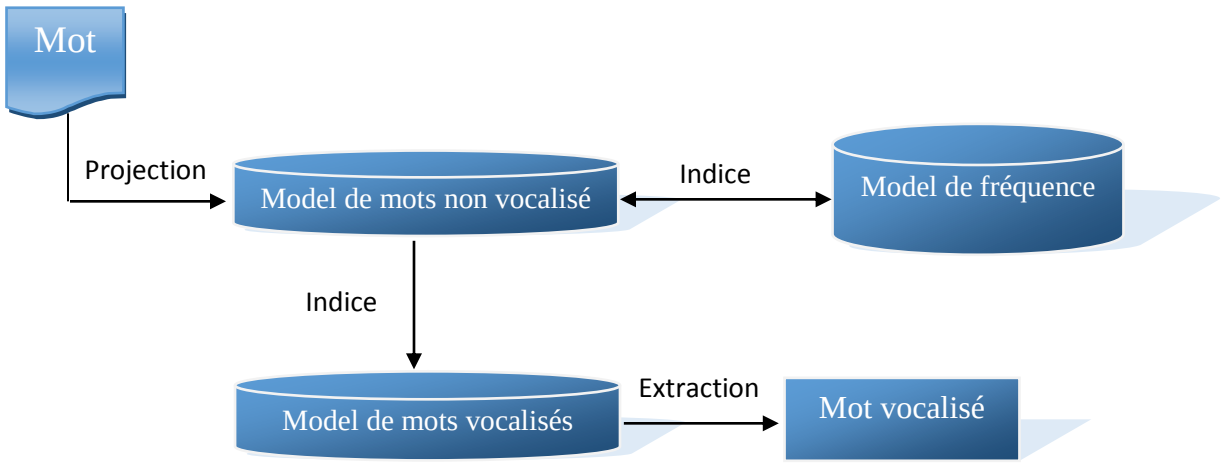

Figure 31 : *Processus du premier module de vocalisation à base model*

3.5.4.2 Vocalisation basé sur la séquence des caractères

Ce module a pour but de vocalisé les mots qui sont absent dans le model de vocalisation (dictionnaire) et les mots arabisé comme par exemple « **كمبيوتر** ». Basant sur les règles de vocalisation suivantes :

a. Caractère suivi de la lettre de prolongation « **حرف المَدْ** » prend la même voyelle que la lettre.

Lettres de prolongation	Exemples
ا	البَـاب la porte
و	يُـوسف Yussuf
ي	شـرِيف Noble

Table 33 : *La lettre de prolongation et la vocalisation des mots*

b. Lorsque la hamza « ء » se trouve au début du mot elle prend toujours pour support la lettre Alif « ا »et prend la voyelle de sa position.

Lettres de prolongation	Exemples
أ	أَخْضَرْ Vert
إ	إبِـلْ Chameaux

Table 34 : *La lettre hamza au début de mot*

c. Caractère avant « **تاء التأنيث** » reçois toujours la voyelle Fatha (َ) comme l'exemple suivant :

Mot	Vocalisation
غزوة	غَزْوَةٌ
تفاحة	سِجَادَةٌ

Table 35 : *Vocalisation de la lettre avant « **تاء التأنيث** »*

d. Lorsque la hamza « ء » se trouve à la fin du mot, la lettre avant la hamza prend la voyelle de sa position.

e. La première lettre du mot prend la voyelle Fatha (َ).

f. La vocalisation des lettres de définition « ال » selon la lettre lunaire, le Lam « ل » prend la voyelle Sokoun (ْ) s'il est suivi par une lettre lunaire « ا،ب،ج،ح،خ،ع،غ،ف،ق،ك،م،ه،و،ي ».

Lettre lunaire	exemple
أ	الْأَلِف
ب	الْبَرْق
ج	الْجَمَل
غ	الْغُلَام
ف	الْفَرَس
ق	الْقُوَّة
ك	الْكَلِمَة
م	الْمَوْت
ه	الْهُدَاة
ي	الْيَوْم

Table 36 : *Exemples de vocalisation du Lam suivi par une lettre lunaire*

g. Si la lettre après les lettres de définition « ال » est une lettre solaire on ajoute la voyelle Chadda (ّ) à elle. Les lettres solaires sont : « ت،ث،د،ذ،ر،ز،س،ش،ص،ض،ط،ظ،ل،ن »

Lettre lunaire	exemple
ت	التَّاجِر
ث	الثَّائِر
د	الدِّينَار
ذ	الذَّكَر
ر	الرَّحْمَة
ز	الزَّجَاجَة
س	السَّمَاء
ش	الشَّمْس
ط	الطَّبِيب
ن	النُّور

Table 37 : *Exemples de vocalisation des lettres solaires après «ال»*

3.6 Architecture du système JExtract

3.6.1 Principe

JExtract est un outil permettant la classification des mots en se basant sur leurs informations morphosyntaxiques (catégorie grammaticale, le genre, le nombre, le temps) [41]. Le principe de cet outil est le fait de classer les mots d'un texte selon les paramètres de recherche.

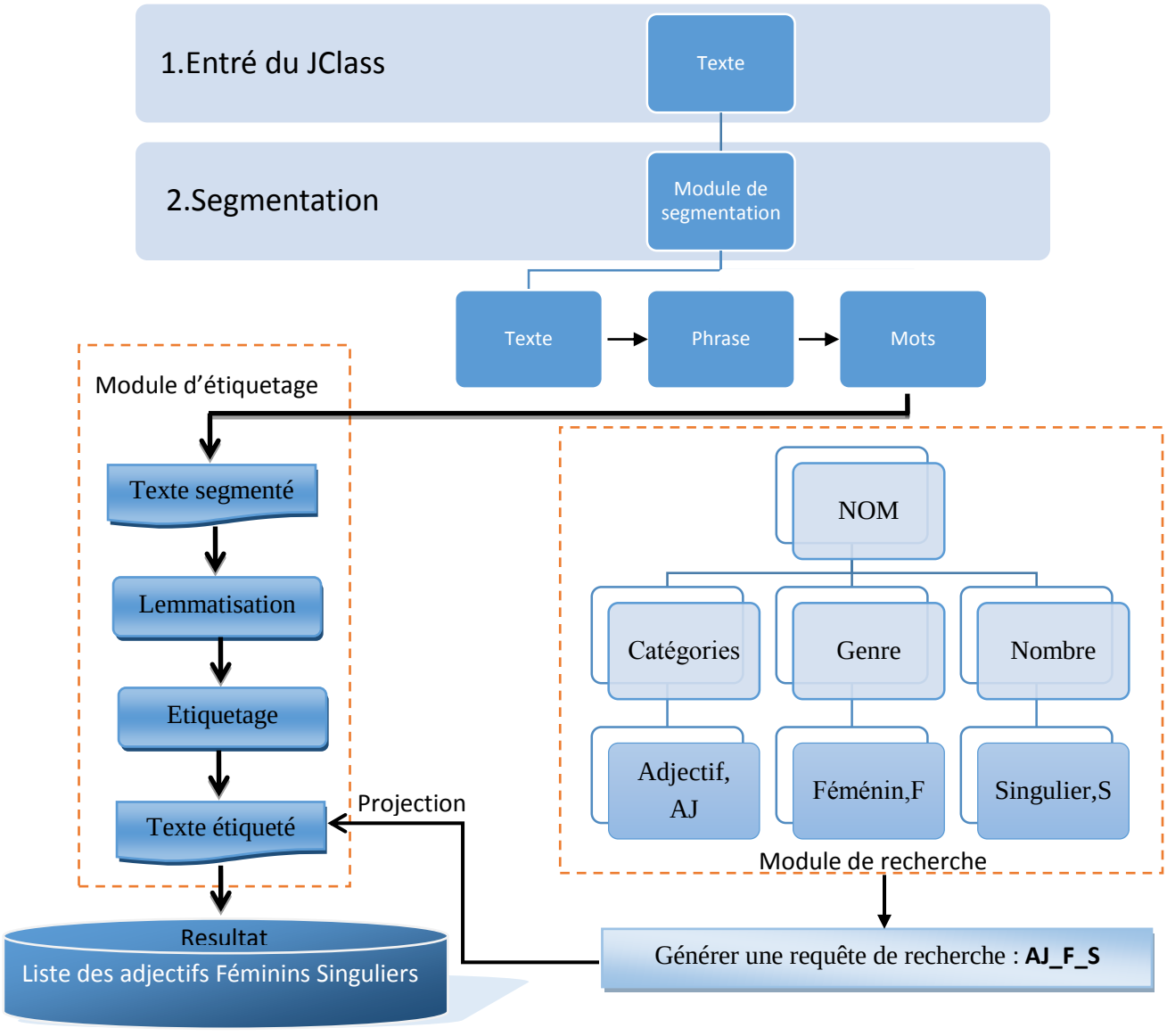

Figure 32 : *Processus de recherche et classification des adjectifs féminins singuliers*

6.6.2 Architecture de JExtract

JExtract est composé de deux (2) modules, le premier module a pour but de généré la requête de recherche pour la classification des mots selon la catégorie grammaticale sélectionné, et le deuxième pour étiqueter le texte à analyser on utilise l'étiqueteur SAIE[5] et leur grammaire d'étiquette.

6.6.2.1 Module d'étiquetage (SAIE)

Un étiqueteur morphosyntaxique de la langue arabe est un système complexe constitué de plusieurs modules (segmentation, étiqueteur non contextuel (analyseur morphologique), désambiguïseur) utilisant des ressources variées (lexique, jeu d'étiquettes, corpus, etc…).

a. Module de segmentation

Le module de segmentation consiste donc à identifier les frontières des mots, c'est-à-dire des unités lexicales autonomes que l'on cherche à étiqueter au niveau des principales parties du discours (verbe, nom, adjectif, adverbe, déterminant, etc…).

b. Module d'analyse morphologique

Le but principal de ce type d'analyse est de vérifier l'appartenance d'un mot donné au domaine linguistique choisi et de pouvoir disposer ainsi de tous les renseignements le concernant pouvant servir à l'analyse syntaxique.

c. Module de désambiguïsation

La désambiguïsation est une étape cruciale dans le processus d'étiquetage morphosyntaxique, à ce niveau du traitement si un mot est mal étiqueté, les règles de la grammaire s'appliqueront mal ou pas du tout.

Il faut dire que le module de désambiguïsation rentre en jeu dans un seul cas de figure, celui où l'unité lexicale (mot) reçoit plus d'une étiquette (plus d'une information morphosyntaxique), ce qui va générer une situation de confusion ou ambiguïté **[**42**]**.

Exemple : Considérons la phrase suivante : **دخل باسم قبل شاكر** / *Bassem entré avant Shakir*

Chaque mot dans la phrase ci-dessus a plus d'une analyse morphologique (voir le tableau 38). L'étiqueteur est responsable de l'attribution à chaque mot l'étiquette morphologique la plus approprié. Dans le tableau 38, l'étiquette correcte pour chaque mot est donnée en caractères gras. Le premier mot est en fait un verbe et donc le second mot est plus convenable d'être un nom propre à la place d'un adjectif ou une préposition qui est attaché à un nom.

[5] SAIE est un étiqueteur de la langue arabe développé par Fatma Nasser Al Shamsi en 2005 dans l'université de Sharjah.

Mot	Translittération	Etiquette	Signification
دخل	Dakhala	**verbe**	entrée
	Dakhl	nom	revenu
باسم	Baasim	**nom propre**	Bassem/Bassim
	Baasim	adjectif	souriant
	bi-smi	Préposition + Nom	Par / avec + nom
قبل	Qabla	**Préposition**	avant
	Qabila	Verbe parfait	acceptée
شاكر	shaakir	adjectif	reconnaissant
	shaakir	**nom propre**	Shakir

Table 38 : *L'étiquetage de la phrase* ***دخل باسم قبل شاكر***

6.6.2.2 Description du jeu d'étiquette d'étiqueteur SAIE

L'analyse qui a pour rôle d'associer à un mot graphique un ensemble d'informations décrivant les unités morphologiques et grammaticales entrant dans sa composition (proclitiques, préfixes, base, suffixes, enclitique) [43].

Dans la langue arabe, il existe deux genres : masculin et féminin. Et on distingue trois personne (1er, 2ème et 3ème). L'arabe se distingue des autres langues comme l'anglais et le français en ce qu'il dispose de trois nombres au lieu de deux (singulier, pluriel, et duel).

Ceci un exemple indiquant la structure morphologique et les catégories grammaticales du mot « **تأكلين** » :

	Suffixe	Racine	Préffixe
Mot arabe	**ين (iyn)**	**أكل (‘kul)**	**ت (ta)**
Analyse morphlogique	**Suffixe, 2ème personne,féminin**	**Verbe**	**Préffixe, 2ème personne**

Table 39 : *La structure morphologique et les catégories grammaticales du mot* **« تأكلين »**

a. Les noms

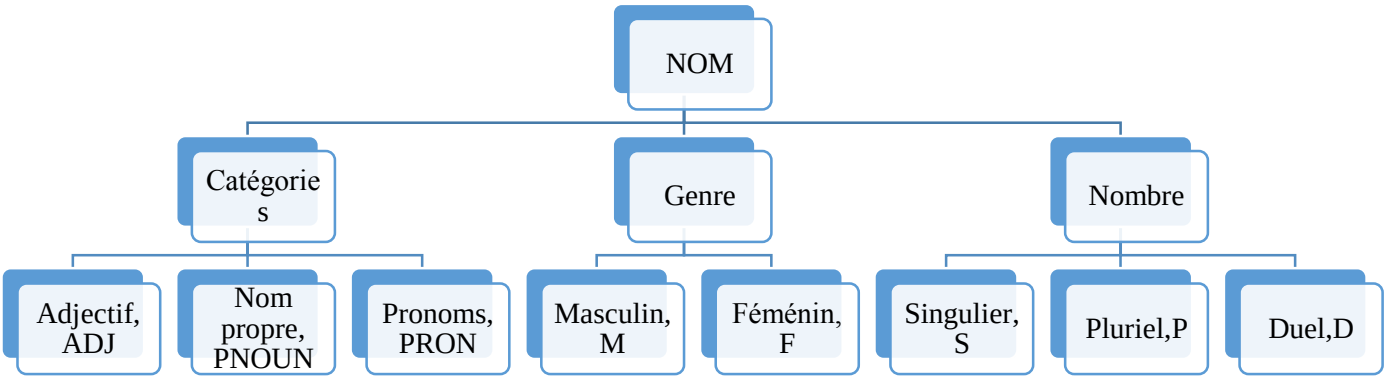

Figure 33 : *Les différentes classifications du nom et leurs étiquettes*

Exemple :

Mot	مسلمون	مسلمين	مسلمان	مسلمات
Translitération	Muslimuwn	muslimiyn	muslimaan	muslimaat
Description	Musulmans Masculin, pluriel	Musulmans Masculin, pluriel	Deux musulmans Masculin, duel	Musulmans Féminin, pluriel
Etiquette du suffixe	ون/ SUFF_M_P	ين/ SUFF_SUBJ_ALL	ان/ SUFF_M_D	ات/SUFF_F_P

Table 40 : *Différentes formes plurielles et double du mot* « مسلم »

b. Les verbes

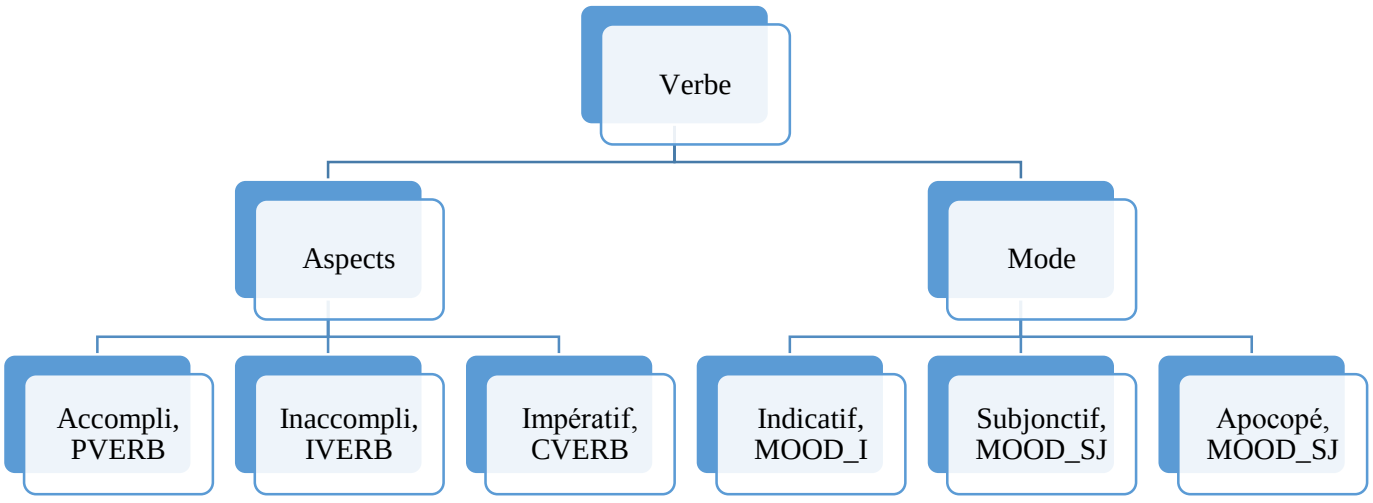

Figure 34 : *Les différentes classifications du verbe et leurs étiquettes*

c. Les particules

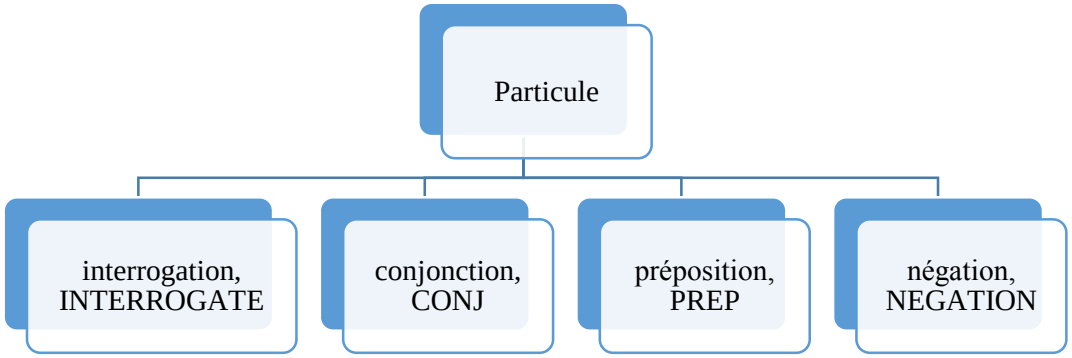

Figure 35 : *Les différentes classifications des particules et leurs étiquettes*

6.6.2.3 Module de recherche et classification

Ce module est composé de deux phases, une pour la création de la requête de recherche et l'autre pour la classification des mots étiqueté par la même étiquette de recherche, après la phase d'étiquetage du texte de l'entrée.

a. Classification des mots étiquetés

La classification des mots étiquetés se base sur le jeu d'étiquettes d'étiqueteur SAIE. Cet étiqueteur prend en entré le texte arabe et comme résultats on aura un texte étiqueté mais les mots de texte sont segmenté et séparé, on prend par exemple le mot suivant : « **طموحة** »

طموحة		
Segmentation	ة	طموح
Etiquette	SUFF_F_S	ADJ
Représentation	طموح/ADJ ة/ SUFF_F_S	

Table 41 : *Etiquetage du mot « **طموحة** » par SAIE*

Donc le problème c'est que quand on lance une requête de recherche pour les adjectifs on trouve l'adjectif « **طموح** », mais l'adjectif correcte est « **طموحة** », c'est une adjectif féminin singulier. Dans ce cas on ne peut pas lancer une recherche plus détaillé au niveau des catégories grammaticales.

Pour résoudre ce problème on doit concaténer tous les mots séparer résultants depuis l'étiqueteur SAIE avant de générer la requête de recherche complète. Mais comment on distingue le mot segmenté ? Et leurs composants ?

Nous avons fait quelques expériences sur un groupe de textes étiqueté et nous avons extrait toutes les suffixes et les préfixes possible qui peuvent être attaché à chaque catégorie grammaticale (noms, verbes, adjectifs, etc…). Et on a construit pour chaque catégorie un dictionnaire d'étiquettes qui contient les suffixes et les préfixes pour crée la liste des mots complets et correcte pour chaque catégorie avec leurs étiquettes détaillés.

Exemple :

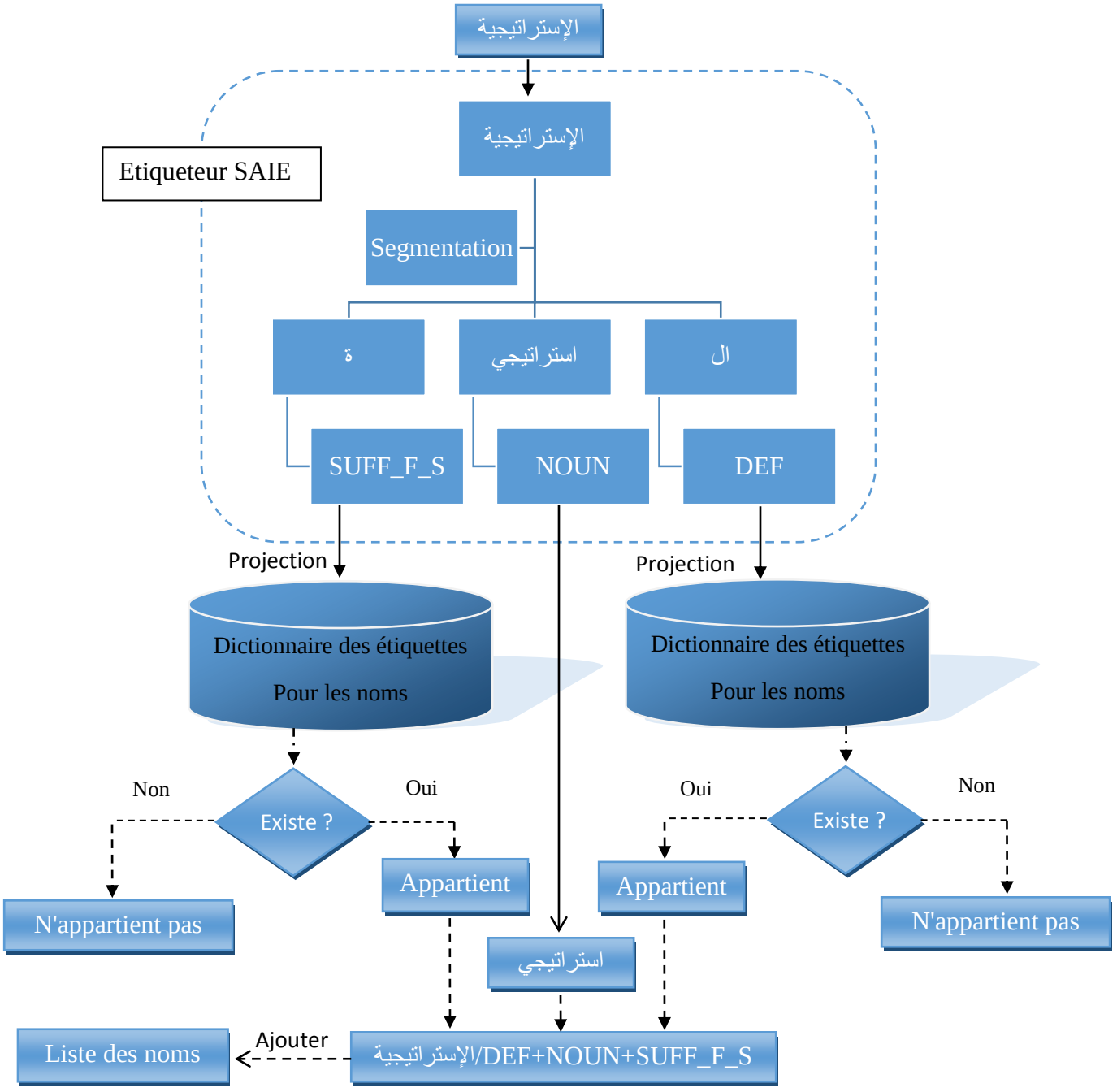

Figure 36 : *Processus de concaténation des mots étiqueté par SAIE*

b. Création de la requête de recherche

La création de la requête de recherche possède un prétraitement concernant le jeu d'étiquettes. La requête généré après le choix des catégories grammaticales a extrait doit être modifié afin de résoudre le problème d'insertion des mots étranges à le résultat de la catégorie sélectionnée.

Exemple : On veut chercher les noms féminins singuliers depuis la liste des mots étiquetés suivantes :

إدارة/NOUN+SUFF_F_S , طفرة/ NOUN+SUFF_F_S , نفس/ NOUN+SUFF_M_S ,حجم/ NOUN+SUFF_M_S

La requête est : NOUN pour sélectionner les noms, F pour féminin et S pour singulier (NOUN+SUFF_F_S)

Pour le premier mot de la liste il contient le mot NOUN donc il est un nom, il contient la lettre F donc il est un nom féminin et il contient la lettre S donc le mot vérifie tous les conditions de sélection même le deuxième mot. Mais le troisième contient le mot NOUN donc il est un nom, et contient la lettre F dans le mot SUFF mais il est un mot masculin, et contient la lettre S pour le singulier. La liste des résultats pour la requête des noms féminin singulier est : « إدارة،طفرة،نفس،حجم » est ça c'est faux**.**

Pour résoudre ce problème on a changé le jeu d'étiquette selon le tableau 42 :

Étiquette originale	Modification	Étiquette originale	Modification
SUFFDO	Z	PRON	R
SUFF	U	INDEF	B
DPRON	K	PREP	Q
PPRON	L	FUTURE	W
FUNC_WORD	T	SUBJ	Y
MOOD	G	DEF	E
INTERROGATE	H	NEGATION	X
CONJ	C	ADJ	AJ

Table 42 : *Table des étiquettes utilisées dans les requêtes de recherche*

Exemple : On veut chercher les noms féminins singuliers dans la phrase « **السماء صافية** »

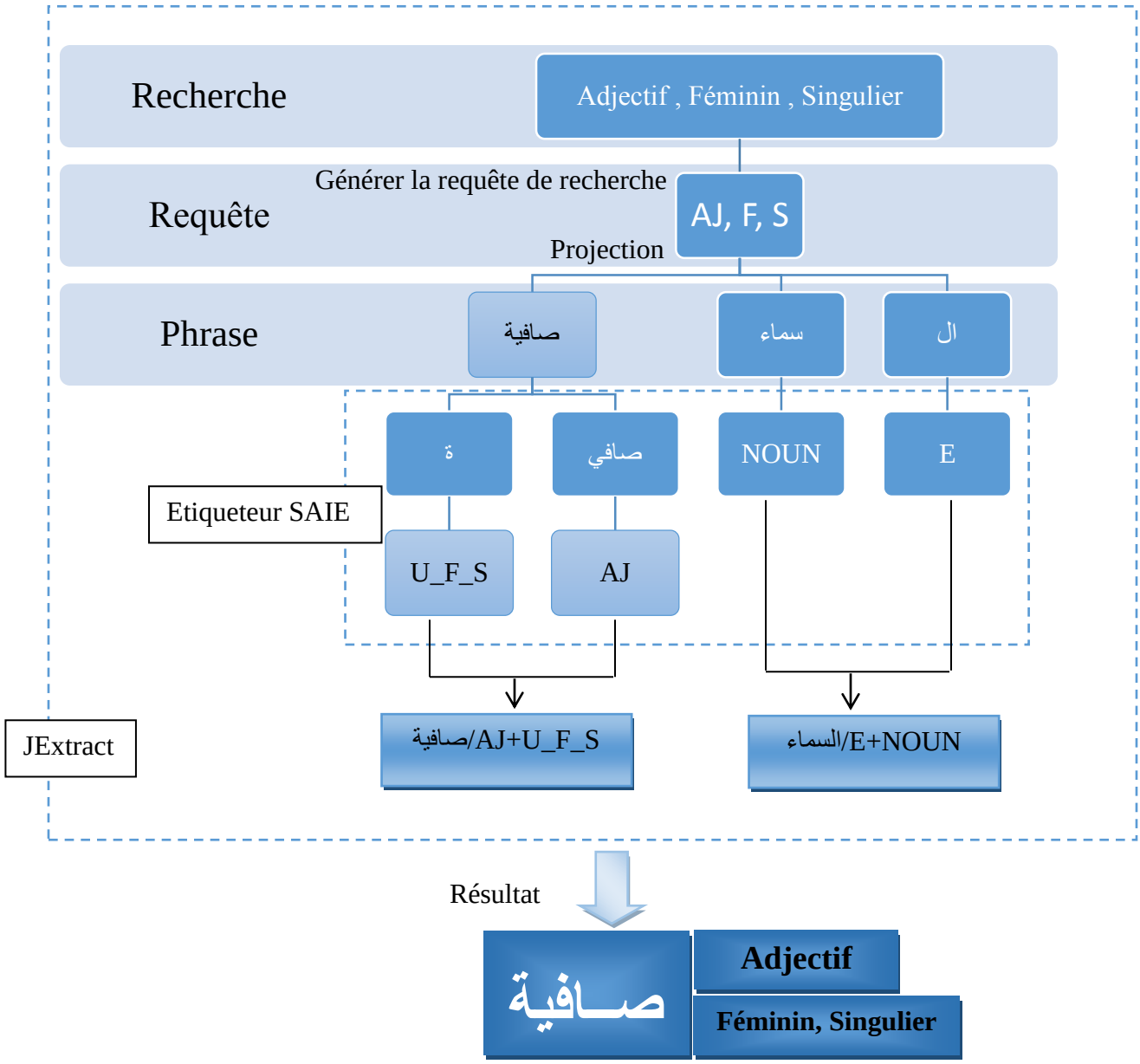

Figure 37 : *Processus de recherche d'un adjectif, féminin et singulier dans la phrase « **السماء صافية** »*

4. Conclusion

Le traitement automatique a facilité la tache d'acquisition des connaissances dans les langues naturelles et a étendu leurs champs d'application a de nombreuses disciplines scientifiques. Dans le cas de la langue arabe, l'aboutissement d'une boite à outils informatique pour l'acquisition des connaissances nécessite un travail préalable faisant appel à des resources lexicales et des outils de translittération et d'étiquetages morpho-syntaxique.

Dans le prochain chapitre nous présentons l'implémentation, l'expérimentation, et l'évaluation de notre boite à outils JEEM BOX.

CHAPITRE IV

Implémentation et résultats

1. Introduction

Après la phase de conception dans laquelle nous avons vu les différentes composantes de notre boite à outil JEEM BOX. Arrive la phase de présentation de JEEM BOX, d'où l'intérêt de ce chapitre qui est composé de cinq (05) axes qui tournent autour de notre boite à outil, le premier axe s'intéresse au langage de programmation, le second va mentionner les conditions de développement, le troisième va donner une description superficielle de JEEM BOX (interface graphique), le quatrième axe est consacré à une série d'exemples illustrant le fonctionnement des différents outils de JEEM BOX, et enfin le cinquième axe est consacré à une série d'expériences, des résultats et d'analyse de JEEM BOX.

2. Langage de développement

Figure 38 : *Icones de langage de développement*

Le langage de programmation que nous avons adopté pour implémenter notre application est le C#.Net[1] (prononcez « C sharp dot Net ») est un langage dit de « haut niveau », il sera immédiatement familier à C et C + +.

Visual C#[1] est l'environnement de développement des outils de Microsoft. Il comprend un environnement interactif de développement, des concepteurs visuels pour les applications Web, un compilateur et un débogueur. Visual C # fait partie d'une gamme de produits, appelée Visual Studio, qui comprend également Basic. NET, Visual C + +. NET et le langage de script JScript.

2.1 Pourquoi choisir C#

Il faut tout de même savoir, que ce langage s'adapte bien au domaine de l'application à savoir le traitement automatique des langues. Comme la manipulation des mots avec un langage de programmation dont la tâche n'est pas facile, le C#[1] offre une multitude d'instructions et de fonctions (prédéfinies) permettant le traitement des chaînes de caractères.

[1] Introduction au langage C# de Serge Tahé disponible sur http://tahe.developpez.com/dotnet/csharp/ (Accédé le 17/05/2014)

2.2 Caractéristiques et principes de conception du C#

- ✓ Etre facile d'utilisation pour les débutants.
- ✓ Etre un langage généraliste.
- ✓ Autoriser l'ajout de fonctionnalités pour les experts (tout en gardant le langage simple pour les débutants).
- ✓ Etre interactif.
- ✓ Fournir des messages d'erreur clairs et conviviaux.
- ✓ Avoir un délai de réaction faible pour les petits programmes.
- ✓ Ne pas nécessiter la compréhension du matériel de l'ordinateur.
- ✓ Isoler l'utilisateur du système d'exploitation.

Comme langage de programmation, C#[2] possède plusieurs caractéristiques parmi celles-ci on peut citer :

- ✓ Objets dynamiques (permet notamment la Programmation orientée prototype et la communication entre des langages dynamiques (JScript…) et les langages de la plateforme DotNet)
- ✓ Gestion implicite des interfaces
- ✓ Gestion des méthodes anonymes
- ✓ Simplification de l'écriture des tableaux, collections, listes et dictionnaires

3. L'environnement de développement

Nous avons développé notre boite à outil dans des conditions bien spécifiques, la liste suivante indique clairement les exigences de développement :

- ❖ Version du langage de programmation : C#.Net 2012
- ❖ Microsoft Access pour manipuler l'extension de notre la base de données (.mdb)
- ❖ Caractéristiques de la machine (ordinateur) :
  - ✓ Disque dur : 700Gb ;
  - ✓ Résolution de l'écran : 1366*768.

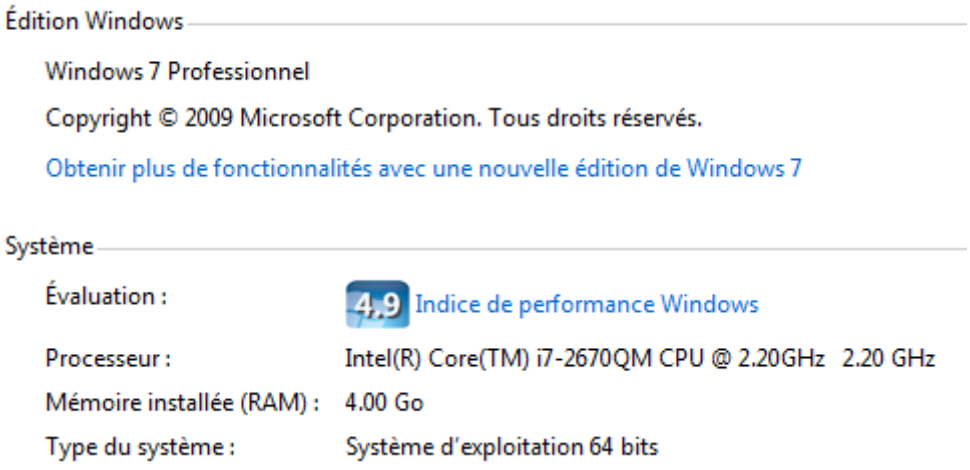

Figure 39 : Capture d'écran décrivant les ca*ractéristiques de la machine*

[2] Introduction au langage C# de Serge Tahé disponible sur http://tahe.developpez.com/dotnet/csharp/ (Accédé le 17/05/2014)

4. Description de l'interface graphique de JEEM BOX

Nous allons présenter dans ce point, la description de l'interface graphique de notre application, de ce fait, on va essayer de décrire chaque outil et chaque composant de l'interface toute en mentionnant sa fonctionnalité.

4.1 Accueil

La figure ci-dessous présente l'interface graphique d'accueil

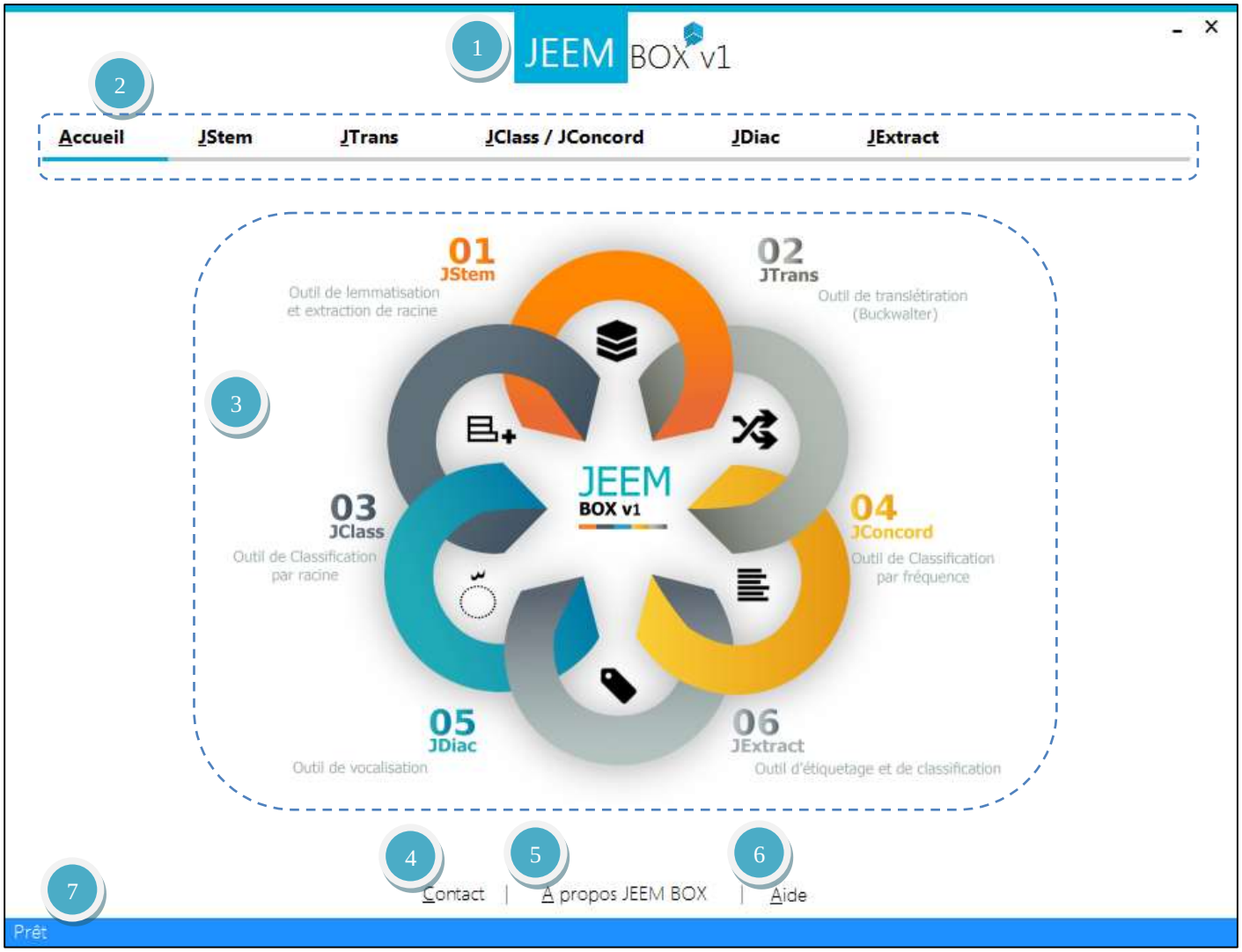

Figure 40 : *Capture d'écran de l'interface générale de JEEM BOX (Accueil)*

Numéro	Description
1	Nom et la version du la boite à outil JEEM BOX, version 1
2	Barre de navigation
3	Page d'accès rapide aux outils

Table 43 A : *Description des composants de l'interface d'accueil*

Numéro	Description
4	Afficher les informations de contact
5	A propos JEEM BOX
6	Affiche l'aide et les instructions d'utilisation
7	Barre de notification

Table 43 B: *Description des composants de l'interface d'accueil*

4.2 JStem

La figure ci-dessous présente l'interface graphique de JStem

Figure 41 : *Capture d'écran de l'interface générale de JStem*

Numéro	Description
1	Zone du texte d'entrée
2	Zone de texte pour affiche le résultat
3	Zone de texte pour la saisie d'un seul mot à la fois
4	Zone de texte pour affiche le résultat d'un seul mot saisie dans la case numéro 3
5	Bouton pour ouvrir un fichier texte qui contient des données
6	Boutons de fonction du lemmatiseur
7	Zone de boutons raccourcis d'édition de texte
8	Zone des méthodes
9	Zone de structure de mot

Table 44 : *Description des composants de l'interface de JStem*

4.2.1 Zone de boutons raccourcis d'édition de texte (7)

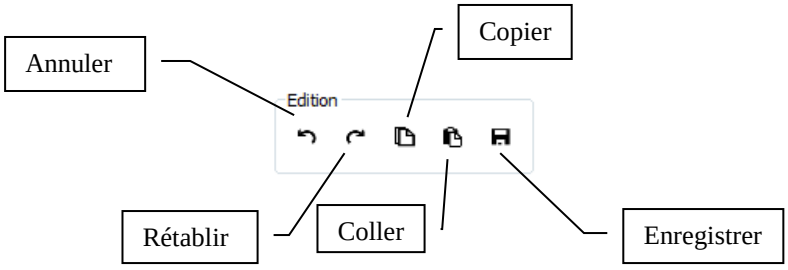

Figure 42 : *Description des composants de l'interface de la zone d'édition de JStem*

4.2.2 Zone des méthodes de JStem (8)

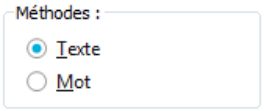

Figure 43 : *Composants de la zone des méthodes de JStem*

Méthode	Description
Texte	Faire la lemmatisation de texte dans la zone du texte entré
Mot	Faire la lemmatisation de mot dans saisie dans la zone 3 et afficher la structure de mot lemmatiser

Table 45 : *Description des composants de la zone des méthodes de JStem*

4.2.3 Zone de structure de mot (9)

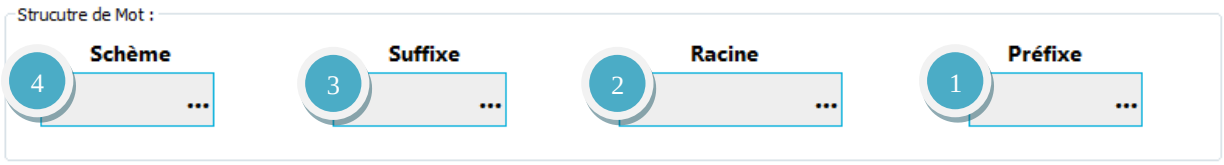

Figure 44 : *Composants de la zone de structure de mot*

Numéro	Description
1	Afficher le préfixe de mot après la lemmatisation
2	Afficher la racine de mot après la lemmatisation
3	Afficher le suffixe de mot après la lemmatisation
4	Afficher le schème de mot après la lemmatisation

Table 46 : *Description des composants de la zone de structure de mot*

4.3 JTrans

La figure ci-dessous présente l'interface graphique de JTrans

Figure 45 : *Capture d'écran de l'interface générale de JTrans*

Numéro	Description
1	Zone du texte d'entrée
2	Zone de texte pour affiche le résultat
3	Zone de texte pour la saisie d'un seul mot à la fois
4	Zone de texte pour affiche le résultat d'un seul mot saisie dans la case numéro 3
5	Bouton pour ouvrir un fichier texte qui contient des données
6	Boutons de fonction de la translitération
7	Zone de boutons raccourcis d'édition de texte
8	Zone des méthodes

Table 47 : *Description des composants de l'interface de Jtrans*

4.3.1 Zone des méthodes de JTrans (8)

Figure 46 : *Composants de la zone des méthodes de JTrans*

Méthode	Description
Un seul mot	Faire la translitération de mot dans saisie dans la zone 3 et afficher la structure de mot lemmatiser
Texte	Faire la translitération de texte dans la zone du texte entré
L'inverse	Faire la translitération inverse (buckwalter vers arabe)
Quran texte	Prend en considération les signes utilisés dans le quran

Table 48 : *Description des composants de la zone des méthodes de JTrans*

4.4 JClass et JConcord

La figure ci-dessous présente l'interface graphique de JClass et de JConcord

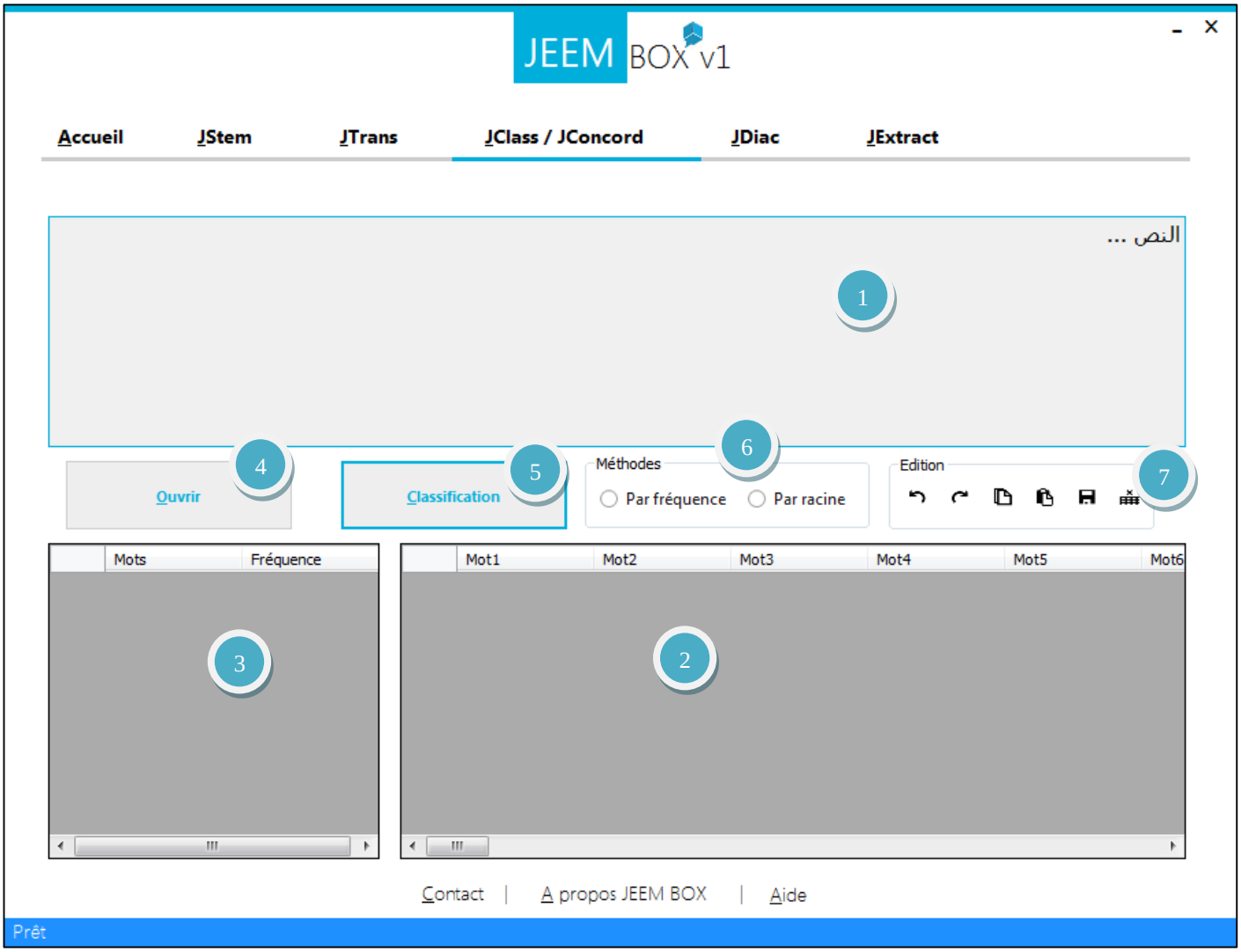

Figure 47 : *Capture d'écran de l'interface générale de JClass et de JConcord*

Numéro	Description
1	Zone du texte d'entrée
2	Zone de texte pour affiche le résultat de classification par racine
3	Zone de texte pour affiche le résultat classification par fréquence
4	Bouton pour ouvrir un fichier texte qui contient des données
5	Boutons de fonction de la classification/concordance
6	Zone des méthodes
7	Zone de boutons raccourcis d'édition de texte

Table 49 : *Description des composants de l'interface de JClass/JConcord*

4.4.1 Zone des méthodes de JClass/JConcord (6)

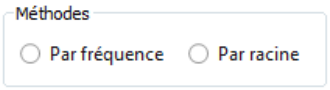

Figure 48 : *Composants de la zone des méthodes de JClass/JConcord*

Méthode	Description
Par fréquence	Fonction de classification par fréquence (JConcord)
Par racine	Fonction de classification par racine (JClass)

Table 50 : *Description des composants de la zone des méthodes de JClass/JConcord*

4.4.2 Zone de boutons raccourcis d'édition de texte de JClass/JConcord (7)

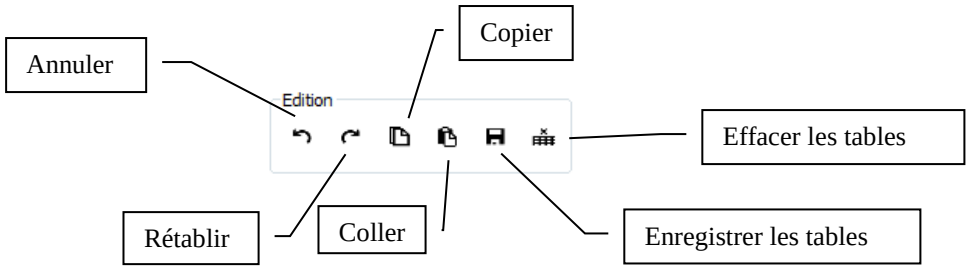

Figure 49 : *Description des composants de la zone d'édition de JClass/JConcord*

4.5 JDiac

La figure ci-dessous présente l'interface graphique de JDiac

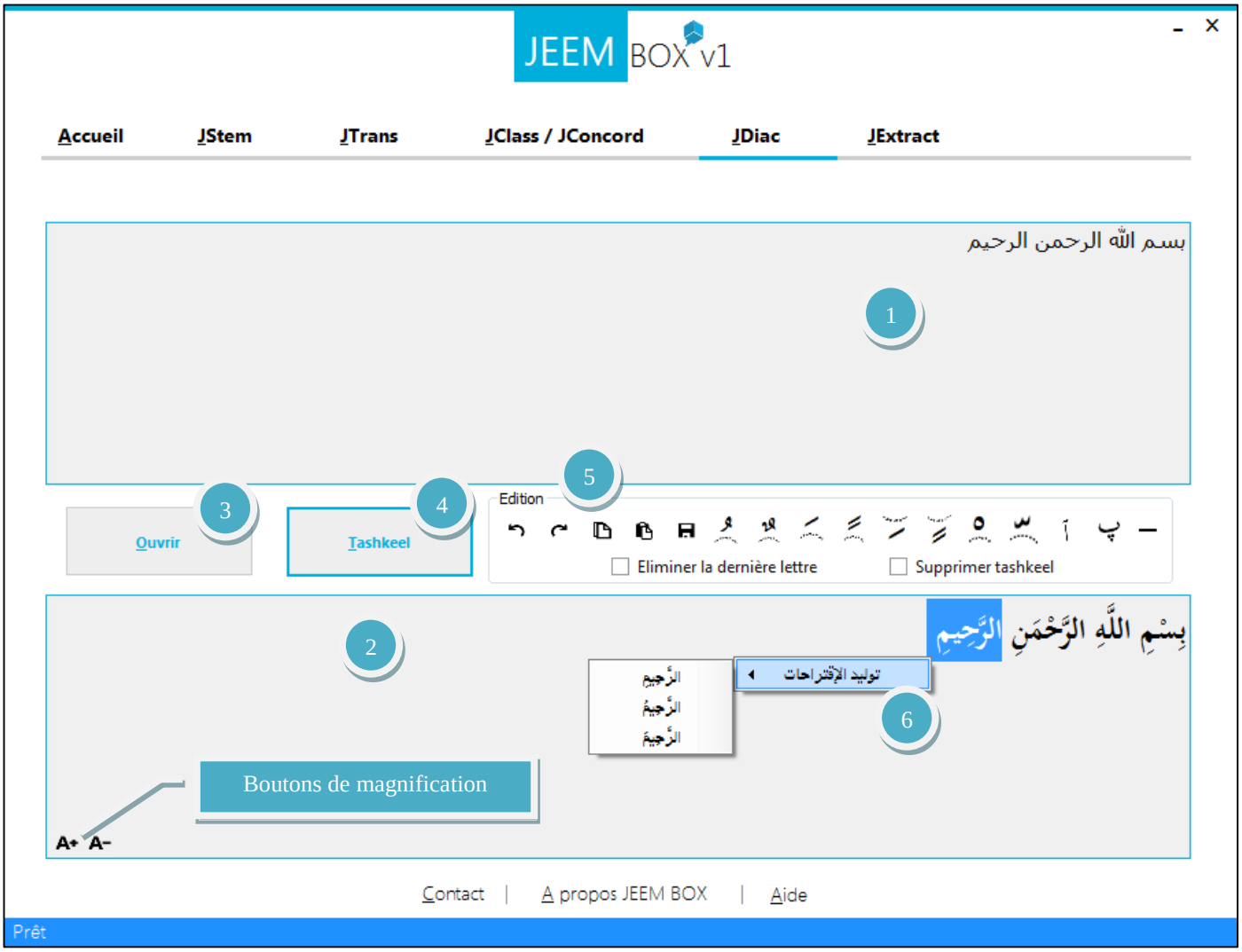

Figure 50 : *Capture d'écran de l'interface générale de JDiac*

Numéro	Description
1	Zone du texte d'entrée
2	Zone de texte pour affiche le résultat de vocalisation
3	Bouton pour ouvrir un fichier texte qui contient des données
4	Boutons de fonction de la vocalisation
5	Zone de boutons raccourcis d'édition de texte
6	Afficher les suggestions de vocalisation (modification) d'un mot

Table 51 : *Description des composants de l'interface de JDiac*

4.5.1 Zone de boutons raccourcis d'édition de texte de JDiac (5)

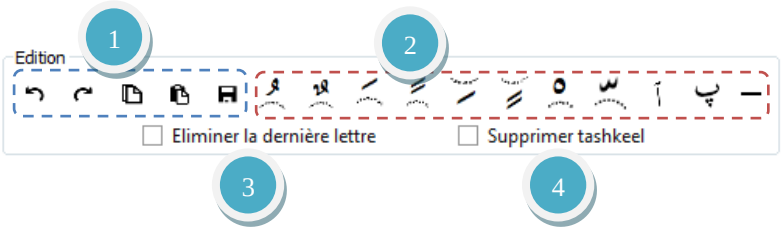

Figure 51 : *Composants de la zone d'édition de JDiac*

Numéro	Description
1	Zone de boutons raccourcis d'édition de texte classique (copie, coller …)
2	Pour ajouter des signes de vocalisation rapidement
3	Eliminer la dernière lettre dans la phase de vocalisation
4	Supprimer la vocalisation dans le texte dans la zone d'entré

Table 52 : *Description des composants de la zone d'édition de JDiac*

4.5.2 Liste des suggestions de vocalisation (6)

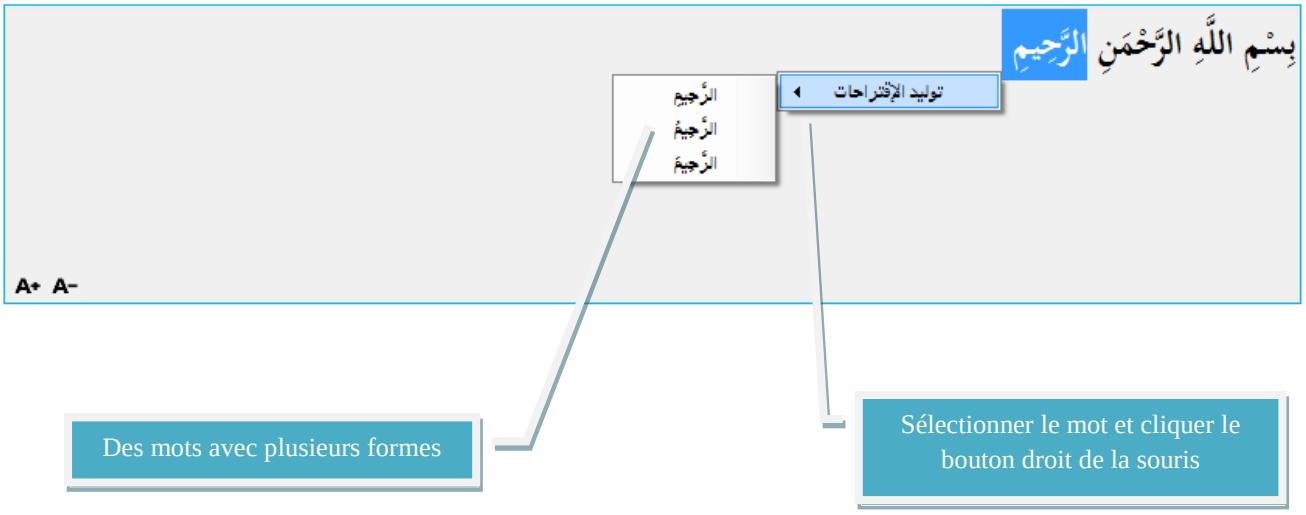

Figure 52 : *Description des composants de la liste des suggestions de JDiac*

4.6 JExtract

La figure ci-dessous présente l'interface graphique de JExtract

Figure 53 : *Capture d'écran de l'interface générale de JExtract*

Numéro	Description
1	Zone du texte d'entrée
2	Zone de texte pour affiche le résultat de recherche (filtrage)
3	Bouton pour ouvrir un fichier texte qui contient des données
4	Boutons de la fonction d'extraction
5	Exporter la liste d'étiquetage sous forme (mot/étiquette)
6	Zone de boutons raccourcis d'édition de texte

Table 53 A: *Description des composants de l'interface de JExtract*

Numéro	Description
7	Zone d'étiqueteur
8	Zone des options de recherche (filtrage)

Table 53 B: *Description des composants de l'interface de JExtract*

4.6.1 Zone d'étiqueteur (7)

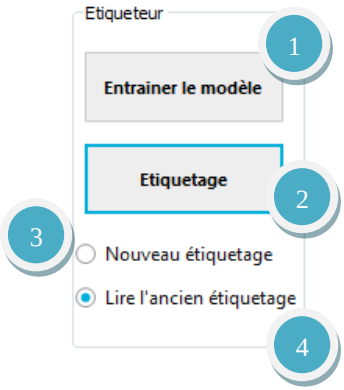

Figure 54 : *Composants de la zone d'étiqueteur de JExtract*

Numéro	Description
1	Entrainer le modelé HMM d'étiqueteur à nouveau
2	Etiqueter le texte d'entré si l'option 3 est sélectionné ou lire l'ancienne donnée d'étiquetage (depuis le texte précédant)
3	Générer des nouveau données d'étiquetage pour le texte dans la zone d'entré
4	Lire l'ancienne donnée d'étiquetage

Table 54 : *Description des composants de la zone d'étiqueteur de JExtract*

4.6.2 Zone des options de recherche (8)

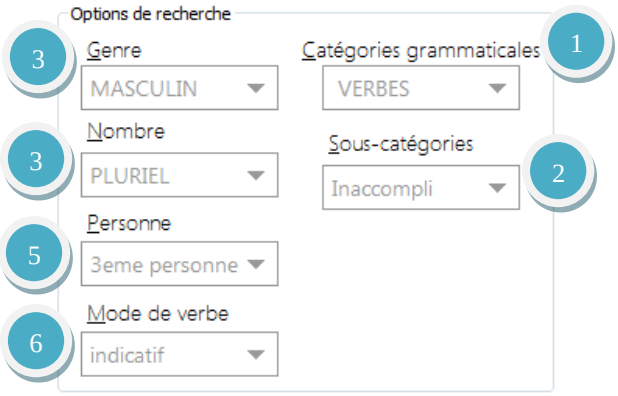

Figure 55 : *Exemple de la zone des options pour l'extraction dans JExtract*

Numéro	Description
1	Contient les 3 catégories grammaticales (Nom, Verbes, Particules)
2	Contient les sous-catégories des 3 catégories grammaticales
3	Contient le genre d'un mot (Masculin, Féminin)
4	Contient le Nombre d'un mot (Singulier, Duel, Pluriel)
5	Contient la personne d'un mot (Singulier, Duel, Pluriel)
6	Contient le mode d'un verbe (indicatif, subjonctif ou apocopé)

Table 55 : *Description des composants de la zone des options de JExtract*

5. Exemples sur le fonctionnement du JEEM BOX

Nous présentons dans ce point une série d'exemples démontrant le fonctionnement de notre boite à outil.

5.1 JStem

Exemple 1 :

Lemmatisation de la phrase :

ذهبت الطالبة الصغيرة الى المدرسة، ودرست الدروس جميعها، وحين قرب وقت الاختبار، نجحت طالبتنا بامتياز! المدارس لها دور كبير في تعليم طلابنا.

Exemple 2 :

Et le mot : **يستأذنونك**

Figure 56 : *Exemples de lemmatisation avec JStem*

5.2 JTrans

Exemple de translitération de la phrase précédente dans les deux sens (sens inverse dans la figure 57)

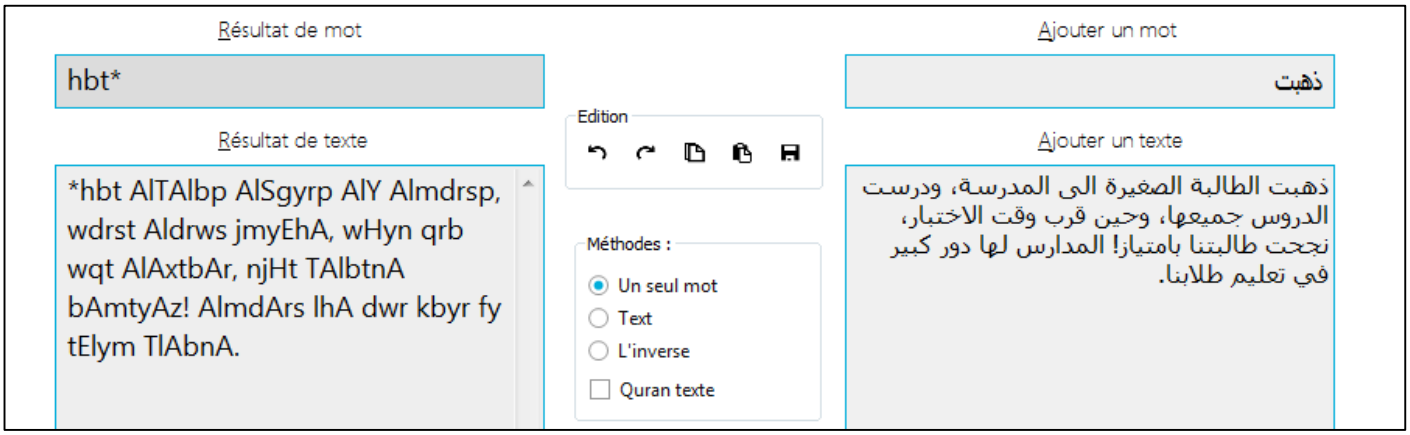

Figure 57 : *Exemple 1 de la translitération avec JTrans*

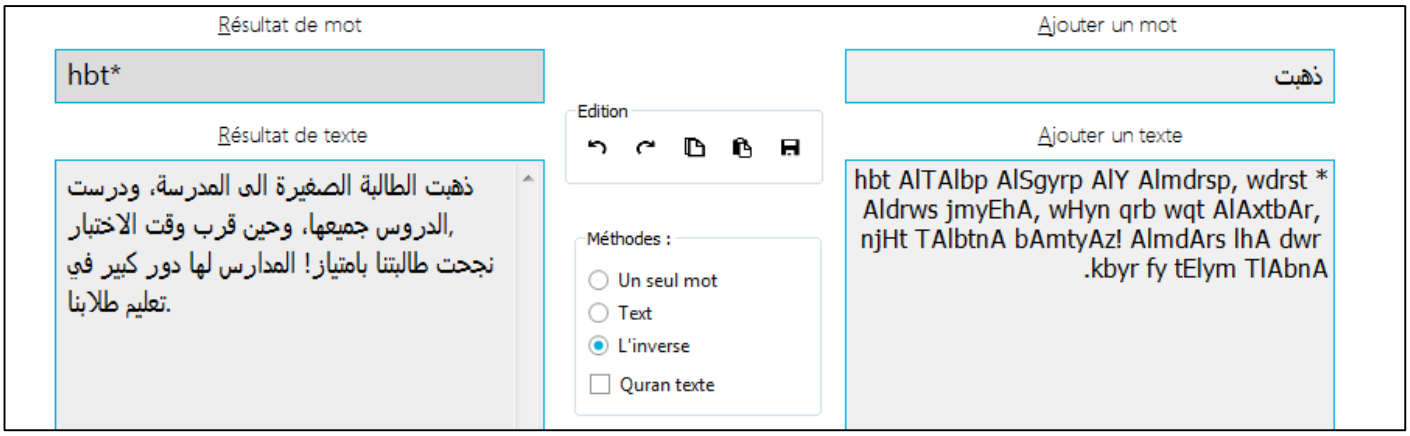

Figure 58 : *Exemple 2 de la translitération avec JTrans (sens inverse)*

5.3 JClass/JConcord

Exemple de Classification de la même phrase précédente par JClass et JConcord :

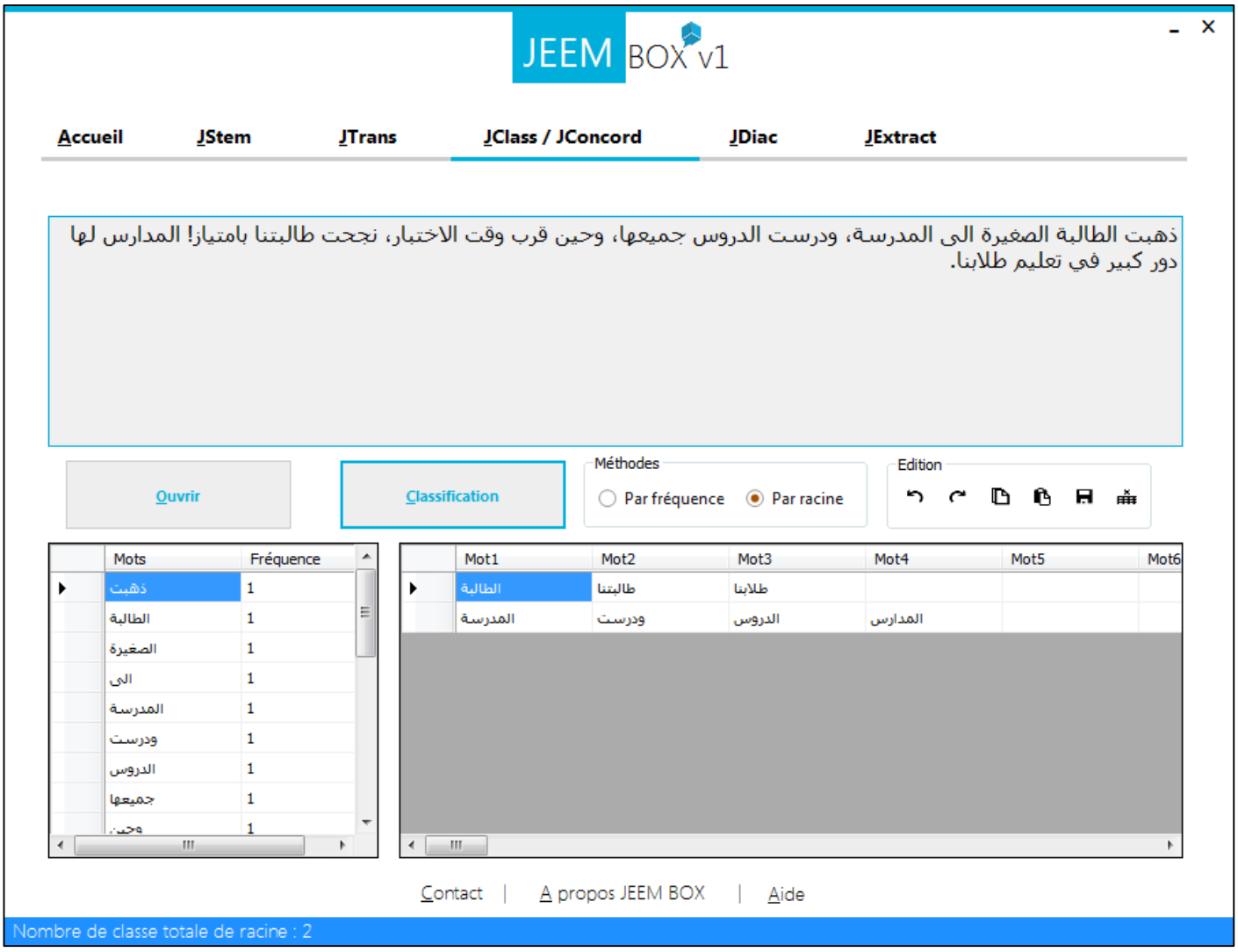

Figure 59 : *Exemple de classification par JClass et JConcord*

5.4 JClass/Jconcord

Exemple de vocalisation de la phrase : **ذهبت سلمى إلى السوق و عادت.**

Figure 60 : *Exemple de vocalisation par JDiac*

5.5 JExtract

Dans l'exemple suivant, on cherche de trouver les verbes selon les options suivantes :

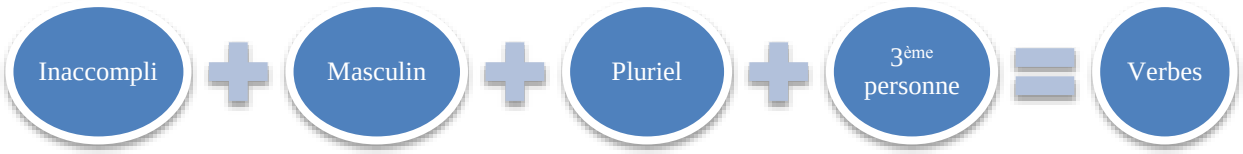

Figure 61 : *Exemple des options d'extraction pour trouver les verbes*

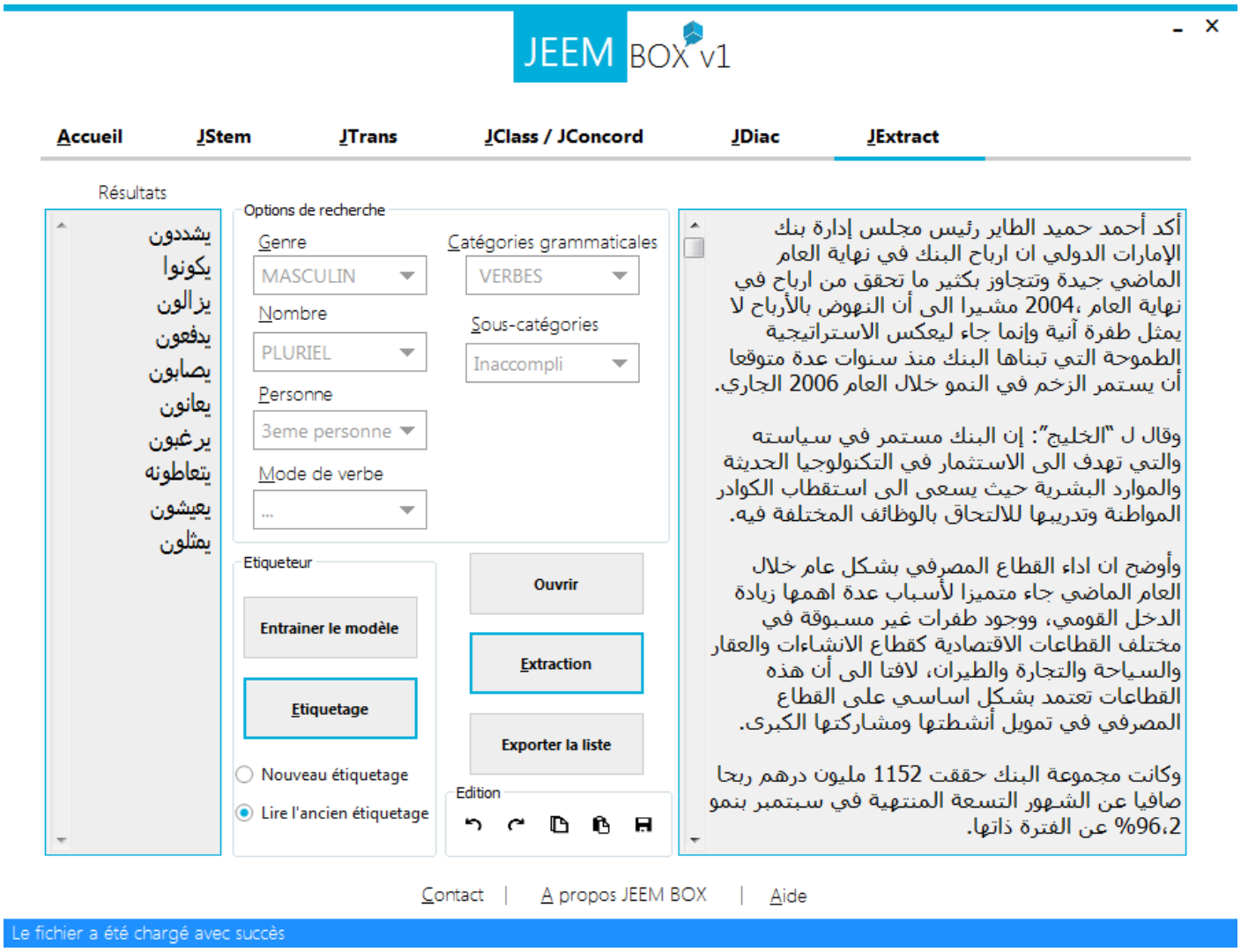

Figure 62 : *Exemple d'extraction des verbes selon les options citées précédemment par JExtract*

6. Expériences et résultats

6.1 JStem

Des expériences sont appliquées en utilisant une partie du « **Quran** » sur notre outil pour l'extraction des racines. Les mots du « **Quran** » ont été classifie selon l'ordre alphabétique de racines, et la fréquence du mot. Les mots sont organisés dans des classes. Nous avons organisé les classes dans 4 groupes, 150 classes par groupe.

Exemple :

Racine : **صفحٌ**

Mots ayant la même racine et leur fréquence :

الصَّفْحَ (1) صَفْحًا (1) فَاصْفَحْ (1) فَاصْفَحِ (1) وَاصْفَحْ (1) وَاصْفَحُوا (1) وَتَصْفَحُوا (1) وَلْيَصْفَحُوا (1)

Texte	Caractéristiques	Valeur
Quran	Nombre de documents	1
	Taille	1264 KB
	Nombre de catégories	1
	Nombre de mots	~10000
	Nombres de classes	544

Table 56 : *Caractéristique du test de JStem*

6.1.1 Exemple d'une table de résultats

Moyenne de mots correcte (%)	Mot correcte	Totale des mots
94.73684211	18	19
95.83333333	23	24
100	9	9
100	76	76
100	14	14
85.71428571	6	7
79.16666667	19	24
97.14285714	34	35
100	26	26

Table 57 : *Exemple d'une table des résultats des tests*

6.1.2 Graphes de résultats pour chaque groupe

Nous avons obtenus les graphes de résultats suivants, nous citons dans chaque graphe les mesures suivants :

- ✓ Moyenne de mots correcte (Moyenne %)
- ✓ Totale des mots (Nombre de mots dans chaque classe)

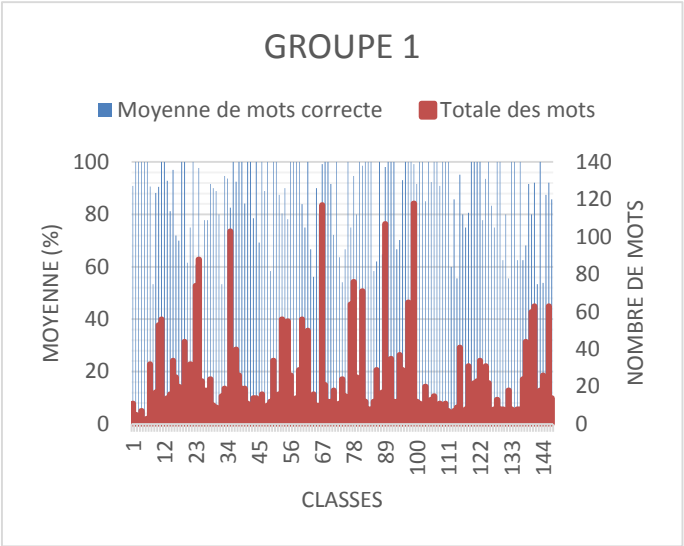

Figure 63 : *Résultats du groupe 1*

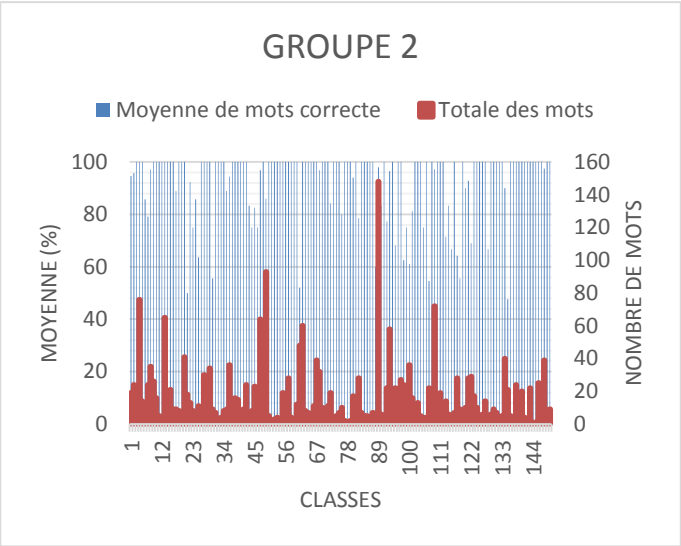

Figure 64 : *Résultats du groupe 2*

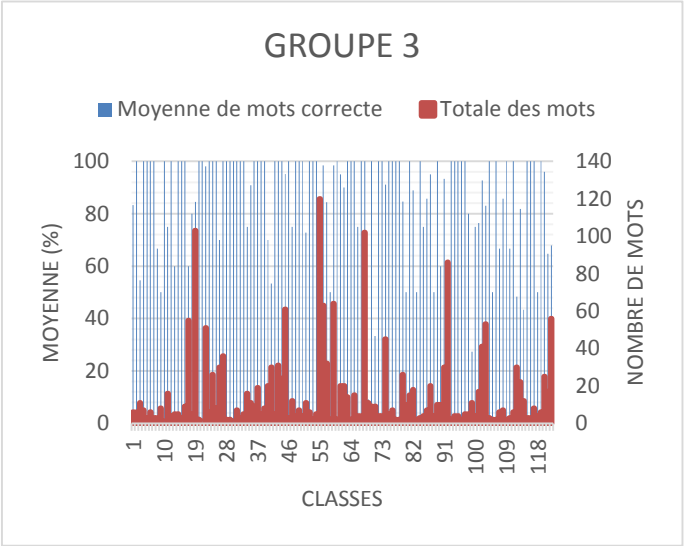

Figure 65 : *Résultats du groupe 3*

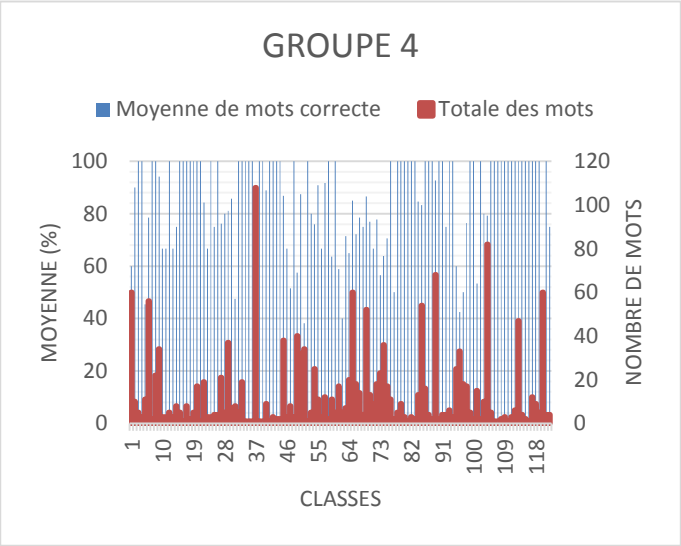

Figure 66 : *Résultats du groupe 4*

6.2 JClass

Des expériences sont appliquées en utilisant une partie du corpus "Al-Watan 2004" sur notre outil.

Il est composé de 300 documents de différentes catégories (économie, politique, locale, culture, internationale et sportives).

Corpus	Caractéristiques	Valeur
Al-watan 2004	Nombre de documents	360
	Taille	1440 KB
	Nombre de catégories	6
	Nombre de mots	~180000
	Nombre de mots dans chaque document	450-500

Table 58 : *Caractéristique du test de JClass*

6.2.1 Exemples des tables de résultats de chaque catégorie

A. Culture

Catégorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
Culture	500	67	57	10	85.07462687	4 Kb
Culture	500	91	89	2	97.8021978	4 Kb
Culture	500	71	60	11	84.50704225	4 Kb
Culture	500	81	74	7	91.35802469	4 Kb
Culture	500	80	74	6	92.5	4 Kb
Culture	500	80	77	3	96.25	4 Kb
Culture	500	67	63	4	94.02985075	4 Kb
Culture	500	81	77	4	95.0617284	4 Kb
Culture	500	60	56	4	93.33333333	4 Kb

Table 59 : *Exemple d'une table de résultats de la catégorie culture*

B. Economie

Catégorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
Economy	500	101	95	6	94.05940594	4 Kb
Economy	500	102	97	5	95.09803922	5 Kb
Economy	500	111	107	4	96.3963964	6 Kb
Economy	500	103	97	6	94.17475728	7 Kb
Economy	500	84	79	5	94.04761905	8 Kb
Economy	500	96	95	1	98.95833333	9 Kb
Economy	500	96	92	4	95.83333333	10 Kb
Economy	500	93	89	4	95.69892473	11 Kb
Economy	500	95	90	5	94.73684211	12 Kb

Table 60 : *Exemple d'une table de résultats de la catégorie économie*

C. Internationale

Catégorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
Internationale	500	72	70	2	97.22222222	5 Kb
Internationale	500	75	73	2	97.33333333	4 Kb
Internationale	500	73	69	4	94.52054795	3 Kb
Internationale	500	75	73	2	97.33333333	2 Kb
Internationale	500	59	56	3	94.91525424	1 Kb
Internationale	500	62	60	2	96.77419355	1 Kb
Internationale	500	84	79	5	94.04761905	1 Kb
Internationale	500	75	72	3	96	2 Kb
Internationale	500	74	71	3	95.94594595	3 Kb

Table 61 : *Exemple d'une table de résultats de la catégorie internationale*

D. Local

Categorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
Local	500	81	78	3	96.2962963	5 Kb
Local	500	87	77	10	88.50574713	4 Kb
Local	500	66	66	0	100	3 Kb
Local	500	73	73	0	100	2 Kb
Local	500	60	59	1	98.33333333	1 Kb
Local	500	71	70	1	98.5915493	1 Kb
Local	500	76	70	6	92.10526316	1 Kb
Local	500	84	82	2	97.61904762	2 Kb
Local	500	71	64	7	90.14084507	3 Kb

Table 62 : *Exemple d'une table de résultats* de *la catégorie local*

E. Religion

Categorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
Religion	500	93	85	8	91.39784946	4 Kb
Religion	500	75	73	2	97.33333333	5 Kb
Religion	500	70	70	0	100	6 Kb
Religion	500	74	71	3	95.94594595	7 Kb
Religion	500	105	98	7	93.33333333	8 Kb
Religion	500	80	78	2	97.5	9 Kb
Religion	500	56	49	7	87.5	10 Kb
Religion	500	93	89	4	95.69892473	11 Kb
Religion	500	80	70	10	87.5	12 Kb

Table 63 : *Exemple d'une table de résultats de la catégorie religion*

F. Sport

Categorie	Nombre Mots	N Classe	N classe Correcte	N Classe Faux	Avg Classe correcte (%)	Taille
sports	500	82	69	13	84.14634146	4 Kb
sports	500	86	76	10	88.37209302	5 Kb
sports	500	83	74	9	89.15662651	6 Kb
sports	500	81	68	13	83.95061728	7 Kb
sports	500	88	74	14	84.09090909	8 Kb
sports	500	66	56	10	84.84848485	9 Kb
sports	500	75	67	8	89.33333333	10 Kb
sports	500	87	82	5	94.25287356	11 Kb
sports	500	78	58	20	74.35897436	12 Kb

Table 64 : *Exemple d'une table de résultats de la catégorie sport*

6.2.2 Graphes de résultats pour chaque catégorie

Selon les 6 catégories existantes dans le corpus Al-watan 2004 nous avons obtenus les graphes de résultats suivants, nous citons dans chaque graphe les mesures suivants :

- ✓ Nombre des classes correctes (N classe correcte)
- ✓ Nombre des classes faux (N classe faux)
- ✓ Nombre des articles pour chaque catégorie (Articles)
- ✓ La moyenne (%) des classes correctes
- ✓ Nombre totale des classes (N Classe)

A. Culture

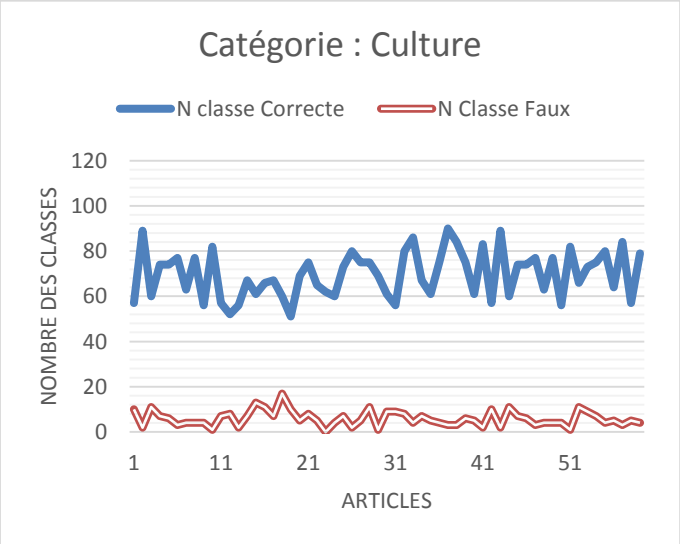

Figure 67 : *Résultats de la catégorie : culture*

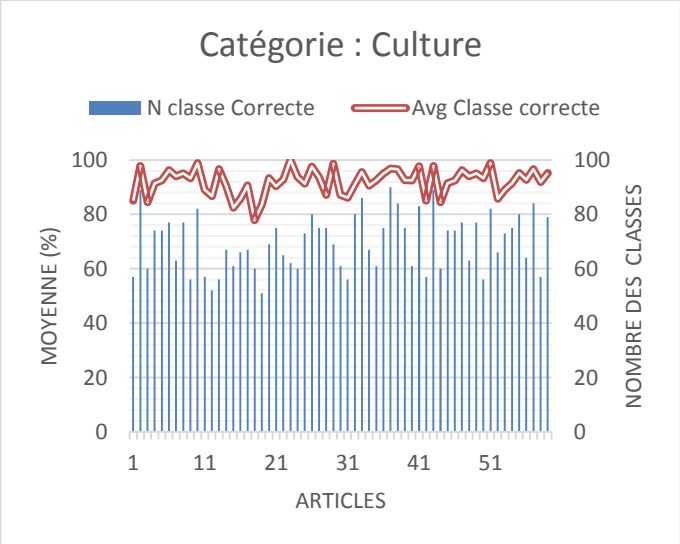

Figure 68 : *Représentation de la Moyenne des classes correctes de la catégorie culture*

B. Economie

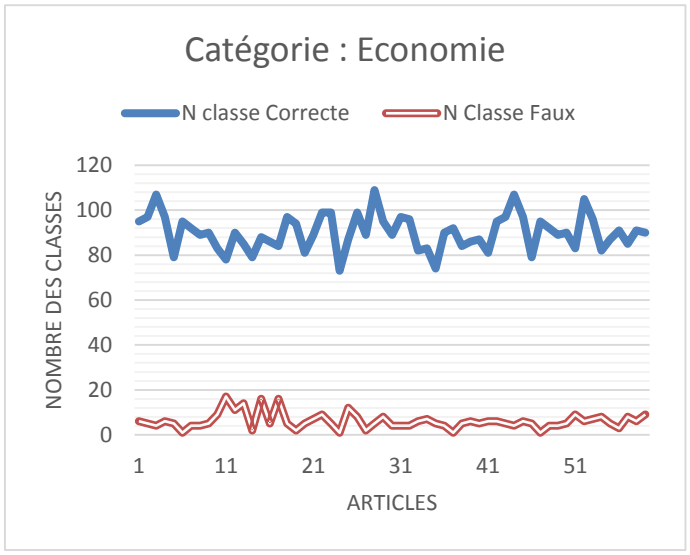

Figure 69 : *Résultats de la catégorie : économie*

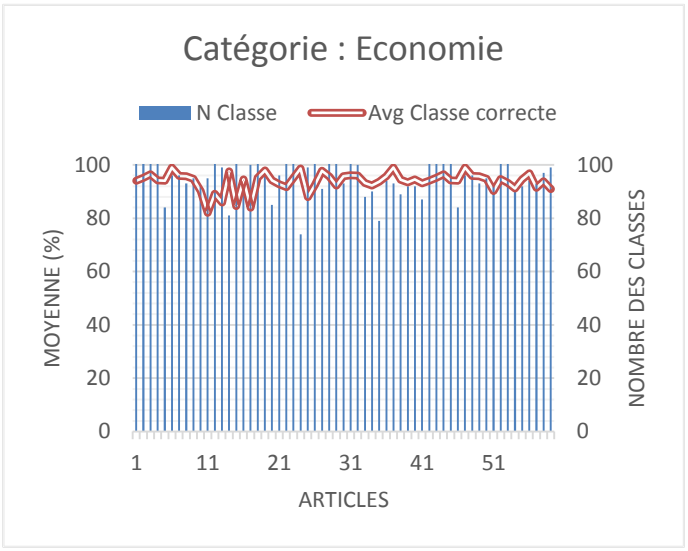

Figure 70 : *Représentation de la Moyenne des classes correctes de la catégorie économie*

C. Internationale

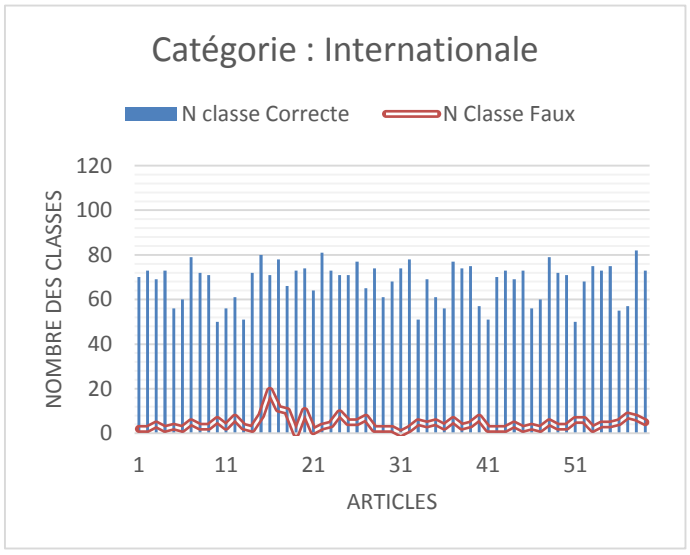

Figure 71 : *Résultats de la catégorie : internationale*

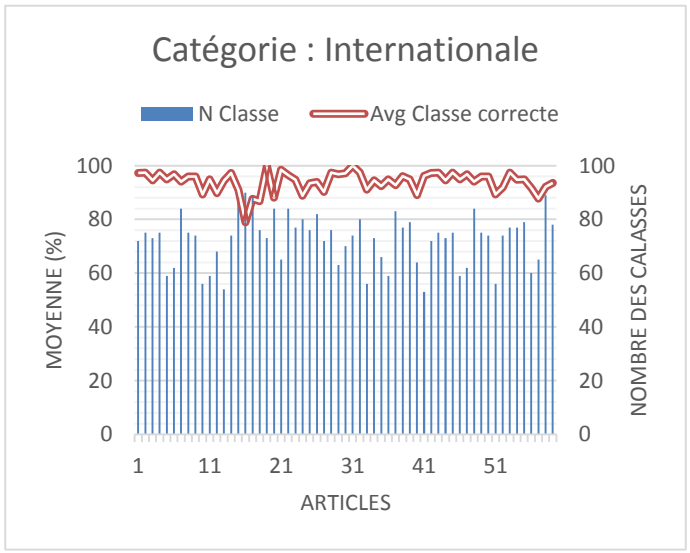

Figure 72 : *Représentation de la Moyenne des classes correctes de la catégorie internationale*

D. Local

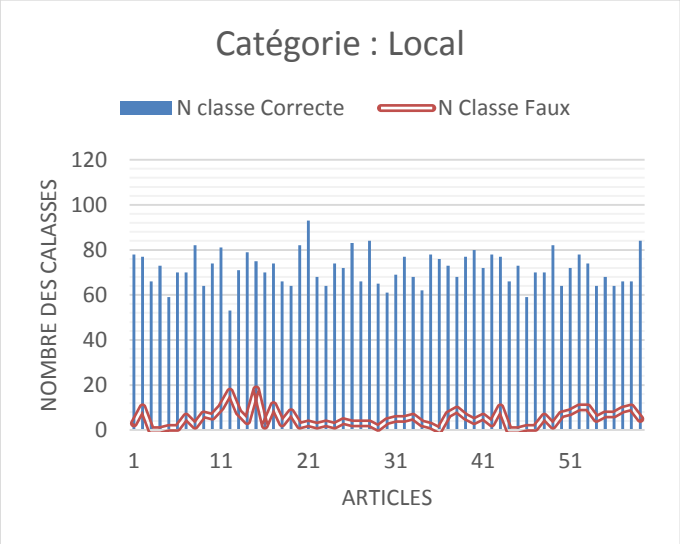

Figure 73 : *Résultats de la catégorie : local*

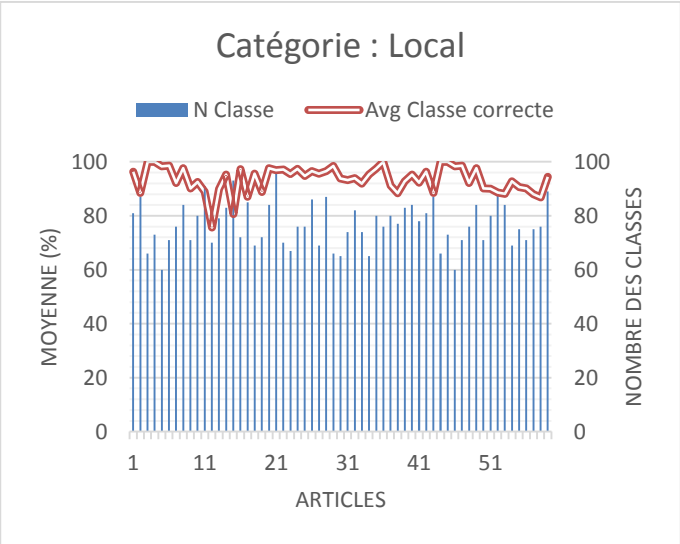

Figure 74 : *Représentation de la Moyenne des classes correctes de la catégorie local*

E. Religion

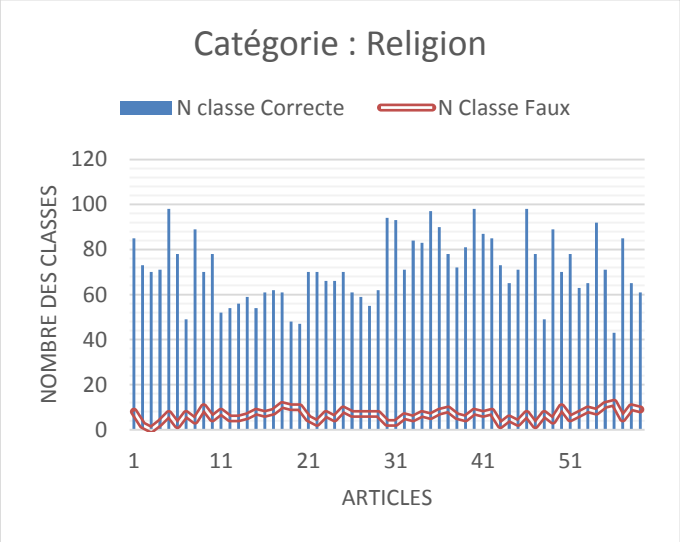

Figure 75 : *Résultats de la catégorie : religion*

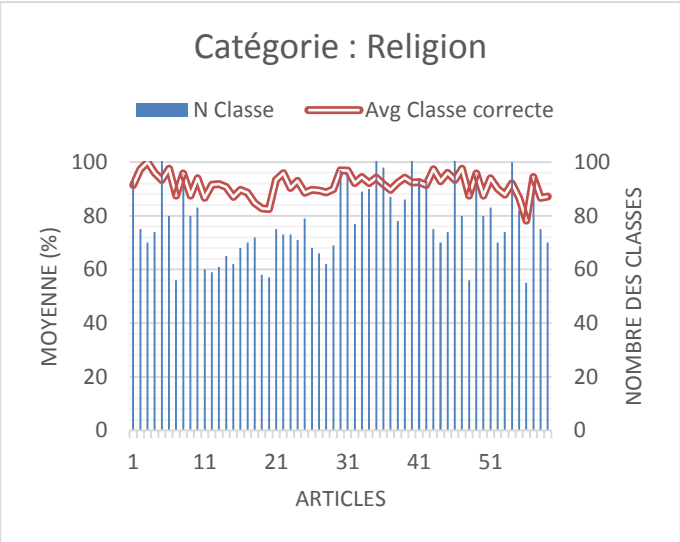

Figure 76 : *Représentation de la Moyenne des classes correctes de la catégorie religion*

F. Sport

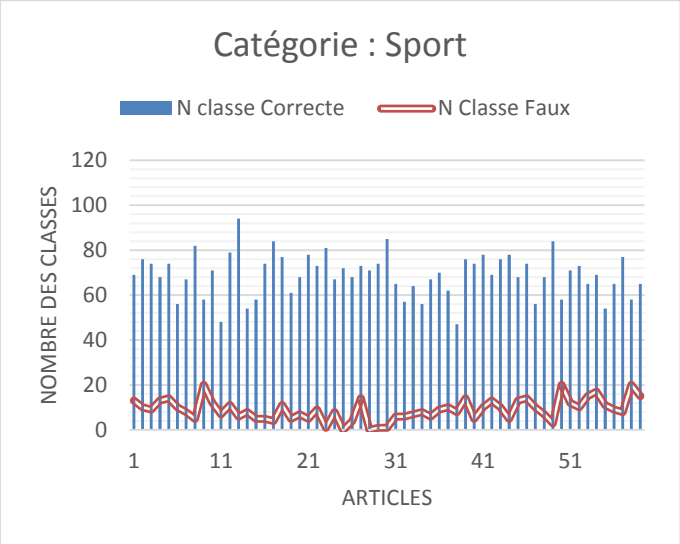

Figure 77 : *Résultats de la catégorie : sport*

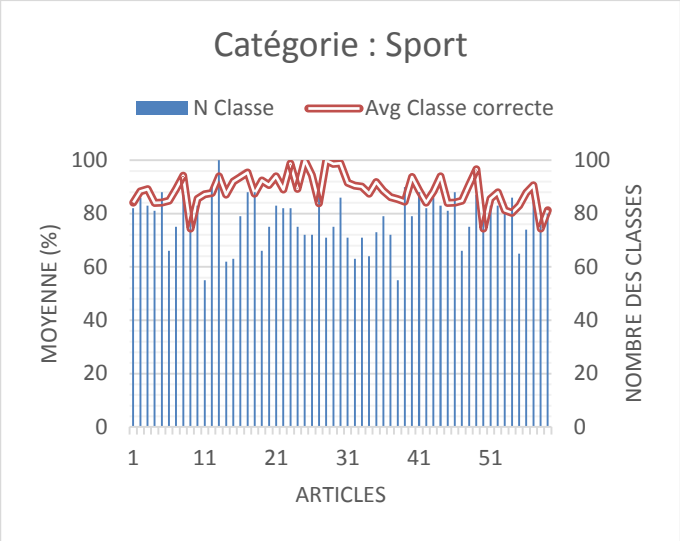

Figure 78 : *Représentation de la Moyenne des classes correctes de la catégorie sport*

6.3 JDiac

Deux types des expériences sont appliqués en utilisant une partie de corpus vocalisé "Tashkeela" sur notre outil. Le corpus a été décomposé sur 4 groupes de fichiers (S1, S2, S3, S4). Nous avons fait deux expériences pour chaque groupe. Dans la premiers nous avons analysé et vocalisé les mots du corpus complètement et dans la deuxièmes nous avons analysé et vocalisé les mots du corpus partiellement (sans la derniers lettre du mot). Puis on a calculé le dégrée de similarité entre le texte vocalisé généré par notre outil et le texte originale de corpus, et nous avons calculé la moyenne des mots identique dans le test.

Corpus	Caractéristiques	Valeur
Tashkeela	Nombre de documents	7
	Taille	9 Mo
	Nombre de catégories	1
	Nombre de mots	~460000
	Nombre de mots dans chaque document	~1000

Table 65 : *Caractéristique du test pour JDiac*

6.3.1 Exemples des tables de résultats

6.3.1.1 Groupe 1 (S1)

A. Expérience 1

Mot complet			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
93.74642665	900	1179	76.33587786
93.65765539	926	1187	78.01179444
92.72210574	862	1158	74.43868739
94.01779013	930	1178	78.94736842
93.26552741	882	1114	79.17414722
94.39295479	965	1229	78.51912124

Table 66 : *Exemple d'une table de résultats* de l'expérience 1 de S1

B. Expérience 2

Mot sans la dernière lettre			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
96.56436708	1029	1109	92.78629396
97.41838443	1029	1094	94.05850091
97.35705141	1062	1127	94.2324756
96.95015516	1096	1165	94.07725322
96.81698934	871	932	93.45493562
96.79345654	862	920	93.69565217

Table 67 : *Exemple d'une table de résultats* de l'expérience 2 de S1

6.3.1.2 Groupe 2 (S2)

A. Expérience 1

Mot complet			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
92.96211892	819	1087	75.3449862
95.16928683	967	1201	80.51623647
94.44168261	915	1154	79.28942808
93.52682233	1022	1300	78.61538462
93.99686501	899	1159	77.56686799
93.58593798	851	1059	80.35882908

Table 68 : *Exemple d'une table de résultats* de l'expérience 1 de S2

B. Expérience 2

Mot sans la dernière lettre			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
94.97717669	966	1087	88.86844526
97.06848795	1120	1201	93.25562032
96.10158156	1062	1154	92.02772964
95.44607006	1189	1300	91.46153846
95.89998745	1053	1159	90.85418464
95.00656489	955	1059	90.17941454

Table 69 : *Exemple d'une table de résultats* de l'expérience 2 de S2

6.3.1.3 Groupe 3 (S3)

A. Expérience 1

Mot complet			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
94.92151858	900	1109	81.15419297
95.86065902	902	1094	82.44972578
95.09175855	889	1127	78.88198758
95.41268046	954	1165	81.88841202
95.28869863	762	932	81.75965665
95.087357	744	920	80.86956522

Table 70 : *Exemple d'une table de résultats* de l'expérience 1 de S3

B. Expérience 2

Mot sans la dernière lettre			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
96.56436708	1029	1109	92.78629396
97.41838443	1029	1094	94.05850091
97.35705141	1062	1127	94.2324756
96.95015516	1096	1165	94.07725322
96.81698934	871	932	93.45493562
96.79345654	862	920	93.69565217

Table 71 : *Exemple d'une table de résultats* de l'expérience 2 de S3

6.3.1.4 Groupe 4 (S4)

A. Expérience 1

Mot complet			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
93.66310456	915	1200	76.25
93.45648457	989	1286	76.90513219
91.88951367	864	1173	73.657289
93.46977547	954	1201	79.43380516
93.72633601	653	835	78.20359281
94.81616119	941	1177	79.94902294

Table 72 : *Exemple d'une table de résultats* de l'expérience 1 de S4

B. Expérience 2

Mot sans la dernière lettre			
Similarité %	Mot 100% Juste	Mots totale	Avg Mot 100% Juste
96.16218541	1106	1200	92.16666667
95.25002271	1166	1286	90.66874028
94.27327842	1024	1173	87.29752771
95.20405787	1082	1201	90.09159034
95.79276704	758	835	90.77844311
96.77446411	1081	1177	91.84367035

Table 73 : *Exemple d'une table de résultats* de l'expérience 2 de S4

6.3.2 Graphes de résultats pour chaque groupe :

Selon les 4 groupes nous avons obtenus les graphes de résultats suivants, nous citons dans chaque graphe les mesures suivants :

- ✓ Le dégrée de similarité (Similarité %)
- ✓ La moyenne des mots identique (Avg mot 100% juste)
- ✓ Le nombre de fichier dans chaque groupe (Numéro de fichier)

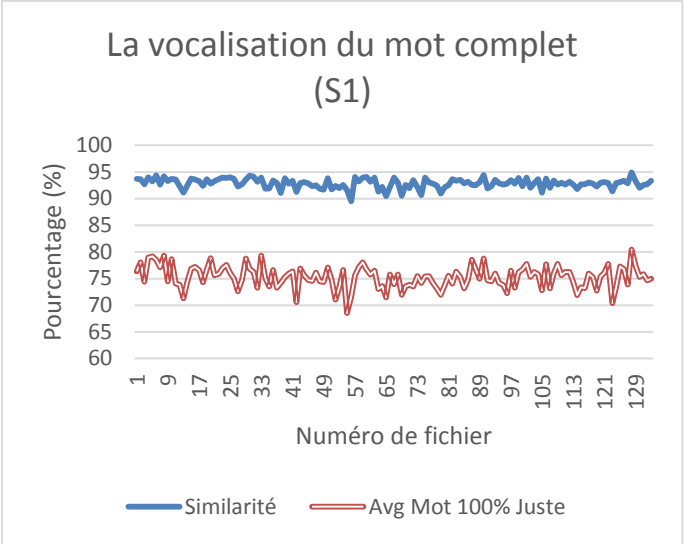

Figure 79 : *Expérience 1 – groupe 1*

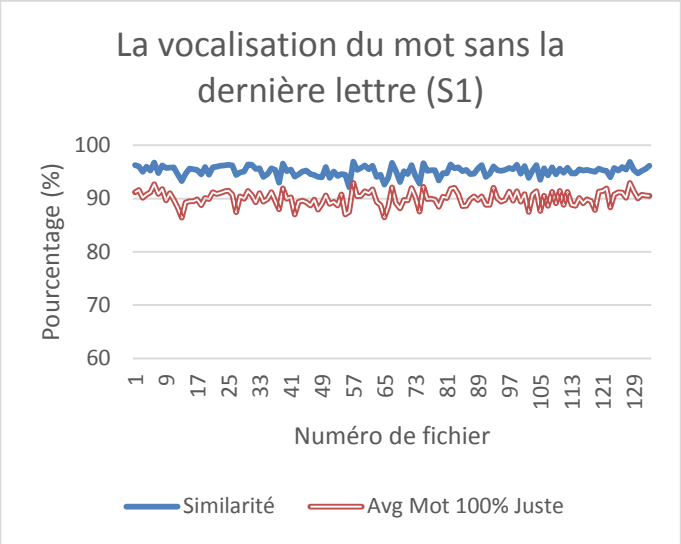

Figure 80 : *Expérience 2 – groupe 1*

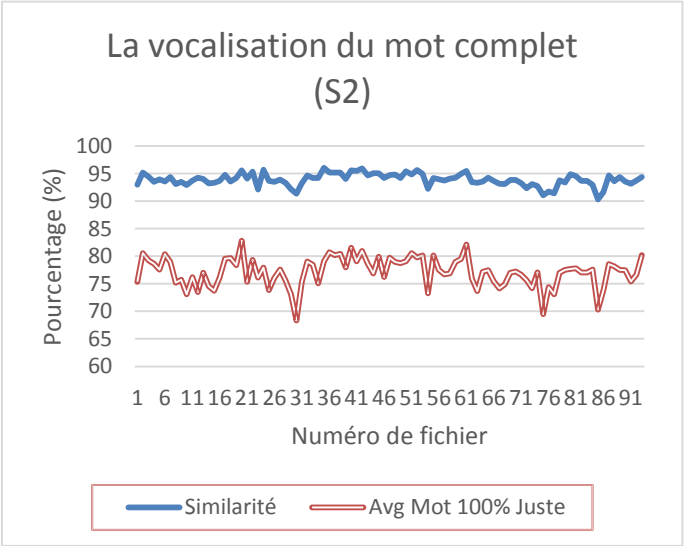

Figure 81 : *Expérience 1 – groupe 2*

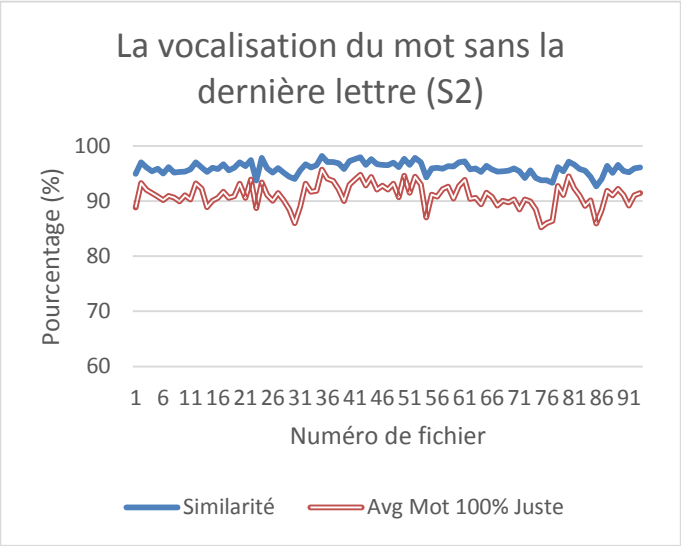

Figure 82 : *Expérience 2 – groupe 2*

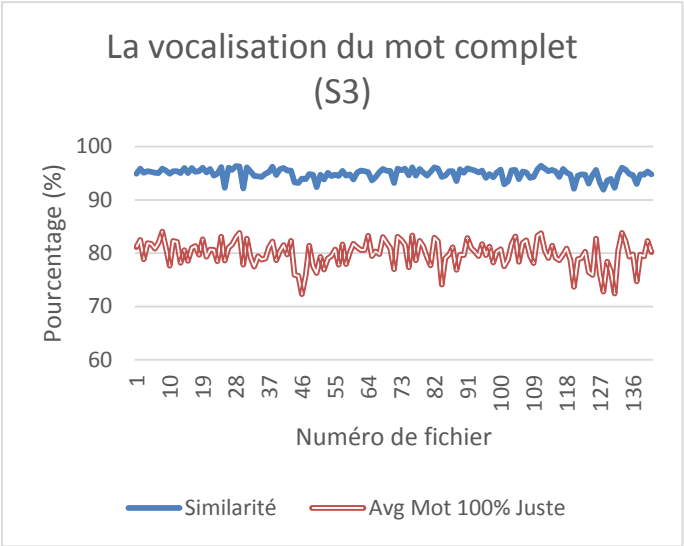

Figure 83 : *Expérience 1 – groupe 3*

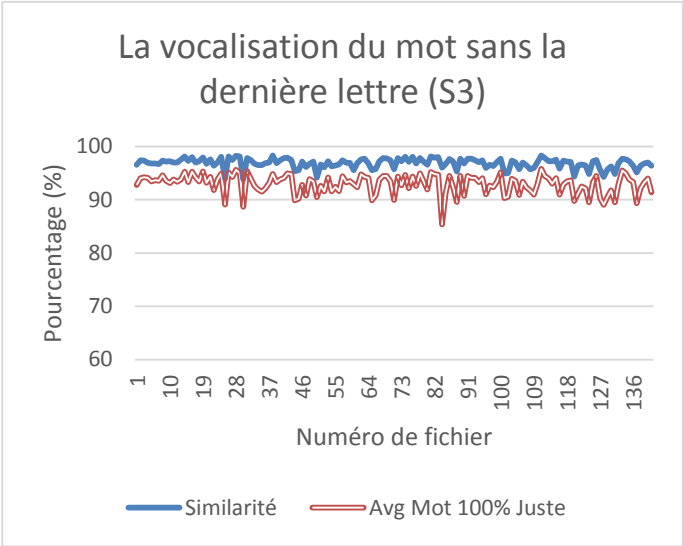

Figure 84 : *Expérience 2 – groupe 3*

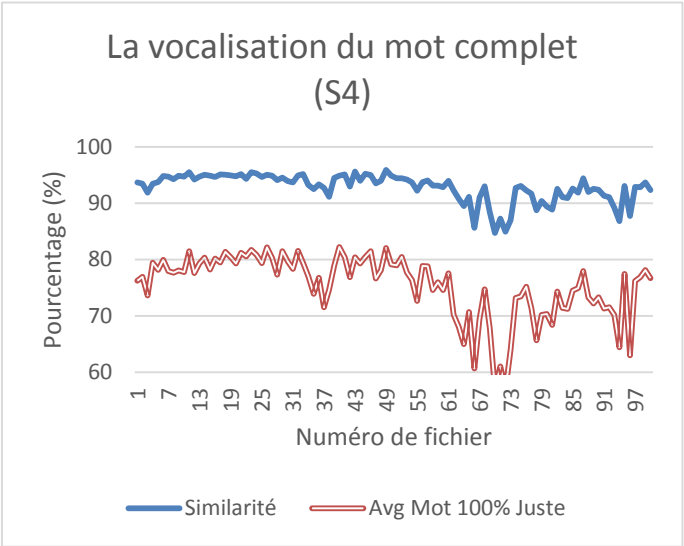

Figure 85 : *Expérience 1 – groupe 4*

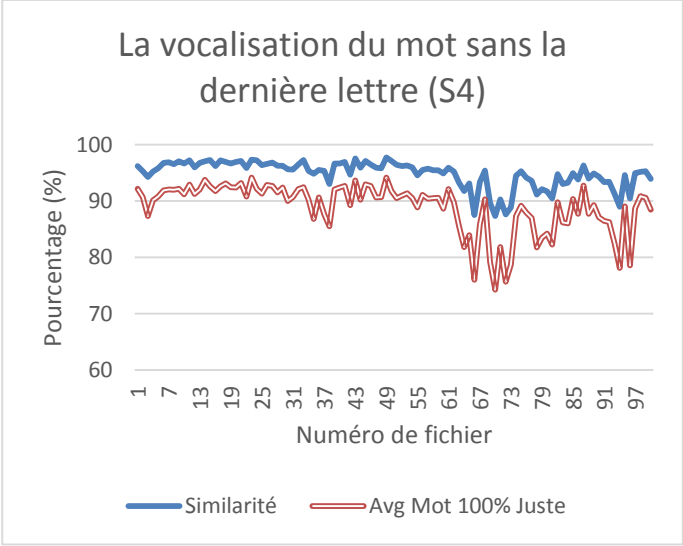

Figure 86 : *Expérience 2 – groupe 4*

6.4 JExtract

Des expériences sont appliquées en utilisant un texte arabe composé de 3517 mots sur notre outil pour l'extraction des informations selon les différentes catégories grammaticales. Les résultats obtenus sont organisé dans le tableau suivant, selon les mots clés :

M : Masculin, **F** : Féminin, **S** : Singulier, **D** : Duel, **P** : Pluriel, **1ère** : 1ère personne, **2ième** : 2ième personne, **3ième** : 3ième personne.

Catégorie grammaticale	Genre	Nombre	Personne	Mode de verbe
Catégorie 1 Les noms	**NOM : 1075 mots / Totale des mots correcte : 97.06 %**			
	M : 31 **F : 344**	**S : 271** **D : 5 /P : 94**	**3ième : 62**	
	Tous les cas possibles des noms			
	M, S : 20 **M, D : 4** **M, P : 6**	**F,S : 257** **F,D : 1** **F,P : 87**	**M, S, 1ère : 20** **M, P, 3ième : 1**	**F,S, 3ième : 48** **F, P, 3ième : 1**
	Adjectif : 223 mots / **Totale des mots correcte : 96 %**			
	M : 4 **F : 103**	**S : 104** **D : 2 /P : 1**	**3ième : 1**	
	Tous les cas possibles des adjectifs			
	M, S : 0 **M, D : 2** **M, P : 1**	**F,S : 103** **F,D : 0** **F,P : 0**	**F,S,3ième : 1**	
	Nom propre : 89 mots / Totale des mots correcte : 97.2 %			
	M : 0 **F : 6**	**S : 4** **D : 0** **P : 2**		
	Pronom : 17 mots / Totale des mots correcte : 98 %			
	M : 6 **F : 4**	**S : 9** **D : 1** **P : 2**	**1ère : 2** **2ième : 1** **3ième : 6**	
	Tous les cas possibles des pronoms			
	M, S : 4 **M, D : 0** **M, P : 2**	**F,S : 3** **F,D : 0** **F,P : 0**	**M,S, 3ième : 2** **M,P, 3ième : 1**	**F,S, 3ième : 2**

Table 74 A : *Exemple d'une table de résultats d'extraction*

<table>
<tr><th>Catégorie grammaticale</th><th>Genre</th><th>Nombre</th><th>Personne</th><th>Mode de verbe</th></tr>
<tr><td rowspan="12">Catégorie 2
Les verbes</td><td colspan="4">verbe : 281 mots / Totale des mots correcte : 96.9 %</td></tr>
<tr><td colspan="4">Accompli : 143 mots / Totale des mots correcte : 94.3 %</td></tr>
<tr><td>M : 1
F : 5</td><td>S : 5
D : 2
P : 1</td><td>1ère : 0
2ième : 0
3ième : 5</td><td></td></tr>
<tr><td colspan="4">Tous les cas possibles des verbes accomplis</td></tr>
<tr><td>M, S : 0
M, D : 2
M, P : 1</td><td colspan="2">F,S : 5
F,D : 0
F,P : 0</td><td>F,S, 3ième : 5</td></tr>
<tr><td colspan="4">Inaccompli : 149 mots / Totale des mots correcte : 93.7 %</td></tr>
<tr><td>M : 10
F : 0</td><td>S : 2
D : 0
P : 11</td><td>1ère : 1
2ième : 38
3ième : 77</td><td>Indicative : 1</td></tr>
<tr><td colspan="4">Tous les cas possibles des verbes inaccomplis</td></tr>
<tr><td>M, S : 1
M, D : 0
M, P : 10</td><td colspan="2">F,S : 0
F,D : 0
F,P : 0</td><td>M,S, 3ième : 1
M,P, 3ième : 10</td></tr>
<tr><td colspan="4">Future : 8 mots / Totale des mots correcte : 100 %</td></tr>
<tr><td colspan="4">Personne : 1ère : 0 / 2ième : 3 / 3ième : 5</td></tr>
<tr><td rowspan="2">Catégorie 3
Les particules</td><td colspan="4">Particules : 102 mots / Totale des mots correcte : 98.7 %</td></tr>
<tr><td colspan="4">Interrogation : 0
Négation : 4
Conjonction : 21
Préposition : 27
Mot de fonction : 4
Exception : 2
Forme courte : 0
Numéro : 35
Ponctuation : 9</td></tr>
</table>

Table 74 B : *Exemple d'une table de résultats d'extraction*

7. Analyse

7.1 JStem

Les figures des différents groupes illustrent la comparaison de la moyenne des mots correctement lemmatisé par rapport au nombre totale des mots pour différentes classes. Les résultats montrent que la moyenne de lemmatisation correcte des mots est **89.7 %**, quelque résultat ayant la moyenne entre 40 % et 60 % dans quelque classe durant l'opération de lemmatisation. L'explication de cette chute brutale dans les résultats en raison de la catégorie grammaticale du mot. C'est-à-dire la plupart des mots dans les classes ayant des résultats faible sont des mots irréguliers (**ex : verbes irréguliers افعال معتلة**).

7.2 JClass

Deux catégories de figure ont été utilisé pour illustré les résultats de cet outil. La premier catégorie contient les figures (67, 69, 71, 73, 75,77) illustrent la comparaison de nombre de classe correctement classé par rapport au nombre de classe mal classé pour différentes articles. La deuxième catégorie contient les figures (68, 70, 72, 74, 76,78) illustrent la comparaison de moyenne de classes correctement classé par rapport au nombre totale de classes pour différentes articles. La moyenne générale des classes correctes est **92.2 %**. La seule catégorie qui a plus la faible moyenne est la catégorie du sport avec une moyenne de **88.54 %**. L'explication de cette moyenne en raison des mots du sport arabisés existant dans les articles de cette catégorie et l'absence de ces derniers de notre base de connaissance lexicale de notre lemmatiseur (JStem).

7.3 JDiac

Pour obtenir des résultats généraux sur notre outil de vocalisation on a appliqué deux expériences. La première expérience est d'analyser le mot vocalisé complètement, les figures (79, 81, 83,85) illustrent la comparaison de moyenne de similarité par rapport à la moyenne des mots identique au mot originale du corpus de test pour différentes fichiers (articles). La moyenne générale de similarité de cette expérience est **93.69 %** et la moyenne des mots identique est **78.91 %**. La deuxième expérience est d'analyser le mot vocalisé partiellement, les figures (80, 82, 84,86) illustrent la comparaison de moyenne de similarité par rapport à la moyenne des mots sans la vocalisation de la dernière lettre pour différentes fichiers (articles). La moyenne générale de similarité de cette expérience est **95.71 %** et la moyenne des mots partiellement vocalisé identique est **90.74 %**. La seule explication pour ces résultats est le manque d'une analyse morphologique pour la vérification et la correction du signe de vocalisation de la dernière lettre du mot dans notre outil.

7.4 JExtract

Le tableau 74 présente un exemple sur les résultats des expériences obtenus avec un texte arabe composé de 3517 mots. Les résultats expriment les différentes catégories grammaticales et la fréquence de ces derniers dans le texte analysé. Les mots répétés sont calculés qu'une seule fois. Le résultat obtenus dans la premier catégorie est de **97.06 %**, pour la deuxième catégorie est de **96.9 %** et pour la troisième catégorie est de **98.7 %**. Le résultat général de ces expériences est de **97.55 %**. L'explication pour ces résultats est l'étiquetage erroné de quelque mot et l'ambiguïté dans la phase de concaténation des mots étiquetés**.**

8. Conclusion

Dans ce chapitre, nous avons présenté l'implémentation de la boite à outils JEEM BOX et ces outils. Ensuite, nous avons appliqué plusieurs expériences pour tester la performance des outils suivants : JStem, JClass, JDiac et JExtract. Les expériences ont montrées de bons résultats pour les différents outils. Cependant, ils manquent un peu d'optimisation pour obtenir des résultats plus performants à l'avenir.

CONCLUSION GÉNÉRALE

Bilan et Perspectives

Conclusion générale

1. Bilan

Dans cette étude, nous avons développé la boite à outil JEEM BOX pour l'acquisition des connaissances à partir d'un texte arabe. Pour ce faire, nous avons organisé notre travail selon trois étapes principales. D'abord, nous avons récolté et préparé toutes les données linguistiques nécessaires : corpus de travail, lexique, jeux d'étiquettes et corpus d'apprentissage. Ensuite, nous avons développé un outil avec une méthode hybride de lemmatisation, d'autre outils ont été développées : outil de vocalisation, outil de translitération, outil de classification, un concordancier et un outil d'extraction de connaissance basant sur les informations morphosyntaxique d'un mot arabe. Enfin, nous avons terminé par une évaluation quantitative et qualitative de notre boite à outil.

D'un point de vue général, le traitement automatique de la langue arabe (et en particulier l'acquisition de connaissance) reste un domaine très ouvert et présente des marges de progression importantes, du fait de la richesse morphologique de cette langue.

Au cours de notre recherche, nous avons fait face à certaines difficultés, et accompli un certain nombre de développement, que l'on peut résumer ainsi :

Comme nous l'avons montré, la lemmatisation, l'une des opérations de base souvent considérée comme triviale dans des langues comme l'anglais ou le français, reste un des problèmes clé de l'arabe, où de grandes améliorations peuvent encore être apportées.

Ensuite, nous avons présenté l'outil de translittération utilisé dans JEEM BOX. Cette translittération est une translittération complète et facile à lire un à un, compatible avec les codages informatiques arabes. Nous espérons que ce système de translittération deviendra une norme à suivre dans le milieu de la recherche de traitement automatique de la langue arabe.

Ainsi nous avons pu à travers cette étude à la fois théorique et pratique, concevoir deux outils de classification automatique de texte arabe, ces outils ont pour principale objectif de classifier le texte arabe en se basant sur la racine du mot ou la fréquence du mot dans le texte. Il faut tout de même dire que la réalisation de ces outils pour la langue arabe peut être bénéfique, cela se traduit par l'intégration de ces applications dans différents produits spécialisés, à titre d'exemples dans notre outils : JStem, JDiac et JExtract.

Les résultats de l'évaluation quantitative montrent un gain de notre méthode hybride de vocalisation par rapport aux autres méthodes. Par contre, nous n'avons pas fait une évaluation qualitative sur la totalité de

notre corpus de test, puisque ce dernier est très grand. Nous avons conclu que JDiac donne plus de performance et particulièrement au niveau des mots arabisés.

Nous avons présenté aussi l'outil d'extraction des informations à partir d'un texte arabe basé sur les informations morphosyntaxiques d'un mot arabe, utilisant un étiqueteur du texte arabe avec une modification concernant le jeu d'étiquette.

Au final, nous avons obtenu de bons résultats au niveau de lemmatisation, classification, vocalisation et extraction comme la montré l'évaluation qualitative et quantitative effectuée sur notre boite à outils. C'est pourquoi, on peut conclure que nous avons atteint notre objectif initial à savoir offrir aux chercheurs une boite à la fois polyvalente et performante.

2. Perspectives

Plusieurs perspectives citées dans cette section peuvent amener des améliorations qui rendent efficace les performances de notre boite à outil JEEM BOX. On peut en citer quelques-unes :

- Enrichir la base de connaissance lexicale avec plus de ressources Schèmes, Racines, mots outils, mots spéciaux, dont l'optique de toucher le maximum de catégories de la langue.
- Amélioration de la segmentation des mots ambigus, c'est-à-dire les mots qui admettent plusieurs découpages différents. La correction de la segmentation peut être effectuée de façon manuelle et semi-automatique, au moyen d'un corpus pré-segmenté.
- Etablir des règles concernant la lemmatisation des cas particulier, comme : verbes faibles (assimilé, concave, lafif, défectueux), verbes doublés et le pluriel brisé.
- Optimisation de la classification pour traiter plusieurs fichiers à la fois, et l'utilisation des algorithmes de recherche plus rapide dans la classification comme par exemple l'algorithme de Boyer-Moore.
- Il est intéressant de travailler sur un analyseur syntaxique pour optimiser la fonction de vocalisation de la dernière lettre du mot dans l'outil JDiac. La technique d'utilisation d'un analyseur syntaxique dans la vocalisation a été déjà expliquée dans le chapitre 3.
- Amélioration du corpus d'apprentissage de l'outil JDiac, il faut que notre corpus d'apprentissage soient riche et consistant (notre corpus doit être très varié : accueillir des phrases de tous les domaines : religion, art, éducative, littéraire…. etc.)

BIBLIOGRAPHIE

Bibliographie

[1]	Jean-Marie Pierrel. Ingénierie des Langues. Hermès, Paris, 2000.
[2]	Delafosse L. 1999. Glossaire de Linguistique Computationnelle.
[3]	J. LEON, « Le traitement automatique des langues », CNRS, Université Paris 7, 2001 (France).
[4]	F. YVON, « Une petite introduction au traitement Automatique du langage naturel », support de cours, Ecole Nationale Supérieur des télécommunications, Avril 2007.
[5]	P. BOUILLON, « Traitement automatique des langues naturelles », édition Duculot, 1998. (France).
[6]	A. Martinet. Qu'est-ce que la morphologie ? Cahiers F er dinand de Saussure, 26 :85-90, 1969.
[7]	F. Neveu. Dictionnaire des sciences du langage. Armand Colin, 2004.
[8]	Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL recognising textual entailment challenge.In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.
[9]	D. MAINGUENEAU, « Aborder la linguistique », édition Seuil, 1996 (France).
[10]	F. YVON, « Introduction au Traitement Automatique des Langues Naturelles », support de cours, 2006.
[11]	Aljlayl.M and Frieder.O. (2002). On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemmng Approach. In 11th International Conference on Information and Knowledge Management (CIKM), 340-347.
[12]	Baloul.S,Alissali.M, Baudry.M and Boula de Mareuil.P. (2002). Interface syntaxe-prosodie dans un système de synthèse de la parole à partir d'un texte en arabe. 24es Journnées d'étude sur la parole, 329-332.
[13]	Chaâben N., Belguith L., ″Implémentation du système MORPH2 analyse morphologique pour l'arabe non voyellé″, GEI'04, Monastir, Tunisie, 2004.
[14]	Mars M, Zrigui M, Belgacem M,Zouaghi A, Antoniadis G. "A Semantic Analyzer for the Comprehension of the Spontaneous Arabic Speech", 9th International Conference on Computing CORE08, Journal Research in Computing Science (Journal RCS), ISSN: 1870-4069, Vol 34, pp 129-140, CORE0, Mexico. 2008.
[15]	Abdelwahed A.,"بنية الفعل قراءة في التصريف العربي", 1996 ,كلية الآداب و العلوم الإنسانية بصفاقص، تونس.
[16]	Chaari F., Gargouri B., Jmaiel M., "Vers une interface logicielle pour l'exploitation d'une base lexicale normalisée par les applications du TALN : cas de la morphologie de l'arabe", GEI'06, Hammamet, Tunisie, 2006.
[17]	Abbes, Ramzi. (décembre 2004). la conception et la réalisation d'un concordancier électronique pour l'arabe. Thèse de doctorat en sciences de l'information, Lyon, ENSSIB/INSA.
[18]	El-dahdeh A., "معجم قواعد اللغة العربية في جداول ولوحات" , مكتبة لبنان بيروت , لبنان ,1999.
[19]	El-dahdeh A., "معجم تصريف الأفعال العربية العربية في جداول ولوحات" , مكتبة لبنان بيروت , لبنان ,1999.
[20]	Blachère R., Gaudefroy-Demombynes M., ″Grammaire de l'arabe classique″, Edition Maisonneuvre-Larose, Paris, 1975.
[21]	E. DITTERS. The description of modern standard arabic syntax in terms of functions and cate-

	gories. Langues et Littératures du Monde Arabe, 2:115–151, 2001.
[22]	Lamia Hadrich Belguith, Chafik Aloulou, MASPAR : De la segmentation à l'analyse syntaxique de textes arabes, Laboratoire de Recherche LARIS – MIRACL, Faculté des Sciences Economiques et de Gestion de Sfax B.P. 1088, 3018 - Sfax – TUNISIE.
[23]	E. Souissi, Etiquetage grammatical de l'arabe voyellé ou non, Thèse de doctorat, Université Paris VII, 1997.
[24]	J. DICHY. Pour une lexicomatique de l'arabe : l'unité lexicale simple et l'inventaire fini des spécificateurs du domaine du mot. Meta, XLII, 2:291–306, 1997.
[25]	L.S. LARKEY, L. BALLESTEROS et M.E. CONNELL. Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In Proceedings of the 25th annual interna-tional ACM SIGIR conference on Research and development in information retrieval, pages 275–282, Tampere, Finland, 2002.
[26]	S. KHOJA ET G. GARSUDE, R. AND KNOWLES. A tag set for the morph syntactic tagging of Arabic. In Corpus Linguistics 2001 conference, pages 1–13, Lancaster, UK, 2001.
[27]	Taghva, K., Elkoury, R., and Coombs, J. 2005. Arabic Stemming without a root dictionary.Information Science Research Institute. University of Nevada, Las Vegas, USA.
[28]	Al-Fedaghi, S.S., and Al-Anzi, F.S. A New Algorithm to Generate Arabic Root-Pattern Forms, Proceedingsofthe11th National Computer Conference and Exhibition, March, Dhahran, Saudi Arabia, pp.391-400, 1989.
[29]	Al-Shalabi R. and M. Evens. "A computational morphology system for Arabic". In Workshop on Computational Approaches to Semitic Languages, COLING-ACL98. August 1998.
[30]	R. BESANÇON. Intégration de connaissances syntaxiques et sémantiques dans les représentations vectorielles des textes, Application au calcul de similarité sémantique dans le cadre du modèle DSIR. Thèse de Doctorat, Ecole polytechnique fédérale de Lausanne, Suisse, 2001.
[31]	M. Diab, K. Hacioglu, et D. Jurafsky. Automatic tagging of arabic text: From raw text to base phrase chunks. In Proceedings of HLT-NAACL 2004, pages 149-152, Boston, 2004.
[32]	Z. Zemirli, S. Khabet, TAGGAR : Un analyseur morphosyntaxique destiné à la synthèse vocale de textes arabes voyellés. JEP-TALN 2004, Traitement Automatique de l'Arabe, Fès, 20 avril 2004
[33]	L. ROMARY. Outils d'accès à des ressources linguistiques. Ingénierie des langues, pages 193–212, 2000.
[35]	E. LAPORTE. Mot et niveau lexical. Ingénierie des langues, pages 25–46, 2000.
[36]	M. Abbas, K. Smaili. Comparison of Topic Identification Methods for Arabic Language, International conference RANLP05: Recent Advances in Natural Language Processing, 21-23 september 2005, Borovets, Bulgary.].
[37]	D. W. Oard and F. C. Gey. The TREC-2002 Arabic/English CLIR Track. In Proceedings of the Text Retrieval Conference (TREC-11), pages 17-26, NIST,Gaithersburg, MD., 2002.
[38]	Zajac, Rémi, Malki, Ahmed, Abdelali, Ahmed, Cowie, James, Ogden William C. 2001. Arabic-English NLP at CRL, Proceedings of the Arabic NLP Workshop ACL/EACL 2001.
[39]	Andries, Patrick. Unicode 5.0 en pratique: codage des caractères et internationalisation des logiciels et des documents, Paris, Dunod, 2008, 399 p.

[40]	Kontorovich,Leonid and Daniel D. Lee (2001) Learning Semitic languages with Hidden Markov Models, NIPS 2001 Workshop on Machine Learning Methods for Text and Images.
[41]	Lechel M. : Analyse et conception d'un correcteur grammatical libre pour le français, thèse de magister, université de Gronoble 3, (2005).
[42]	Fatma Al Shamsi, Ahmed Guessoum: A Hidden Markov Model –Based POS Tagger for Arabic, University of Sharjah 2005.
[43]	Tuerlinckx L.: La lemmatization de l'arabe non classique, U.C.L. Institut orientaliste, Belgique.

ANNEXE

ANNEXE A

Transcription de Buckwalter

UNICODE			BUCKWALTER	
Decimal	Hex	Glyph	ASCII	Orthography
1569	U+0621	ء	'	Hamza
1571	U+0623	أ	>	Alif + HamzaAbove
1572	U+0624	ؤ	&	Waw + HamzaAbove
1573	U+0625	إ	<	Alif + HamzaBelow
1574	U+0626	ئ	}	Ya + HamzaAbove
1575	U+0627	ا	A	Alif
1576	U+0628	ب	b	Ba
1577	U+0629	ة	p	TaMarbuta
1578	U+062A	ت	t	Ta
1579	U+062B	ث	v	Tha
1580	U+062C	ج	j	Jeem
1581	U+062D	ح	H	HHa
1582	U+062E	خ	x	Kha
1583	U+062F	د	d	Dal
1584	U+0630	ذ	*	Thal
1585	U+0631	ر	r	Ra
1586	U+0632	ز	z	Zain
1587	U+0633	س	s	Seen
1588	U+0634	ش	$	Sheen
1589	U+0635	ص	S	Sad
1590	U+0636	ض	D	DDad
1591	U+0637	ط	T	TTa
1592	U+0638	ظ	Z	DTha
1593	U+0639	ع	E	Ain
1594	U+063A	غ	g	Ghain
1600	U+0640	-	_	Tatweel

Table 75 A: *Transcription de buckwalter*

UNICODE			BUCKWALTER	
Decimal	Hex	Glyph	ASCII	Orthography
1601	U+0641	ف	f	Fa
1602	U+0642	ق	q	Qaf
1603	U+0643	ك	k	Kaf
1604	U+0644	ل	l	Lam
1605	U+0645	م	m	Meem
1606	U+0646	ن	n	Noon
1607	U+0647	ه	h	Ha
1608	U+0648	و	w	Waw
1609	U+0649	ى	Y	AlifMaksura
1610	U+064A	ي	y	Ya
1611	U+064B	ً	F	Fathatan
1612	U+064C	ٌ	N	Dammatan
1613	U+064D	ٍ	K	Kasratan
1614	U+064E	َ	a	Fatha
1615	U+064F	ُ	u	Damma
1616	U+0650	ِ	i	Kasra
1617	U+0651	ّ	~	Shadda
1618	U+0652	ْ	o	Sukun
1619	U+0653	ٓ	^	Maddah
1620	U+0654	ٔ	#	HamzaAbove
1648	U+0670	ٰ	`	AlifKhanjareeya
1649	U+0671	ٱ	{	Alif + HamzatWasl
1756	U+06DC	ۜ	:	SmallHighSeen
1759	U+06DF	۟	@	SmallHighRoundedZero
1760	U+06E0	۠	"	SmallHighUprightRectangularZero
1762	U+06E2	ۢ	[	SmallHighMeemIsolatedForm
1763	U+06E3	ۣ	;	SmallLowSeen
1765	U+06E5	ۥ	,	SmallWaw
1766	U+06E6	ۦ	.	SmallYa
1768	U+06E8	ۨ	!	SmallHighNoon
1770	U+06EA	۪	-	EmptyCentreLowStop
1771	U+06EB	۫	+	EmptyCentreHighStop
1772	U+06EC	۬	%	RoundedHighStopWithFilledCentre
1773	U+06ED	ۭ	]	SmallLowMeem

Table 75 B: *Transcription de buckwalter*

ANNEXE B

Codification des consonnes arabes

Par le standard Unicode

No	Valeurs Unicode	Caractères arabes	No	Caractères arabes	Valeurs Unicode
1	U+0621	*	16	ض	**U+0636**
2	U+0627	أ	17	ط	**U+0637**
3	U+0628	ب	18	ظ	**U+0638**
4	U+062A	ت	19	ع	**U+0639**
5	U+062B	ث	20	غ	**U+064A**
6	U+062C	ج	21	ف	**U+0641**
7	U+062D	ح	22	ق	**U+0642**
8	U+062E	خ	23	ك	**U+0643**
9	U+062F	د	24	ل	**U+0644**
10	U+0630	ذ	25	م	**U+0645**
11	U+0631	ر	26	ن	**U+0646**
12	U+0632	ز	27	ه	**U+0647**
13	U+0633	س	28	و	**U+0648**
14	U+0634	ش	29	ي	**U+064A**
15	**U+0635**	**ص**			

Table 76 : *Codification des consonnes arabes par le standard Unicode*

ANNEXE C

Fréquences d'occurrence des préfixes sur les mots de la collection «Al-Khat Alakhdar»

Préfixe	Fréquence	Préfixe	Fréquence	Préfixe	Fréquence	Préfixe	Fréquence
ا	13324	الن	334	اي	118	متن	22
و	10232	يت	322	ات	105	ولت	22
ال	9965	فال	313	مي	102	وسا	21
وا	4475	ول	311	اك	99	متم	17
ب	4040	است	298	الي	92	ولك	17
ل	3821	من	291	لن	84	افت	15
م	3344	مس	288	لو	80	اسي	12
وال	3315	مت	277	لب	73	قلا	11
ت	3167	ه	269	ولل	71	لال	11
ي	2040	ون	243	مه	64	وسن	10
ف	1491	سي	233	وسي	64	افا	8
لل	1417	في	192	مم	61	فسي	8
الا	1323	فت	188	قل	59	ولن	8
بال	1257	لي	184	الال	52	فست	7
وت	1233	ست	177	لك	52	فسا	6
ك	987	كال	172	ولا	52	فلي	6
ن	930	ام	162	سن	49	اا	5
الت	816	يس	160	وبال	48	اسب	5
وي	628	ين	148	وست	47	فبال	4
لا	477	اب	135	فن	33	مته	4
فا	466	وك	134	ايا	29	متي	4
ان	432	مست	124	فب	29	وكال	4
لت	429	سا	122	فك	24	ولب	4
وب	360	تم	119	ولي	23	افن	3

Table 78 : *Fréquence d'occurrence des préfixes sur les mots de la collection «Al-Khat Alakhdar»*

ANNEXE D

Fréquences d'occurrence des préfixes sur les mots de la collection «Al-Khat Alakhdar»

Suffixe	Fréquence	Suffixe	Fréquence	Suffixe	Fréquence	Suffixe	Fréquence
ه	10550	نها	196	وك	38	اتك	9
ا	6900	تى	139	هن	35	اهما	8
ت	4660	اته	138	ونى	35	موه	8
ن	3898	وه	137	اتى	34	تانى	8
ى	3297	انى	131	ينى	34	تكم	7
ها	3199	اك	124	نك	30	موها	7
ات	3128	تان	114	الم	29	ينهم	7
يه	2799	تا	109	ناه	29	كما	7
م	2507	اها	108	ننا	27	يكم	6
ات	936	يك	107	اتنا	26	انك	6
ون	936	اتهم	85	ونا	24	ننى	5
هم	803	تنا	83	ينها	24	تاك	5
تها	752	ينا	75	تن	19	تهن	4
ك	648	ينه	75	اكن	16	مانا	4
ته	621	ونه	73	تهما	16	مونى	4
نا	598	انا	61	وهم	16	ونهم	4
و	494	تيه	60	يهما	16	اتكم	4
نه	441	انها	54	مانى	15	اننى	3
وا	436	نهم	54	ناها	14	انهما	3
نى	367	تم	53	تيك	13	اهن	3
اه	286	يهم	52	تنى	12	تاه	3
اتها	232	ونها	50	اننا	10	تينا	3
تهم	225	اى	48	ماه	9	اتهما	2
يها	225	وها	40	نهما	9	تاهما	1

Table 79 : *Fréquence d'occurrence des suffixes sur les mots de la collection «Al-Khat Alakhdar»*