

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE AKLI MOHAND OULHADJ-BOUIRA



Faculté des Sciences et des Sciences Appliquées
Département d'informatique

Mémoire de fin d'étude

Présenté par :

FELLAK Samia

SAIDI Yasmine

En vue de l'obtention du diplôme de **Master 02** en :

Filière : **INFORMATIQUE**

Option : Ingénierie des systèmes d'information et logiciel

Thème :

***Modélisation des erreurs grammaticales communes pour le support
la langue arabe pour le logiciel open source « Language Tool »***

Devant le jury composé de :

BOUSSAIDI Smail	Docteur	UAMOB	Président
ZERROUKI Taha	Docteur	UAMOB	Encadreur
DJOUABRI Abdrezek	Docteur	UAMOB	Examineur
BOUCHEREB Sarrah	Doctorante	UAMOB	Examineur

Année Universitaire 2018/2019

Remerciements

Avant tout, nous remercions ALLAH tout puissant, pour la force, le courage et la patience qu'il nous a donnée durant toutes les années d'étude, et la volonté d'arriver au terme de ce travail.

Nous remercions infiniment nos Parents pour tout ce qu'ils ont fait et ce qu'ils font encore pour nous.

Nous exprimons notre grande reconnaissance et remerciement à notre promoteur MR.ZERROUKI Taha pour leur aide, leur conseils et leur orientation et suivi durant notre travail. Nous remercions les membres de jury d'avoir accepté de juger notre travail.

Un grand merci à toutes nos familles, et nos amies pour leur présence, leur préoccupation et leurs encouragements et leur suivi, avec patience, du déroulement de notre projet.

En fin, nous ne pouvons achever ce projet sans exprimer notre gratitude, à tous les enseignants de l'Université de Bouira pour leur dévouement et leur assistance tout au long de cette année.

Dédicaces

Je dédie ce travail à Mes chers parents

Mon père et Ma mère

Pour leur patience, leur amour, leur soutien, et leurs encouragements et le respect pour ses grands sacrifices

A mes soeurs, mes frères, mes grands-parents

et tous les membres de la famille SAIDI

A ma chère binôme Samia

A tous mes chères amis et collègues de la classe .

A tous ceux qui me connaissent et qui m'aiment surtout.

SAIDI Yasmine

Dédicaces

Je dédie ce travail à Mes chers parents

Mon père et Ma mère

*Pour leur patience, leur amour, leur soutien, et leurs encouragements et le
respect pour ses grands sacrifices*

A mes soeurs, mes frères

*et tous les membres de la famille **FELLAK***

A ma chère binôme Yasmine

A tous mes chères amis et collègues de la classe .

A tous ceux qui me connaissent et qui m'aiment surtout.

FELLAK Samia.

Table des matières

Table des matières	i
Table des figures	iv
Liste des tableaux	vi
Liste des abréviations	vii
Introduction générale	1
1 La correction automatique	3
1.1 Introduction	3
1.2 Le traitement automatique de langage (TAL) :	3
1.3 Les niveaux de traitement en TAL	4
1.4 La vérification grammaticale	5
1.4.1 Définition de grammaire :	5
1.4.2 Définition d'erreur de grammaire :	5
1.4.3 Définition de la correction grammaticale :	5
1.4.4 Historique	5
1.4.5 Définition de correcteur grammaticale	6
1.4.6 L'importance de la correction grammaticale	6
1.4.7 Les fonctionnalités d'un correcteur grammatical	7
1.4.8 Architecture des correcteurs grammaticaux	7
1.5 Les approches principales d'un vérificateur de grammaire	10
1.5.1 Vérification basée sur le Machine Learning (ML) :	11

1.5.2	La vérification basée sur les statistiques :	11
1.5.3	Vérification basée sur des règles	11
1.6	Les vérificateurs grammaticaux existants pour la langue arabe :	12
1.6.1	Microsoft Word :	12
1.6.2	Arabic GramCheck :	14
1.6.3	Lightproof :	14
1.6.4	LanguageTool	15
1.7	Discussions et comparaison entre les vérificateurs grammaticaux :	16
1.8	Conclusion	17
2	Les aspects des erreurs en langue arabe	18
2.1	Introduction	18
2.2	Langue arabe	18
2.3	les caractéristiques de la langue	19
2.3.1	Structure d'un mot :	19
2.3.2	Catégories des mots	20
2.4	Les branches de grammaire pour la langue arabe	22
2.4.1	La morphologie	23
2.4.2	La syntaxe	24
2.5	Analyse des erreurs en langue arabe	27
2.6	Conclusion	28
3	La conception	29
3.1	Introduction	29
3.2	Choix de technique	29
3.3	Description générale	30
3.4	Classification des erreurs de grammaire	31
3.4.1	Erreurs d'accord de genre et de nombre :	32
3.4.2	Erreurs de verbe transitifs :	38
3.4.3	Erreurs lors de la liaison :	40
3.4.4	Erreur de pluriels :	41
3.4.5	Erreurs de (EL NAWASIKH) :	42
3.4.6	Erreurs de prononciation :	42

3.4.7	Erreurs sémantiques :	42
3.5	Conclusion	43
4	L'implémentation	44
4.1	Introduction	44
4.2	Implémentation de l'approche	44
4.2.1	Choix du logiciel	44
4.3	L'implémentation des règles	45
4.4	Développement des règles d'erreurs	48
4.4.1	L'accord de genre et nombre	48
4.4.2	Le verbe transitif	53
4.4.3	L'erreur de pluriel	55
4.4.4	L'erreur sémantiques	56
4.4.5	L'erreur de prononciation	57
4.5	Conclusion	58
	Conclusion générale	60
	Bibliographie	62
	Annexe 1	64
	Annexe 2	65

Table des figures

1.1	Structure générale des correcteurs grammaticaux [Souque A, 2014].	8
1.2	La correction grammaticale avec Microsoft Word	13
2.1	Exemple d’ambiguïté morphologique [Cheragui M, 2012].	23
2.2	Exemple de phrase verbale.	24
2.3	Exemple de phrase nominale.	25
3.1	La structure de LanguageTool.	30
3.2	Prototype des erreurs.	32
3.3	L’accord entre le sujet et le verbe.	33
3.4	Exemple d’erreur entre le sujet et le verbe	33
3.5	Exemple d’erreur d’un pronom démonstratif	36
3.6	les catégories de nombre et l’énuméré	37
3.7	Les types de verbe transitifs	38
3.8	Exemple d’erreur	39
3.9	Exemple d’erreur de pluriel	41
4.1	Les éléments de base d’une règle en XML.	45
4.2	Exemple d’implémentation d’un pronom démonstratif.	48
4.3	Résultat d’un pronom démonstratif.	49
4.4	Exemple d’implémentation d’un pronom relatif.	50
4.5	Résultat d’implémentation d’un pronom relatif.	50
4.6	Exemple d’accord d’adjectif en XML.	51
4.7	Résultat d’accord de l’adjectif.	51

4.8	Exemple d'implémentation de nombre et l'énuméré en XML.	52
4.9	Résultat d'implémentation de nombre et l'énuméré.	53
4.10	Exemple d'implémentation d'un verbe transitif.	54
4.11	Résultat d'implémentation d'un verbe transitif.	54
4.12	Résultat d'implémentation de l'exemple de pluriel	55
4.13	Exemple d'implémentation d'erreur sémantique	56
4.14	Résultat d'implémentation d'erreur sémantique	57
4.15	Exemple d'implémentation d'erreur de prononciation.	58
4.16	Résultat d'implémentation d'erreur de prononciation.	58

Liste des tableaux

1.1	Exemples testé sous Microsoft Word [Shaalan K, 2005]	13
1.2	Exemple des erreurs testées en Light Proof [4].	15
1.3	La comparaison entre les vérificateurs grammaticaux	16
2.1	Structure d'un mot [Araibi et Abdelmoumen, 2016]	20
2.2	Les erreurs grammaticales communes de l'Arabe [Shaalan K, 2005].	28
3.1	Exemple d'accord d'adjectif	34
3.2	Exemple d'un pronom relatif	34
3.3	Exemple d'erreur d'un pronom démonstratif [Adnani M, 1983]	35
3.4	Exemple de règles de nombre et l'énuméré [Abou obeida, 2007].	38
3.5	Exemple de verbe transitif (harf jar) [Mahmoud I,1998]	39
3.6	Exemple de verbe transitif (bi nafsih) [Mahmoud I,1998]	40
3.7	Exemple d'erreurs lors de traduction [10]	40
3.8	Exemple d'erreurs de pluriels [Adnani M, 1983]	41
3.9	Exemple d'erreurs (EL NAWASIKH)	42
3.10	Exemple d'erreurs de la prononciation [Adnani M, 1983]	42
3.11	Exemple d'erreurs sémantiques [10]	43

Liste des abréviations

Ar	Arabe
LT	Language Tool
MSA	Modern Standard Arab
TAL	Traitement Automatique de la langue
TALA	Traitement Automatique de la Langue Arabe
TALN	Traitement Automatique de la Langue Naturelle
UTF-8	Unicode Transformation Format 8 bits
XML	eXtended Markup Language
POS	Part-of-Speech
Regex	expression Régulière
Adj	Adjective
Det	Determinant
NFS	Nom Féminin Singulier
NMS	Nom Masculin Singulier
NFP	Nom Féminin Pluriel
NMP	Nom Masculin Pluriel
DFS	Déterminant Féminin Singulier
DMS	Déterminant Masculin Singulier
DFP	Déterminant Féminin Pluriel
DMP	Déterminant Masculin Pluriel

Introduction générale

Le but de ce mémoire est de développer un vérificateur de style et de grammaire Open Source pour la langue arabe. Bien que tous les principaux logiciels de traitement de texte Open Source offrent une vérification orthographique, aucun d'entre eux n'offre une fonction de vérification du style et de la grammaire. Une telle fonctionnalité n'est pas disponible en tant que programme gratuit séparé non plus. Ainsi, le résultat de ce mémoire sera un programme libre qui peut être utilisé à la fois comme vérificateur de style et de grammaire autonome et comme partie intégrante d'un logiciel de traitement de texte.

Le vérificateur de style et de grammaire décrit dans ce mémoire prend un texte et renvoie une liste des erreurs possibles. Pour détecter les erreurs, on attribue à chaque mot du texte son étiquette de partie de parole et chaque phrase est scindée en morceaux, par ex : phrases nominales. Ensuite, le texte est comparé à toutes les règles d'erreur prédéfinies du vérificateur. Si une règle correspond, le texte est supposé contenir une erreur à la position de la correspondance. Les règles décrivent les erreurs sous forme de motifs de mots, de balises de partie de parole et de morceaux. Chaque règle inclut également une explication de l'erreur, qui est montrée à l'utilisateur.

Le système de règles sera rendu plus puissant afin qu'il puisse être utilisé pour exprimer des règles décrivant des erreurs au niveau de la phrase, pas seulement au niveau du mot. L'intégration dans les traitements de texte sera améliorée afin que les erreurs puissent être détectées, c'est-à-dire pendant la saisie de texte. Pour de nombreuses erreurs, le logiciel proposera une correction pouvant être utilisée.

L'approche système basée sur des règles est assez simple pour permettre aux utilisateurs

d'écrire leurs propres règles, mais elle est assez puissante pour détecter de nombreuses erreurs typiques. La plupart des règles sont exprimées dans un format XML simple, qui décrit non seulement les erreurs, mais contient également un message d'erreur utile et des exemples de phrases.

Organisation du mémoire :

Ce travail est organisé d'une introduction générale, et de deux chapitres de l'état de l'art et la partie conception et réalisation, et d'une conclusion générale dans laquelle nous présentons les principales conclusions ainsi que les perspectives de nos travaux.

Chapitre I : Le premier chapitre est réparti en cinq sections. La première est destinée à définir le traitement automatique de la langue (TAL) et ses applications. La seconde étudie brièvement la vérification grammaticale par les définitions . La troisième section est consacrée à présenter l'historique et l'architecture des correcteurs grammaticaux qui fournit un banc d'essai pour l'évaluation de notre système de reconnaissance.

la quatrième sections est consacré par les approches principales d'un vérificateur de grammaire, et on termine par la comparaison entre les différentes correcteurs existants pour la langue arabe .

Chapitre II : Dans le deuxième chapitre nous étudions brièvement la langue arabe comme son origine et ses caractéristiques...etc, ensuite nous détaillons la grammaire de la langue arabe et nous terminons par analyse des erreurs grammaticales communes en langue arabe.

Chapitre III : Nous présentons notre approche par une conception. Nous commençons par présenter le choix de technique ensuite la description générale de notre système. Nous donnons ensuite la démarche de correction de différentes catégories de cette conception.

Chapitre VI : Nous présentons l'implémentation de notre approche après avoir défini l'environnement de développement et l'outil utilisé. Ce chapitre comporte aussi les résultats des tests effectués afin de l'évaluer .

La correction automatique

1.1 Introduction

Le traitement automatique du langage naturel (TALN) regroupe l'ensemble des recherches et développements visant à modéliser à l'aide de machines.

Le but du TAL est réellement de “comprendre” le sens des phrases, les idées qui s'en dégagent de manière à pouvoir “traiter” de la manière la plus optimale et la plus naturelle d'un point de vue humain. Parmi les applications de TAL on trouve la correction grammaticale qui est l'une de ses technologies les plus utilisées du grand public.

Dans ce premier chapitre introductif, nous présentons quelques notions de TAL puis nous passons à la vérification grammaticale avec le détail du fonctionnement des correcteurs de façon générale et les différents vérificateurs existants, ainsi les approche principale d'un vérificateur de grammaire.

1.2 Le traitement automatique de langage (TAL) :

Le traitement automatique de la langue arabe est une discipline en pleine expansion, dans laquelle on voit de plus en plus de recherches et de technologies se soucier des spécificités de cette langue [Motasem,Ghassan et Djioua, 2004] et proposer des outils nécessaires au développement de son traitement automatique [Boubekeur Y,2016].

Les principales applications du TAL :

•Traitement documentaire :

- La traduction automatique.
- La recherche de documents .
- La Reconnaissance de caractères.
- La Correction orthographique / grammaticale.

•La production de documents :

- les claviers « auto-correcteurs » .
- Les correcteurs d'orthographe ou de syntaxe .
- Les correcteurs « stylistiques».
- L'apprentissage assisté par ordinateur des langues naturelles.
- La génération automatique de documents à partir de spécifications formelles .

•Les interfaces naturelles :

- L'interrogation en langage naturel de bases de données.
- Les interfaces vocales[Gardent C, 2011].

1.3 Les niveaux de traitement en TAL

Nous introduisons dans cette section les différents niveaux de traitement nécessaires pour parvenir à une compréhension complète d'un énoncé en langage naturel. Chacun de ces niveaux suppose un traitement très spécifique permettant l'idée d'un système modulaire en traitement automatique [Chouchaoui et Brahimia, 2016].

Nous brossons ici les grands domaines du TAL, nous appuyant sur un découpage méthodologique classique dans le domaine et en linguistique :

- L'analyse morpho-lexicale** : qui se préoccupe de la structure des mots.
- L'analyse syntaxique** : consacrée à l'analyse de la structure des phrases.
- L'analyse sémantique** : qui s'intéresse au sens des phrases considérées individuellement.
- L'analyse pragmatique** : qui s'approche et prends en charge l'analyse de la phrase placée dans son contexte [Boubekeur Y, 2016].

1.4 La vérification grammaticale

1.4.1 Définition de grammaire :

La grammaire peut être présentée comme l'ensemble des règles de la langue, comme l'ouvrage qui les contient, ou de manière plus globale comme « l'étude systématique des éléments constitutifs et du fonctionnement de la langue » Grevisse 1993 [Souque A, 2014].

1.4.2 Définition d'erreur de grammaire :

Les erreurs grammaticales, telles que nous les définissons, sont des erreurs ayant trait aux règles de grammaire. Il s'agit essentiellement d'erreur d'accord en genre ou en nombre. Elle peuvent concerner tout mot du langage susceptible de s'accorder ou de se conjuguer [Augustin B, 2013].

1.4.3 Définition de la correction grammaticale :

La correction des erreurs grammaticales consiste à corriger les différents types d'erreurs dans le texte, telles que les erreurs d'orthographe, de syntaxe, de ponctuation, de grammaire et de choix de mots. La correction de la langue est également connue sous le nom de suppression des défauts et des erreurs de parole [Kali et Tchoketch, 2015].

1.4.4 Historique

Les correcteurs orthographiques existent depuis plusieurs décennies, tandis que les correcteurs dits "grammaticaux" n'arrivent véritablement sur le marché des Industries de la langue que vers les années 1990 [1].

En 1922, Odell et Russell avaient déjà proposé des méthodes pour la correction orthographique. Mais c'est à partir de 1957 qu'ont été développés des programmes destinés à corriger des erreurs orthographiques [Jacquet-Pfau C, 2001].

Ces premiers programmes ont permis la conception des correcteurs dits de première génération, dont le processus correspond à celui de « correcteurs lexicaux ».

Leur base de données est une liste de mots ordonnée alphabétiquement à laquelle est comparée la liste non ordonnée alphabétiquement des mots du texte. La comparaison se

fait entre des chaînes de caractères : si le mot du texte n'est pas dans la liste de référence, le correcteur signale une erreur. Ce type de correcteurs est aujourd'hui le fondement de la plupart des correcteurs intégrés aux logiciels de traitement de texte [Jacquet-Pfau C, 2001].

Les correcteurs dits de deuxième génération sont des « correcteurs grammaticaux heuristiques » . Ils représentent un progrès considérable sur les précédents dans la mesure où ils font intervenir des grammaires locales, qui analysent le contexte immédiat du mot et permettent de déterminer, par exemple, les règles d'accord propres à la langue pour laquelle ils ont été élaborés. Toutefois ces règles ne sont appliquées qu'à des contextes simples et localisés. C'est ce type de correcteur qui est encore utilisé dans les correcteurs grammaticaux intégrés dans les traitements de texte [2].

Enfin, les correcteurs dits de troisième génération fonctionnent sur une analyse grammaticale complète de phrases complexes. Autonomes, beaucoup plus élaborés, multifonctionnels, ces correcteurs sont maintenant intégrables dans plusieurs logiciels d'aide à la rédaction (traitement de texte, messageries, moteur de recherche, Publication Assistée par Ordinateur (PAO)...) parmi les plus courants prédéfinis par le concepteur du logiciel (Word, Wordperfect, ClarisWorks...). Ils emploient, entre autres, des technologies relevant de l'intelligence artificielle. Mais la complexité même qui en permet les performances est la cause de nombreuses fausses détections. Leur fonctionnement fait appel à des technologies innovatrices en matière d'intelligence artificielle [2].

1.4.5 Définition de correcteur grammaticale

Un correcteur est en informatique un outil logiciel permettant de vérifier et de corriger un texte en langue naturelle afin de détecter les fautes saisi. La «déttection » d'erreurs et l'intervention d'une « correction » s'avèrent cruciales et justifient la raison d'être de ce type d'application [Charnet C, 1998].

1.4.6 L'importance de la correction grammaticale

La correction grammaticale revêt une grande importance, de même que l'importance de la correction grammaticale dans un certain nombre de choses, la plus importante de

ces choses :

- Permettre d'améliorer la qualité du document réalisé.
- Le respect de la grammaire .
- Réduire le nombre de fautes présentes dans un travail écrit.
- Produire un texte sans fautes.
- Utiliser un correcteur comme une aide à la réflexion, comme un outil, un soutien dans le cadre de l'enseignement d'une langue [Jacquet-Pfau C, 2001].

1.4.7 Les fonctionnalités d'un correcteur grammatical

Le correcteur de grammaire a pour fonction de :

- signaler les phrases ou un mot incorrect dans un document.
- Rechercher les erreurs dans le texte lui-même.
- proposer des corrections, avec éventuellement une explication « linguistique » de l'erreur [Kali et Tchoketch, 2015].

1.4.8 Architecture des correcteurs grammaticaux

Les correcteurs grammaticaux ont vocation à détecter les erreurs pour lesquelles les correcteurs orthographiques ne sont pas compétents. Ils sont ainsi chargés d'une part des erreurs concernant l'organisation de la phrase (par ex. ordre des mots) et les relations entre les mots (par ex. accords), Leurs fonctionnements sont beaucoup plus complexes que les vérificateurs d'orthographe. Ils se décomposent classiquement en plusieurs étapes successives de traitement [Souque A, 2014] :

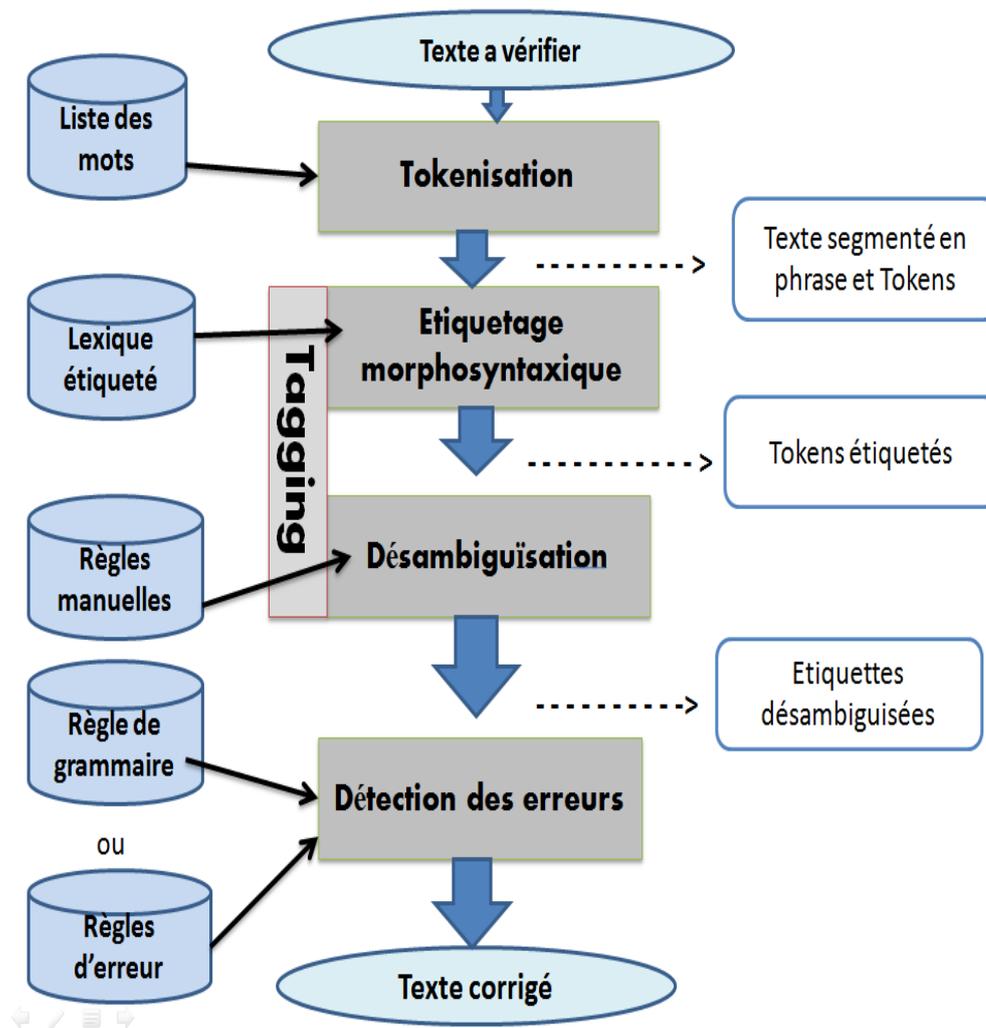


FIGURE 1.1 – Structure générale des correcteurs grammaticaux [Souque A, 2014].

a) Segmentation (Tokenisation)

La segmentation est une étape nécessaire et significative dans le traitement de la langue naturelle. La fonction d'un segmenteur est de couper un texte courant en segments qui sont traditionnellement, des mots simples.

La segmentation est étroitement liée à l'analyse morphologique. C'est beaucoup plus avec le cas des langues riches et complexes morphologiquement [Dilekh T, 2011].

–Les type de segmentation

Il existe plusieurs niveaux d'analyse signes d'intérêt permettent de repérer les différents éléments constituant le texte et en définir les frontières. On peut s'arrêter au niveau mot graphique, au niveau des unités lexicales ou aller au-delà de celles-ci pour arriver aux unités de base (les morphèmes). [Dilekh T, 2011]

Selon la visée de l'analyse à entreprendre : lexicale, morphologique ou syntaxique, on peut

généralement trouver trois types de segmentation :

1. **La segmentation lexicale (tokenization)** : produit des segments lexicaux (tokens).
2. **La segmentation morphologique** : produit des petites unités distinctes appelées morphèmes.
3. **La segmentation syntaxique (chunking)** : produit des unités indépendantes supérieures aux mots, comme les propositions, les syntagmes...etc [Dilekh T, 2011]

b) Etiquetage morphosyntaxique (Tagging) :

Au niveau de cette étape, l'étiqueteur attribue à chaque mot une ou plusieurs étiquettes (tag), contenant des informations sur sa catégorie grammaticale (verbe, nom, pronom . . .) , ainsi que sur ses traits de sous-catégorisation (genre, nombre, temps, personne . . .), appelés aussi traits morphosyntaxiques. Ces tags proviennent d'un lexique de formes fléchies étiquetées. Il s'agit de lexiques qui contiennent, comme pour les correcteurs orthographiques, toutes les formes fléchies de la langue, mais complétées par leurs caractéristiques morphosyntaxiques. Beaucoup de mots, dits « ambigus », reçoivent plusieurs tags. Ce sont les mots ayant des homographes avec des informations morphosyntaxiques différentes, on prend par exemple le segment (ذهب) qui on peut le considérer comme un verbe et aussi comme un nom désignant l'or [Kali et Tchoketch, 2015].

Cette multitude de tags pour un même mot peut facilement conduire à une mauvaise analyse morphosyntaxique du texte. Une désambiguïsation est donc une solution effectuée pour limiter le nombre d'étiquettes de ces mots et améliorer par la suite la détection des erreurs de grammaire. L'approche probabiliste est une approche pour désambiguïser les mots, elle nécessite un corpus d'apprentissage sans erreur, étiqueté avec les informations morphosyntaxique. Des calculs sont alors effectués. Il s'agit de la probabilité pour chaque

mot d’avoir tel ou tel tag. Lors de l’étiquetage, ces probabilités sont appliquées à chaque mot du texte analysé, et chacun reçoit alors l’étiquette qui correspond à la plus forte probabilité [Souque A, 2014].

Une autre approche consiste à utiliser des règles manuelles de désambiguïsation, sous forme d’expression régulières et fondées sur le contexte immédiat. Chaque règle consiste en un modèle d’un contexte en présence duquel tel mot prend tel tag [Souque A, 2014].

c) Détection d’erreurs :

La dernière étape concerne la vérification de la grammaire. Il existe des règles de deux types : les règles de grammaire, les règles d’erreurs.

- **Les règles de grammaire** : décrivent des modèles, des combinaisons de mots grammaticalement correctes. Chaque segment du texte à vérifier est comparé aux modèles contenus dans les règles de la base. S’il est agrammatical, aucune correspondance exacte ne sera trouvée entre lui et les règles de grammaire, puisqu’elles répertorient uniquement des constructions syntaxiques correctes. Une erreur sera alors signalée par l’outil [Souque A, 2014].
- **Les règles d’erreurs** : Le principe est le même que pour les règles de grammaire, à la différence près que ce sont cette fois les combinaisons de mots agrammaticales qui sont décrites dans les règles. Une erreur de grammaire est alors signalée par le correcteur lorsqu’un segment du texte et le modèle d’une règle coïncident [Souque A, 2014].

1.5 Les approches principales d’un vérificateur de grammaire

Un vérificateur de grammaire est un programme complexe qui nécessite beaucoup de recherche et de ressources linguistiques. Il existe trois approches principales pour mettre en œuvre un vérificateur de grammaire. nommément à base sur le Machine Learning, à base de statistiques et à base de règles.

1.5.1 Vérification basée sur le Machine Learning (ML) :

Les approches basées sur le Machine Learning (ML) reposent généralement sur des données de formation volumineuses et des textes parallèles. Lorsque le jeu d'entraînement et le jeu de tests sont similaires, l'approche ML donne de bons résultats mais pour les vérificateurs grammaticales le système basé sur ML ne fournit pas les commentaires nécessaires sur les erreurs, les utilisateurs sont généralement surpris lorsque le système prédit qu'une phrase correcte est fautive. Les résultats des systèmes basés sur le ML sont difficiles à interpréter [Madhvi S, 2018].

1.5.2 La vérification basée sur les statistiques :

Certains systèmes de vérification grammaticale utilisent des outils statistiques pour mettre en œuvre diverses tâches afin de détecter les erreurs grammaticales. Un corpus POS-annoté est utilisé pour créer une liste de séquences de balises POS. Certaines séquences (appelées N-Grams) seront très courantes (par exemple, déterminant, adjectif, nom comme dans le vieil homme), d'autres ne se produiront probablement pas du tout (par exemple, déterminant, déterminant, adjectif). Les séquences qui apparaissent souvent dans le corpus peuvent être considérées comme correctes dans d'autres textes ; des séquences inhabituelles peuvent être des erreurs. En réalité, même des permutations non grammaticales de mots sont encore probables [Naber D, 2003].

1.5.3 Vérification basée sur des règles

Cette approche est similaire à l'approche basée sur les statistiques, mais toutes les règles sont développées manuellement. Elle présente de nombreux avantages, une phrase n'a pas besoin d'être complète pour être vérifiée ; à la place, le logiciel peut vérifier le texte pendant la saisie et donner un retour immédiat. Il est facile à configurer car chaque règle a une description expressive et peut être activée et désactivée individuellement. Il peut offrir des messages d'erreur détaillés avec des commentaires utiles, expliquant même les règles de grammaire. Il est facilement extensible par ses utilisateurs, car le système de règles est facile à comprendre, du moins pour de nombreux cas d'erreur simples mais courants. Il peut être construit progressivement, en commençant par une seule règle, puis en l'étendant règle par règle [Shaan K, 2005].

1.6 Les vérificateurs grammaticaux existants pour la langue arabe :

La plupart des recherches sur le traitement de la langue naturelle en arabe sont consacrées à la phase d'analyse morphosyntaxique sans accorder aucune attention au problème de la vérification grammaticale [Shaalán K, 2005] .

1.6.1 Microsoft Word :

Word est un traitement de texte édité par la société Microsoft. les versions de Microsoft office(2003) comprend un vérificateur de grammaire pour le groupe arabe .C'est le seul vérificateur de grammaire arabe sur le marché. Ce vérificateur de grammaire prend en charge la vérification et la correction de phrases simples en arabe, il est intégré à un correcteur orthographique et présente une fonctionnalité unique qui permet de corriger les erreurs de manière itérative. Cette fonctionnalité permet de corriger des erreurs multiples dans une même phrase. La correction de ponctuation dans le vérificateur de grammaire de l'Office 2003 est une fonctionnalité totalement nouvelle pour la langue arabe. Cette fonctionnalité vérifie les espaces, les virgules et les points d'interrogation [Shaalán K, 2005].

رسالة مايكروسوفت وورد المقترحة	الجملة الصحيحة المفترض كتابتها	الجملة الخاطئة
يجب نصب هذا الاسم لانه سبق بناسخ يتطلب نصبه وابداله بالكلمه المسلمين	إن المسلمين صائمون	إن المسلمون صائمين
المسند اليه مرفوع تستبدل كلمة المسلمين بالكلمه المسلمون لتصحيح الخطأ	المسلمون صائمون	المسلمين صائمين
المفاعيل منصوبه تستبدل كلمة المدرسون بالكلمة المدرسين لتصحيح الخطأ	احترم حسين المدرسين	احترم حسين المدرسون

TABLE 1.1 – Exemples testé sous Microsoft Word [Shaalan K, 2005]

la figure ?? exprime un exemples testés sous le Microsoft Word.

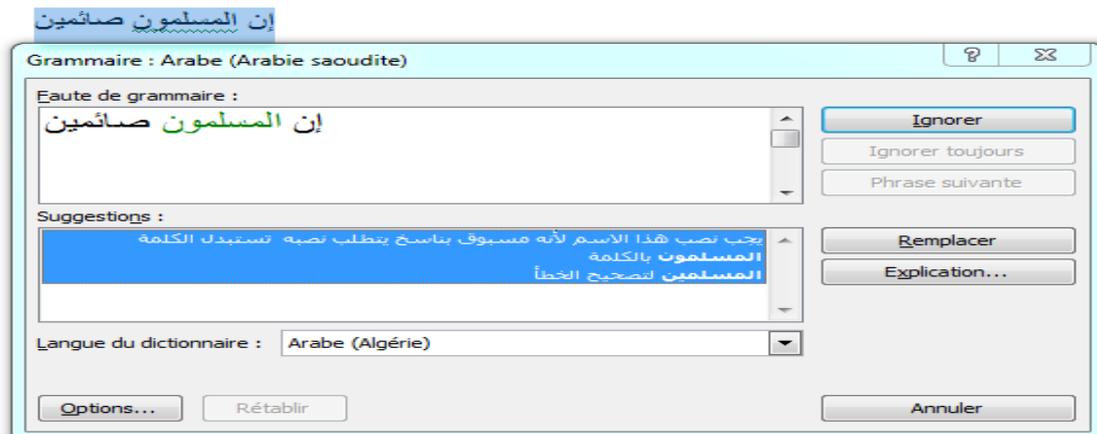


FIGURE 1.2 – La correction grammaticale avec Microsoft Word .

1.6.2 Arabic GramCheck :

Arabic GramCheck (2005) est un vérificateur de grammaire basé sur la syntaxe pour l'arabe standard moderne. Ce système est basé sur une analyse syntaxique approfondie et sur une approche de relaxation des caractéristiques pour la détection de phrases arabes mal formées.

Arabic GramCheck aide l'utilisateur à écrire une phrase en analysant chaque mot et en n'acceptant que si la phrase est grammaticalement correcte.

GramCheck se caractérise principalement par le fait qu'il effectue une analyse grammaticale complète des phrases et vérifie que la phrase ne contient pas d'erreurs grammaticales communes, décrit le problème et propose des suggestions d'amélioration.

Le vérificateur de grammaire est essentiellement composé de deux parties : un analyseur morphologique arabe et un analyseur syntaxique étendu à un gestionnaire de contrôle grammatical. Le système est implémenté dans SICStus Prolog 3.9 qui s'exécute sous Microsoft Windows [Shaalán K, 2005].

1.6.3 Lightproof :

Light proof (2011) est un générateur de vérificateur grammatical en Python. Il est fondé sur des règles de système pris en charge par eux ou analyseur morphologique Hunspell avec licence open source [3].

Ce correcteur est caractérisé par :

- Extension pour OpenOffice.org 3.0.1 ou version ultérieure.
- Conditions Python facultatifs dans les règles.
- Intégration avec l'analyseur morphologique Hunspell.
- Ecrite en Python en utilisant le module regex (expression régulière).
- Développement rapide de nouveaux modules de langues [3].

ZERROUKI Taha à développé le vérificateur de grammaire arabe pour Open Office / Libre Office, basée sur Light Proof. Ce programme propose de corriger la ponctuation, et quelques erreurs courantes. Les erreurs communes telles que :

Explication	Phrase corrigé	Phrase erroné
لأن ثبت فعل لازم يتعدى بالهمزة والتضعيف	ثبت أنه	ثبت بأنه
(Money laundering) لأنها ترجمة مبتذلة ل	تبييض الأموال	غسيل الأموال

TABLE 1.2 – Exemple des erreurs testées en Light Proof [4].

1.6.4 LanguageTool

Language tool ont été écrites au début des années deux mille par Naber (2003) en Python, mais il est réécrit peu après en java pour être intégré à Open Office [Souque A, 2007].

LT est un correcteur à base de règles qui se base sur une analyse syntaxique de surface. Chaque token rencontré se voit assigner toutes les étiquettes morphologiques qu'il est susceptible d'avoir.

LT utilise ensuite des règles désambiguïsation pour tenter de vérifier l'exactitude de la phrase et proposer une correction en cas d'échec. en plus des erreurs grammaticales et lexicales de mot réel, LT peut également corriger des erreurs de mise en forme avec les règles appropriées. il ne traite par contre pas les erreurs de non mot, mais est généralement utilisé en conjonction avec Hunspell [Augustin B, 2013].

le support de la langue arabe a cet outil de correction grammaticale LT a été ajouté par [Kali et Tchoketch, 2015].

Rôle de LanguageTool :

1. Un bon outil, simple et gratuit, à utiliser soi-même ou à conseiller pour vérifier grammaire, orthographe, et règles typographiques d'un texte avant de le publier ou de le rendre. Les logiciels professionnels sont souvent très onéreux, mais heureusement il existe des solutions en ligne.
2. LanguageTool trouve des erreurs en recherchant des motifs définis par des règles XML. Il est aussi possible de définir des règles en Java. On peut aussi l'instal-

ler sous forme d'applications pour OpenOffice et LibreOffice, directement sur PC (LanguageTool nécessite Java 7) ou via une extension Firefox [5].

LanguageTool peut être utilisé :

- En tant que LibreOffice / OpenOffice.org Extension
- Comme une application autonome
- Comme une application autonome sur la ligne de commande
- En tant que Firefox Add-On [6].

1.7 Discussions et comparaison entres les vérificateurs grammaticaux :

LanguageTool	LightProof	Microsoft Word	Arabic Gram-Check
Open Source.	Open Source.	Closed source	Closed source
La langue arabe en cours d'intégration	La langue arabe est intégrée en 2011.	La langue arabe intégré en 2003	La langue arabe intégré en 2005
Disponible.	Disponible.	Disponible	Indisponible.

TABLE 1.3 – La comparaison entres les vérificateurs grammaticaux

Le tableau 1.3 montre le manque existant dans la vérification grammaticale en langue arabe.

La plupart des correcteurs grammaticaux sont payant et ils sont propriétaires, fermés, et on ne peut pas voir ses codes sources et connaitre leurs méthodes de travail. Y a un seul correcteur grammaticale (libre) « LightProof » qui est intégré dans OpenOffice/LibreOffice. Ce programme propose de corriger la ponctuation et quelques erreurs connues. Ceci est une version de démonstration et dans la phase de développement.

Au long de ce travail, on va essayer de modéliser les erreurs grammaticales communes de la langue arabe dans le système de correction des fautes grammaticales et linguistiques

LanguageTool, dont les ressources linguistiques seront accessibles et modifiables, et dont le formalisme générique le rendra adaptable à d'autres recherches et améliorations.

1.8 Conclusion

A travers ce premier chapitre, nous avons mis le point sur les concepts de base de TAL, nous avons ainsi retenu dans le cadre de ce travail le détail du fonctionnement des correcteurs de façon générale avec différents vérificateurs existants, ainsi les approche principale d'un vérificateur de grammaire .

Dans ce qui suit, nous allons mettre le point sur les aspects des erreurs en langues arabes.

Les aspects des erreurs en langue arabe

2.1 Introduction

La langue arabe est l'une des langues les plus anciennes qui est considérée comme une langue dans le domaine du traitement automatique de la langue à cause de sa richesse morphologique.

Et vu que l'arabe est une des langues les plus parlées dans le monde, et avec l'évolution de l'informatique, la nécessité a poussé les chercheurs à aborder des problématiques variées comme la morphologie, la traduction automatique, l'indexation des documents arabes, etc...

Au cours de ce chapitre nous présenterons les caractéristiques de la langue arabe ainsi que ses propriétés morphologiques et syntaxiques.

2.2 Langue arabe

L'arabe (العربية) est une langue sémitique aujourd'hui parlée en première ou seconde langue par plus de 440 millions de personnes au sein du monde arabe et de la diaspora arabe.

La langue arabe est marquée par une importante diglossie entre l'arabe littéral, langue véhiculaire surtout écrite, et l'arabe dialectal, langue vernaculaire surtout orale. L'arabe littéral comprend l'arabe classique (pré-coranique, coranique, et post-coranique) et l'arabe standard moderne. L'arabe dialectal comprend de nombreuses variétés régionales, pas toutes intelligibles entre elles. L'arabe est parlé à des degrés divers dans les pays arabes du Moyen-Orient, en Iran (province du Khouzistan), dans les pays d'Afrique du Nord, du Sahara, du Sahelet sur les côtes de la Corne de l'Afrique. Il est également pratiqué dans la diaspora arabe [Boubekeur Y, 2016].

2.3 les caractéristiques de la langue

L'arabe est caractérisée par

2.3.1 Structure d'un mot :

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 schématise une structure possible d'un mot. Notons que la lecture et l'écriture d'un mot se fait de droite vers la gauche [Araïbi et Abdelmoumen, 2016].

Enclitique	Suffixe	Cors schématique	Préfixe	Proclitique
-------------------	----------------	-------------------------	----------------	--------------------

- **Proclitiques** : sont des prépositions ou des conjonctions.
- **Cors schématique** : représente la forme de base pour chaque mot.
- **Préfixes** et **suffixes** : expriment les traits grammaticaux et indiquent les fonctions : cas du nom, mode du verbe et les modalités (nombre, genre, personne,...)
- **Enclitiques** : sont des pronoms personnels [Araïbi et Abdelmoumen, 2016].

Exemple : أتساعدوننا؟

Enclitique	Suffixe	Cors schématique	Préfixe	Proclitique
نا	ون	ساعد	ت	أ
pronom suffixe complément du nom.	suffixe verbal exprimant le pluriel.	La racine	préfixe verbal du temps de l'inaccompli	conjonction d'interrogation

TABLE 2.1 – Structure d'un mot [Araïbi et Abdelmoumen, 2016]

2.3.2 Catégories des mots

L'arabe considère trois (03) catégories de mots :

- **Le verbe** : entité exprimant un sens dépendant du temps, c'est un élément fondamental auquel se rattachent directement ou indirectement les divers mots qui constituent l'ensemble.
- **Le nom** : l'élément désignant un être ou un objet qui exprime un sens indépendant du temps.
- **Les particules** : entités qui servent à situer les événements et les objets par rapport au temps et l'espace, et permettent un enchaînement cohérent du texte [7].

1) verbe :

Le verbe en arabe, la majorité des mots dérivent d'un verbe de trois (03) consonnes qui représente une racine d'un groupe de mots. Le mot en arabe se détermine à partir d'un radical en rajoutant des préfixes, des suffixes ou les deux en même temps. Comme les autres langues, la conjugaison de verbe en arabe dépend de facteurs suivants [7] :

- **L'aspect** : accompli (passé) ou inaccompli (présent).
- **Le nombre du sujet** : singulier, pluriel ou duel.
- **Le genre du sujet** : masculin ou féminin.
- **La personne** : première, deuxième ou troisième.

— **La voix** : active ou passive.

La langue arabe possède trois temps :

1. **L’accompli** : indique le passé et les verbes conjugués se distinguent par des suffixes. Pour notre exemple, avec le féminin pluriel, on obtient **نَجَحْنَ** «elles ont réussi» pour le masculin pluriel, on obtient **نَجَحُوا** «ils ont réussi».
2. **L’inaccompli présent** : les verbes conjugués à ce temps se distinguent par les préfixes. Pour notre exemple, au masculin singulier on obtient **يَنْجِحُ** «il réussit» et pour le féminin singulier on obtient **تَنْجِحُ** «elle réussit» .
3. **L’inaccompli futur** : la conjugaison d’un verbe au futur nécessite d’ajouter l’antéposition au début du verbe conjugué à l’inaccompli. En ajoutant l’antéposition à notre exemple **س** on obtient **سَيَنْجِحُ** « il réussira » , qui désigne le futur ; on peut également ajouter l’antéposition **سوف** on obtient **سَوْفَ يَنْجِحُ** « il va réussir» [7].

2) Les noms :

En arabe les noms sont divisés en deux familles, ceux qui sont dérivés à partir d’une racine (verbale) et les autres comme les noms étrangers et certains noms fixes. La première famille est composée des tous les noms qui sont dérivés à partir d’une racine verbale. La variabilité des noms obéit à plusieurs règles, en ajoutant des morphèmes spécifiques [Araibi et Abdelmoumene, 2016] :

- **Le féminin singulier** : pour obtenir le nom féminin singulier, dans la majorité des cas on ajoute la lettre ” ة ” (exemple : **طفل** «enfant» devient **طفلة** «fille»).
- **féminin pluriel externe** : pour obtenir le nom féminin pluriel on ajoute les deux lettres ” ات ” (exemple : **فتاة** devient **فتيات**)
- **Le masculin pluriel externe** : on ajoute les deux lettres ” ين ” ou ” ون ” qui dépendent de la position du nom dans la phrase (avant ou après le verbe).

Exemple : **معلم** «Enseignant» se transforme en **معلمون** ou **معلمين** «enseignants»,

Le pluriel masculin

- **féminin et interne** : c'est le cas le plus complexe en arabe.

la construction de ces types des noms s'obtient en insérant des lettres au début, au milieu ou à la fin (exemple : **طفل** «enfant » se transforme en **اطفال** «enfants» et **فصل** «saison» se transforme en **فصول** «saisons».) Comme en français, les noms en arabe assument des fonctions diverses [Araibi et Abdelmoumene, 2016] :

- **Agent** : celui qui fait l'action.
- **Objet** : celui qui subit l'action.
- **Instrument** : signifiant l'instrument de l'action.
- **Lieu** : qui désigne en général un endroit (exemple : maison " **منزل** ")
- **Nom d'action** : désigne l'action etc... Notons que, la morphologie des noms arabes dépend de ces fonctions.

3) Les particules :

En général, les particules sont les mots outils pour une langue donnée. Ils représentent en particulier les mots qui expriment des faits ou des choses par rapport au temps ou au lieu. Par exemple :

- **Particules temporelles** : **منذ** (pendant), **قبل** (avant), **بعد** (après),...
- **Particules spatiales** : **حيث** (où) .

Les particules peuvent aussi exprimer des pronoms relatifs (la détermination, avec une valeur référentielle) : **الذي** (ce), **الذين**(ceux), **التي**(cette),... [7].

2.4 Les branches de grammaire pour la langue arabe

La grammaire arabe se divise en deux branches :

2.4.1 La morphologie

La morphologie est un domaine de la langue qui permet la description des règles régissant la structure interne des mots (unités lexicales), chez les grammairiens la morphologie est l'étude des formes des mots (flexion et dérivation), en d'autres termes, la morphologie est l'étude des mots considéré isolément (hors contexte) sous le double aspect de la nature et des variations qu'ils peuvent subir [Cheragui M, 2012].

Analyse et ambiguïté

En langue arabe, l'analyse morphologique est d'autant plus importante que les mots sont fortement agglutinés, c'est-à dire qu'ils sont formés dans leur majorité par assemblage d'unités lexicales et grammaticales élémentaires [Cheragui M, 2012].

Ainsi le principale objectif d'une analyse morphologique automatique est de reconnaître ces unités et d'attribuer a chacune divers types d'informations telles que la catégorie grammaticale et les traits morphologique (genre, nombre, la voix, le mode,... etc). C'est a ce stade, que l'ambiguïté morphologique se manifeste le plus souvent ; lorsque l'analyse assigne a une unité lexicale plusieurs ce qui génère la notion de combinatoire [Cheragui M, 2012].

Prenant comme exemple, le mot (احمد), ce dernier peut prendre plusieurs interprétations.

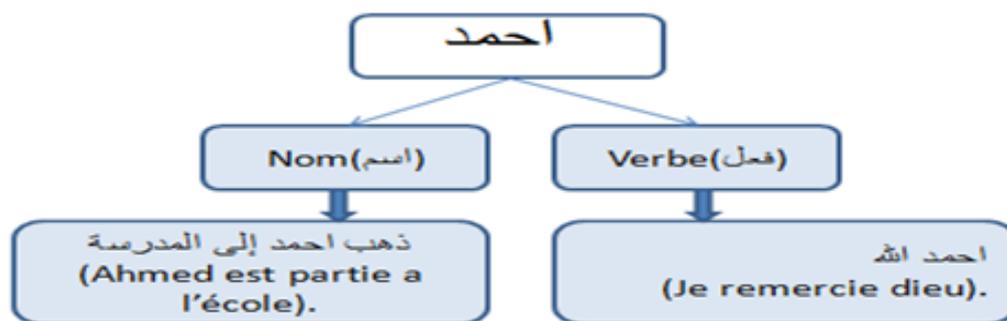


FIGURE 2.1 – Exemple d'ambiguïté morphologique [Cheragui M, 2012].

2.4.2 La syntaxe

La structure d'une phrase arabe :

En langue arabe, il y a deux types de phrases : les phrases verbales et les phrases nominales.

A) Les phrases verbales

Servent à indiquer un évènement ou une action. Elles commencent par un verbe suivi d'un sujet et d'un complément ; ce dernier est optionnel. La phrase verbale est celle que l'on rencontre le plus souvent dans l'expression courante [8].

Elle est composée de trois (03) parties :

— **Le Verbe** : Appelé en arabe **الفعل**

C'est la partie qui décrit une action effectuée dans le passé, le présent, ou le futur.

— **Le Sujet** : Appelé en arabe **الفاعل**

C'est la personne ou la chose effectuant l'action décrite par le verbe.

— **Le Complément** : Appelé en arabe **المفعول**.

Et il y a deux genres :

1. **Direct** : Il correspond à la personne ou la chose sur laquelle s'applique le Verbe.
2. **Indirect** : C'est le complément qui indique le temps, le lieu, les circonstances. . . etc.



FIGURE 2.2 – Exemple de phrase verbale.

B) Les phrases nominales

En langue arabe, une phrase nominale suit les règles suivantes :

— Elle commence soit par un nom ou soit par un pronom.

— Elle est composée de deux parties : la première partie appelée mubtada (الابتداء)

est le sujet de la phrase. La deuxième partie, appelée khabar (الخَبْر), apporte une information sur le sujet [9].



FIGURE 2.3 – Exemple de phrase nominale.

Les règles d'accord

A) l'accord du verbe / sujet

L'accord du verbe avec son sujet obéit à un certain nombre de règles :

Si le sujet est un nom désignant des êtres humains, deux cas :

- Le verbe précède son sujet (cas général) ; l'accord se fait seulement en genre, jamais en nombre, le verbe restant au singulier.

Exemple : تخرج صديقاتي معا مرة كل اسبوع

- Le verbe suit son sujet il s'accorde en genre et en nombre.

Exemple : صديقاتي يخرجن معا مرة كل اسبوع

Si le sujet est un nom désignant des animés non humains ou des inanimés, deux cas :

- Le sujet est un singulier : le verbe s'accorde en genre.
- Le sujet est pluriel, le verbe s'accorde au féminin singulier, qu'il précède ou non son sujet. [Djoudi M, 1991]

B) l'accord d'adjectif (العت)

Les règles d'accord de l'adjectif suit le nom qu'il qualifie en détermination en genre et en nombre [Echeteia A, 2009] .

- Si le nom qu'il qualifie est masculin, alors l'adjectif sera masculin.

Exemple : هذا رجلٌ كريمٌ

- Si le nom qu'il qualifie est féminin, alors l'adjectif sera féminin.

Exemple : هذه امرأة كريمة

- Si le nom qu'il qualifie est au pluriel, alors l'adjectif sera au pluriel.

Exemple : مررت برجال كرماء

- Si le nom qu'il qualifie est مَرْفُوع , alors l'adjectif sera également مَرْفُوع .

Exemple : زارنا رجلٌ كريمٌ

C) L'accord de nombre et l'énuméré (العدد و المعدود)

En Arabe, un nombre se compose de deux parties ; le nombre (a'dad) et l'énuméré (ma'hdoud). Le nom, c'est à dire ce qui "est compté", est le (ma'hdoud) et le chiffre est le (a'dad) [Abou obeida, 2007].

Les règles de grammaire pour le nombre et l'énuméré :

- le nombre 1 et 2 : Si l'énuméré المعدود est féminin, alors le nombre العدد est féminin si l'énuméré المعدود est masculin, alors le nombre العدد est masculin.
- Les nombres de 3 à 10 : si l'énuméré (ma'hdoud) est féminin, alors le nombre (a'dad) est masculin et vice versa si l'énuméré (ma'hdoud) est masculin, alors le nombre (a'dad) est féminin.
- Les nombres de 11 à 19 : sont des nombres composées de deux parties :
 1. 11 et 12 = Si l'énuméré المتعدود est masculin alors la première et la seconde partie sont masculins.
Si l'énuméré المتعدود est féminin alors la première et la seconde partie sont féminins.
 2. De 13 à 19 = Si l'énuméré المتعدود est masculin, alors la première partie est féminin ; et la seconde partie s'accorde avec المتعدود .
L'inverse est valable si l'énuméré est féminin [Abou obeida, 2007].

2.5 Analyse des erreurs en langue arabe

Dans la littérature, l'analyse des erreurs ne concerne que les erreurs grammaticales arabes les plus courantes, sans indication de la fréquence d'apparition de ces erreurs ; Cela a pour but d'aider les écrivains arabes à résoudre la plupart des problèmes grammaticaux qui minent leur écriture [Shaalán K, 2005].

Afin d'étudier la possibilité de développer un vérificateur de grammaire arabe informatisé, sevré et classifié les erreurs grammaticales communes qui se produisent lors de la formulation d'une phrase MSA dans un style formel. Des spécialistes arabes ont vérifié que ces erreurs étaient les plus courantes.

Voilà quelque exemples des erreurs grammaticales que nous avons trouvé à l'aide des livres spécialise dans le domaine des erreurs grammaticales communes de l'arabe.

Type d'erreur	Exemple	Version corrigé
Masculin au lieu de féminin	هذا بئر عميق ضبع مفترس	هذه بئر عميقة ضبع مفترسة (ضبع اسم مؤنث)
Féminin au lieu de masculin	هذه نجمة (كوكب) انتفخت بطني	هذا نجم انتفخ بطني
Prépositions	برهنت عن يدل إلى	برهنت على يدل على
Accord de genre et nombre entre l'accusatif circonstanciel et le sujet	جاءت بعض السيدات تحملن أطفالهن	جاءت بعض السيدات يحملن أطفالهن

Accord de genre et nombre entre l'inchoactif et l'énonciatif	الجنود يدافعان عن الوطن	الجنود يدافعون عن الوطن
Accord de genre, nombre entre l'adjectif et le nom	الرجال الكريم يساعدون الناس	الرجال الكرماء يساعدون الناس
Accord de genre entre le verbe et le sujet	شرب البنت عصير البرتقال	شربت البنت عصير البرتقال
Pluriel	المحلات إطارات	المحال أطر

TABLE 2.2 – Les erreurs grammaticales communes de l'Arabe [Shaalan K, 2005].

Le tableau 2.2 détaillent les erreurs grammaticales possibles de la langue maternelle arabe. Ces erreurs sont représentatives de celles rencontrées par l'utilisateur de traitement de texte moyen lors de la saisie en arabe qui sont basées sur une étude récente. Par souci de clarté, des exemples d'erreur en arabe pertinents suivis de la syntaxe correcte sont donnés avec leur classification.

2.6 Conclusion

Nous avons cherché dans ce chapitre à présenter de façon globale les aspects de la langue arabe , et précisément la morphologie de l'arabe et sa grammaire, en suite les erreurs grammaticales communes en langue arabe.

Dans le prochain chapitre, nous aurons proposé une méthode valide et performante visant la détection puis la correction des erreurs en langue arabe.

La conception

3.1 Introduction

Un vérificateur grammatical "Language Tool" doit être capable de détecter un maximum des erreurs de grammaire que commettent les scripteurs. Il est ainsi nécessaire, pour sa conception, de disposer d'une typologie et d'un modèle des erreurs possibles.

Ce chapitre décrit comment nous avons doté le correcteur des règles nécessaires à la détection des fautes de grammaire dans un texte écrit en arabe.

Tout d'abord, nous décrivons la conception de notre approche avec certaines catégories d'erreurs dont l'objectif est d'enrichir la base des règles d'erreur, pour ce faire nous expliquons et détaillons les différentes étapes de notre prototype.

3.2 Choix de technique

Une approche de vérificateur de grammaire qui nous aurons utilisées dans cette recherche est l'approche basé sur des règles créées à la main par des experts en langues est basé sur les connaissances linguistiques essentielles.

Dans cette approche basé sur des règles, fournit une analyse détaillée de l'écriture de l'apprenant à l'aide des connaissances linguistiques et fournit un retour d'information raisonnable. Ces commentaires aident les apprenants à améliorer leurs compétences en écriture.

Le principal avantage de l'utilisation des règles est la simplicité avec laquelle elles peuvent générer un retour d'information. Une haute précision peut être obtenue en appliquant des contraintes et des règles définies.

3.3 Description générale

Le système que nous avons choisis pour réaliser ce travail est le logiciel "Language Tool" car c'est un bon outil, simple et gratuit, à utiliser soi-même ou à conseiller pour vérifier la grammaire, l'orthographe d'un texte. LT trouve des erreurs en recherchant des motifs définis par des règles XML. Il est aussi possible de définir des règles en Java.

Le système "language tool" est conçu pour détecter les erreurs grammaticales dans un texte arabe. Il se compose de plusieurs phases, chacune d'elles utilisant les informations reçues de la phase précédente pour finalement obtenir un texte contenant le plus petit nombre d'erreurs.

l'outil aura la structure suivante, schématisée comme suite :

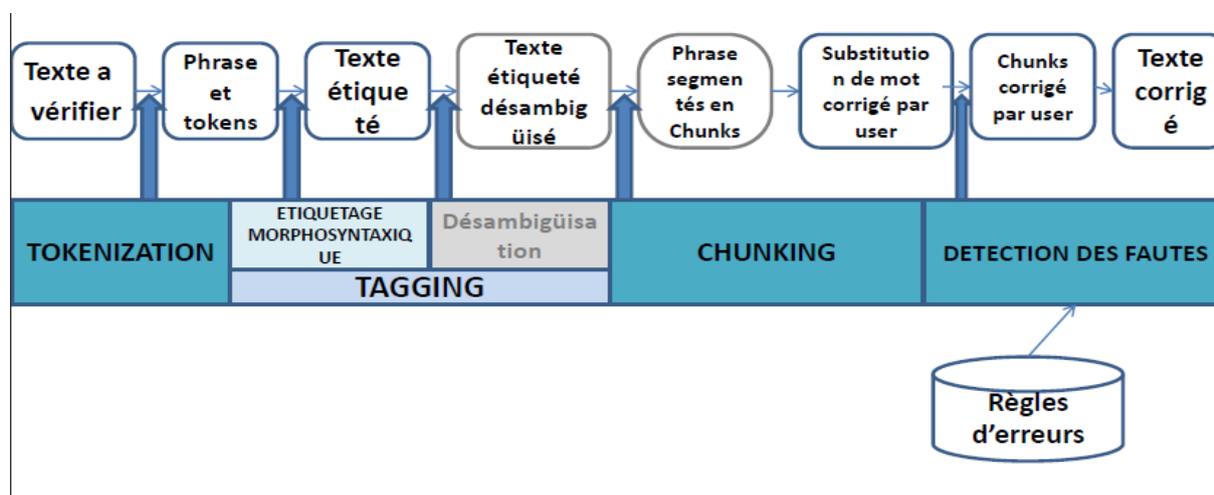


FIGURE 3.1 – La structure de LanguageTool.

1. **Tokenization** : segmentation du texte en phrases, puis en tokens.
2. **Tagging** : étiquetage puis désambiguïsation
 - Les tokens seront étiquetés en XML.
 - Les tokens seront ensuite désambiguïsés, avec une méthode à base de règles.
3. **Chunking** : délimitation des chunks au sein des phrases.

4. **Détection des fautes** : succession de plusieurs types de détection.
- La substitution d'un mot par un autre, comme une confusion d'homophones ou une faute de frappe, aboutit souvent à la présence d'un mot d'une catégorie grammaticale inattendue. Certaines de ces erreurs pourront être détectées, puis corrigées par l'utilisateur. Un retour à l'étape du tagging permettra ensuite de rectifier l'étiquetage, ainsi que la segmentation en chunk.
 - Les erreurs d'accords au sein des chunks seront signalées à l'utilisateur qui pourra les corriger au fur et à mesure. Les traits de ces chunks seront alors rectifiés.
 - Les accords erronés entre chunks pourront enfin être détectés, signalés, et corrigés.

3.4 Classification des erreurs de grammaire

D'après les recherches et les études sur les erreurs communes de la langue arabe nous avons arrivé a classé ces erreurs selon les catégories suivant : Les erreurs d'accord, les erreurs sémantiques (الاطاء الدلالية), de verbe transitifs الفعل متعدي , de liaison, l'erreur de النواسخ et de Pluriel.

Voilà notre prototype des erreurs :

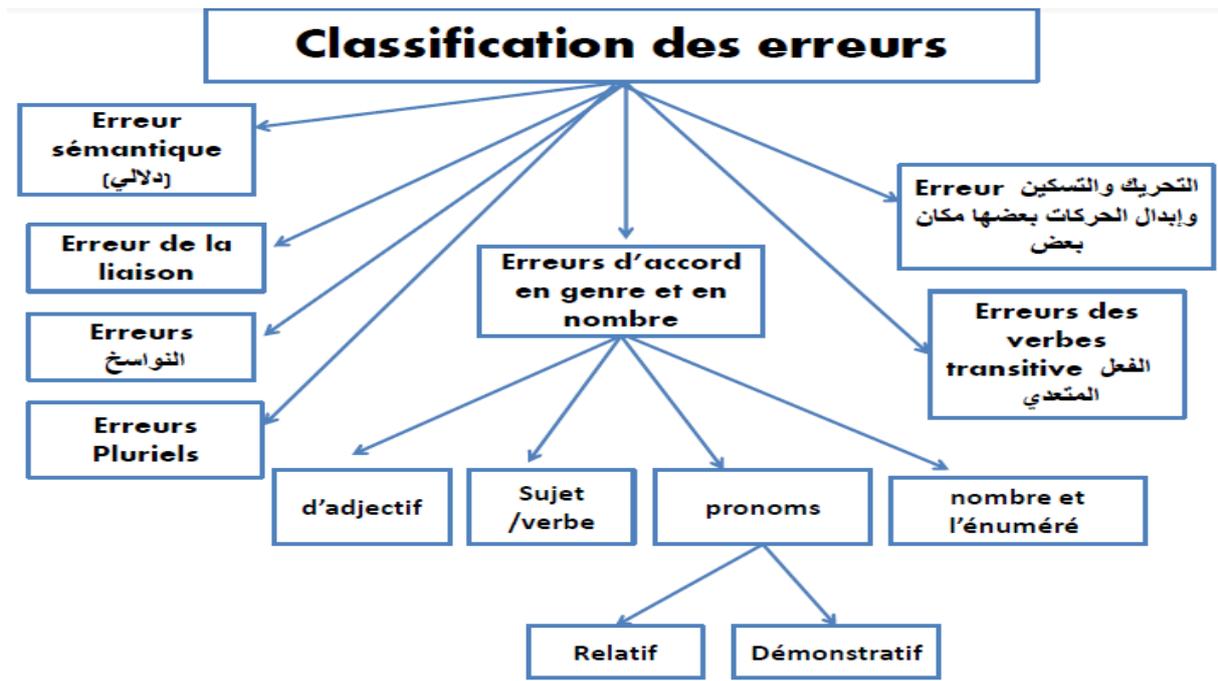


FIGURE 3.2 – Prototype des erreurs.

3.4.1 Erreurs d'accord de genre et de nombre :

Nous avons trois sous catégories qui s'accord en genre et en nombre sont :

l'accord entre le sujet/verbe, l'accord de l'adjectif (النعته) , l'accord de pronom relatif , l'accord de pronom démonstratif et l'accord de nombre et l'énuméré.

1) L'accord entre le sujet et le verbe :

–la règle de l'accord entre le sujet/verbe :

Si le verbe vient après le sujet **alors** l'accord —>en genre et en nombre

Si le verbe vient avant le sujet **alors** l'accord —>en genre

Figure ?? montre un exemple de l'accord entre le verbe et le sujet selon la règle .

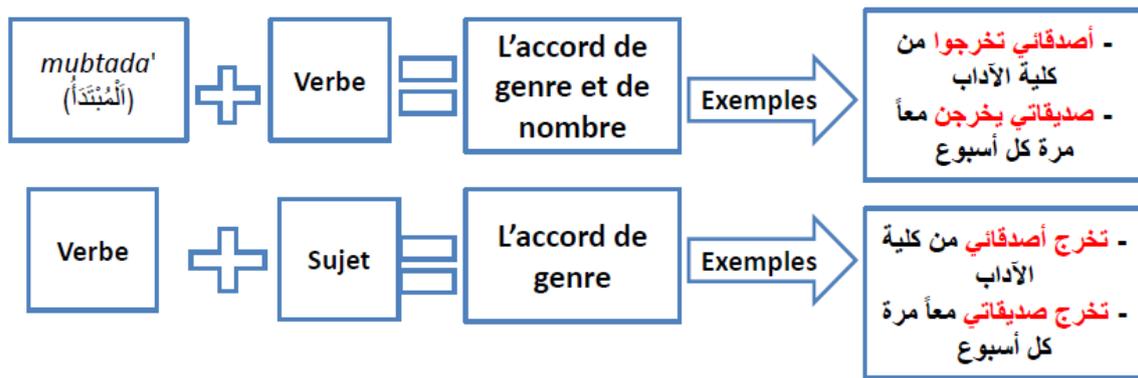


FIGURE 3.3 – L'accord entre le sujet et le verbe.

Exemple selon l'architecture

-L'erreur : شرب البنت عصير البرتقال

L'architecture ci-dessous concerne le cas d'un verbe au singulier masculin (شرب) avec un sujet singulier féminin (البنت). Donc après les deux premiers étapes, la dernière comme les autres cas, consiste à comparer la syntaxe entrée par celle du corpus, alors le token (شربت) est proposé comme une solution pour que la phrase (le sens) soit correcte.

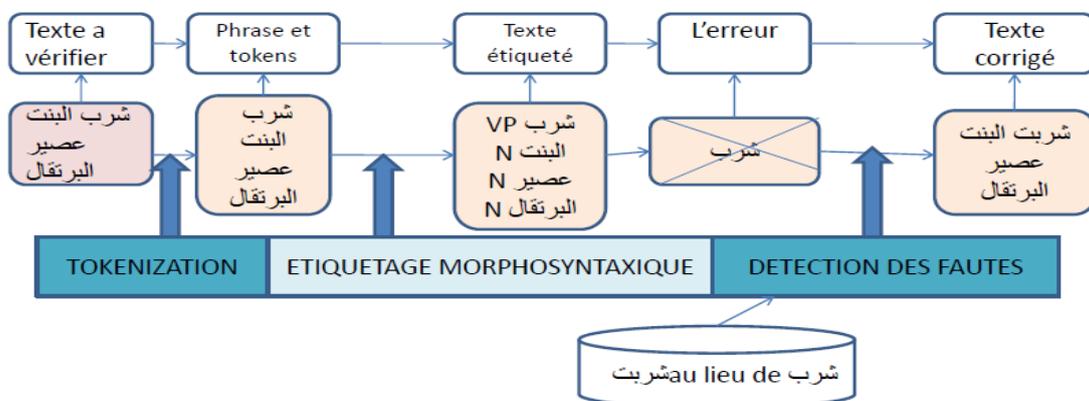


FIGURE 3.4 – Exemple d'erreur entre le sujet et le verbe

2) L'accord d'adjectif (النعته) :

Adjectif qualificatif et nom qualifié en Arabe suit le nom qu'il qualifie.

• Règle d' accord d'adjectif (النعته) :

Si NomQualifie —>masculin **alors** Adjectif —>masculin

Si NomQualifie —>féminin **alors** Adjectif —>féminin

Si NomQualifie —>pluriel **alors** Adjectif —>pluriel

Si NomQualifie —>singulier **alors** Adjectif —>singulier

Exemple :

Explication	La correction	L'erreur
L'adjectif s'accord en genre et en nombre	الرجل الكريم	الرجل الكرماء

TABLE 3.1 – Exemple d'accord d'adjectif

-3) L'accord de pronom relatif (اسم الموصول) :

إذا كانت المفردة مذكرة لا بد من الايتيان باسم الموصول مذكرة والعكس صحيح.

Si Nom —>NFS **alors** Det —>DFS.

Si Nom —>NMS **alors** Det —>DMS.

Si Nom —>NFP **alors** Det —>DFP.

Si Nom —>NMP **alors** Det —>DMP.

le tableau ?? exprime des exemples selon la règle de pronom relatif.

La corrections	Les erreurs
تطور العلاقات إلى المستوى الذي هي عليه اليوم	تطور العلاقات إلى المستوى التي هي عليه اليوم
النشاط الذي بدأت به المرأة	النشاط التي بدأت به المرأة

TABLE 3.2 – Exemple d'un pronom relatif .

-4) L'accord de Pronom Démonstratif (الاسم الإشارة)

Les pronoms démonstratifs sont l'équivalent en français de 'ce', 'cette'

les règles d'un pronom démonstratif :

Si Nom —>NFS alors Det —>DFS.

Si Nom —>NMS alors Det —>DMS.

Si Nom —>NFP alors Det —>DFP.

Si Nom —>NMP alors Det —>DMP.

Le tableau ?? exprime quelques exemples selon les règles de pronom démonstratif :

L'explication	La corrections	Les erreurs
الكأس اسم مؤنث لقوله تعالى وَيُسْقَوْنَ فِيهَا كَأْسًا كَانَ مِزَاجُهَا زَنْجَبِيلًا	هذه الكأس	هذا الكأس
جهنم اسم مؤنث لقوله تعالى آ هَذِهِ جَهَنَّمُ الَّتِي كُنْتُمْ تُوعَدُونَ	هذه جهنم	هذا جهنم
لأنَّ كلمة بئر مؤنثة قال الله تعالى وَبِئْرٍ مُّعَطَّلَةٍ وَقَصْرٍ مَشِيدٍ.	هذه البئر	هذا البئر

TABLE 3.3 – Exemple d'erreur d'un pronom démonstratif [Adnani M, 1983] .

Exemple selon l'architecture

-L'erreur: هذا بئر عميق

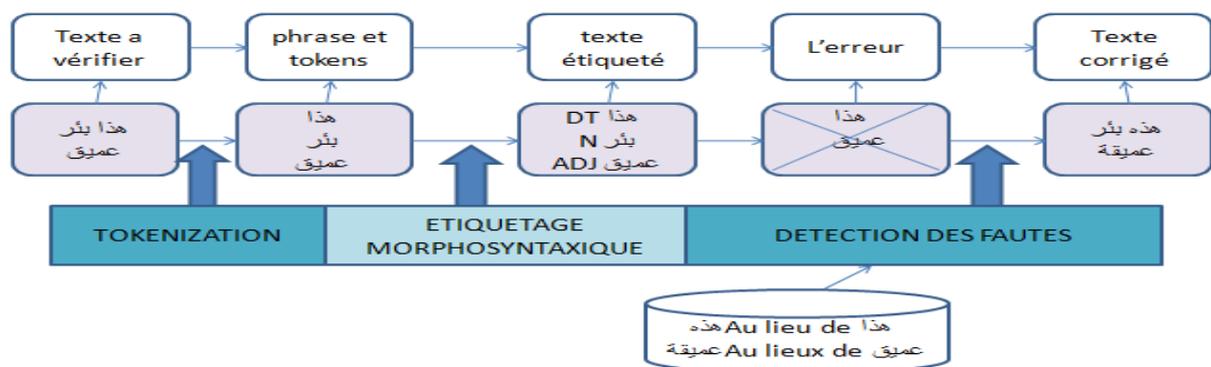


FIGURE 3.5 – Exemple d'erreur d'un pronom démonstratif

Cet exemple se présente dans le but de corriger les erreurs concernant les mots considérées comme des mots masculins et pas féminins, dont l'erreur détectée est au niveau du token " هذا " et " عميق " puisque ce dernier est l'adjectif de mot féminin " بئر " .

Le premier traitement consiste à découper la phrase « هذا بئر عميق » en mots, ou plus exactement en tokens هذا | بئر | عميق . C'est la segmentation, ou tokenization, réalisée par le tokenizer. Un étiquetage morphosyntaxique s'effectue ensuite, chaque mot se voit attribuer l'étiquette (tag) correspondante.

- هذا —>Déterminant (DT)
- بئر —>Nom féminin(N)
- عميق —>Adjectif (ADJ)

Une fois l'étiquetage terminé, commence alors l'étape de vérification de la grammaire, le correcteur compare le texte entré au modèle de règle d'erreur de sa base. Donc il propose de remplacer le token « هذا » par « هذه » et le token « عميق » par « عميقة ».

5) Erreurs de nombre et l'énuméré (العدد و المعدود)

la figure ?? exprime les différentes catégories de (العدد و المعدود).

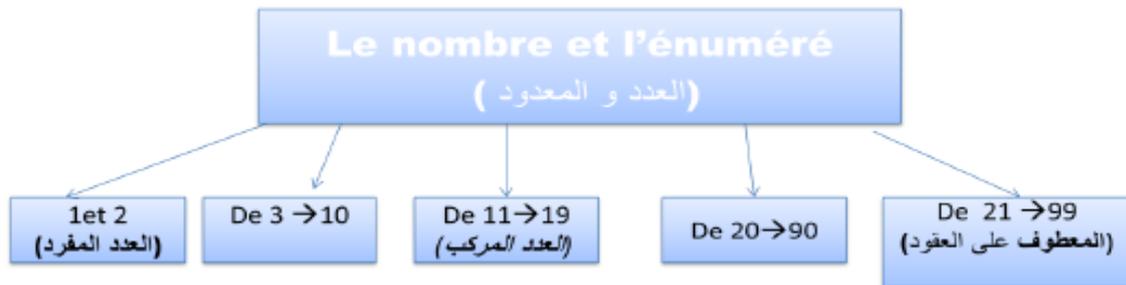


FIGURE 3.6 – les catégories de nombre et l'énuméré

Voilà les règles de nombre et l'énuméré (العدد و المعدود) :

<p><u>Règle de nombre 1 et 2 :</u> Si l'énuméré → masculin Alors Nbr → masculin Si l'énuméré → féminin Alors Nbr → féminin</p>	<p><u>Règle de des nombres de 3 à 10 :</u> Si l'énuméré → masculin Alors Nbr → féminin Si l'énuméré → féminin</p>	<p><u>Règle des nombres de 11 à 19 :</u> 11 et 12 même règle avec 1 et 2 <u>Règle de 13 → 19 :</u> Si l'énuméré → masculin Alors Partie1 → féminin Et Partie2 → masculin Si l'énuméré → féminin Alors Partie 1 → masculin Et Partie2 → féminin</p>
---	--	--

Le tableau ?? exprime quelques exemples selon les règles de nombre et l'énuméré :

Exemples	العدد و المعدود Cas
- حضر رجل واحد - رأيتُ امرأتين اثنتين	Le nombre singulier (العدد المفرد)
- صعد ثلاثة أطفال الطائرة - ست طالبات - حضر المجلس ثلاثة رجال	Les nombres de (3 à 10)
- إحدى عشرة طالب - ثلاثة عشر طالبا	Les nombres composés (العدد المركب) 11 et 12 13 —>19

TABLE 3.4 – Exemple de règles de nombre et l'énuméré [Abou obeida, 2007].

3.4.2 Erreurs de verbe transitifs :

Le verbe transitif est suivis d'un complément d'objet, il peut être suivis par : "harf-jar", verbe lui-même, d'un ou plusieurs complément d'objet direct. Comme ils sont exprimé dans la figure ??.

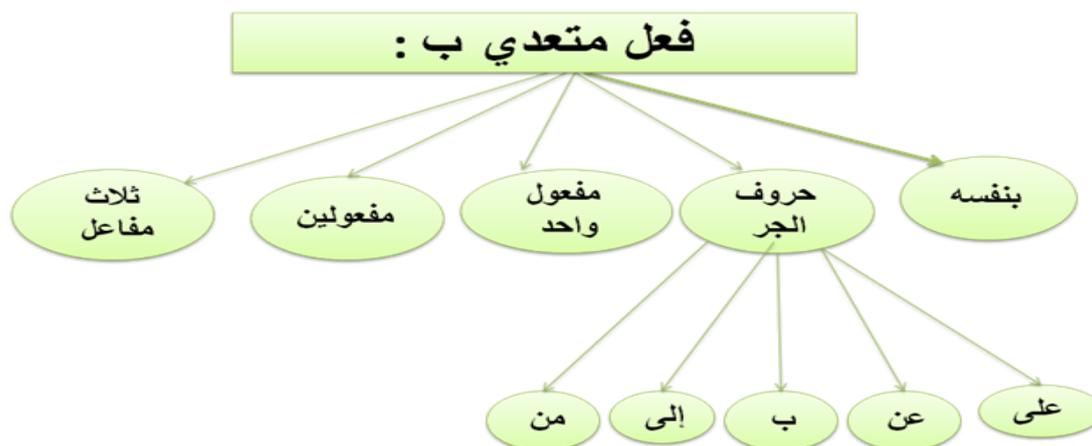


FIGURE 3.7 – Les types de verbe transitifs

•Exemples d'erreur بحرف المتعدي بحرف

Explications	Phrase corrigée	Phrase erronée
تعدي الفعل أجاب بحرف الجر عن	أجاب سؤاله، أجاب عن سؤاله	أجاب على سؤاله
لأن الفعل احتاج يجب أن يتعدى بـ آ إلى أو اللام	احتجتُ إلى بعض الكتب	احتجتُ بعضَ الكتبِ
تعدي الفعل وثق بحرف الجر الباء	أثق بك	أثق فيك

TABLE 3.5 – Exemple de verbe transitif (harf jar) [Mahmoud I,1998]

Exemple selon l'architecture

-L'erreur : يدل إلى

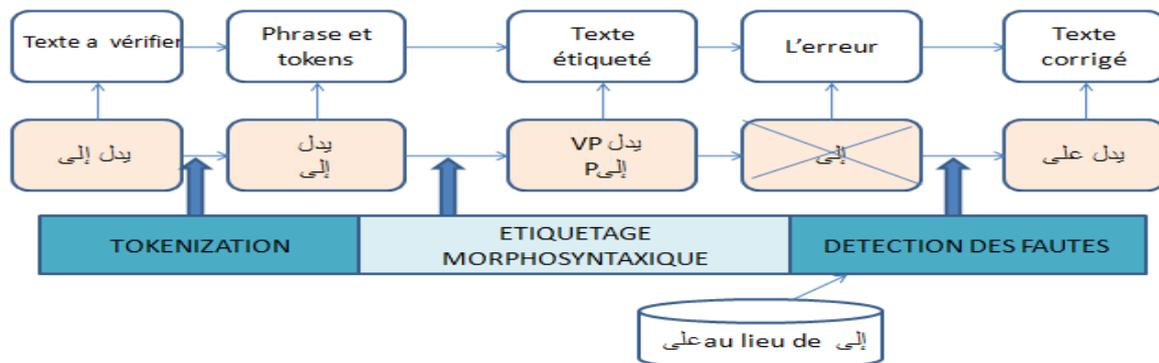


FIGURE 3.8 – Exemple d'erreur

L'architecture ci-dessus s'explique comment le correcteur signale une alerte au niveau du token erroné « إلى », et par la suite propose la préposition « على » comme une solution

d'après ce que la règle indique.

•Exemples d'erreur de الفعل المتعدي بنفسه

Phrase corrigée	Phrase erronée
ازدرى خصمه	ازدرى بخصمه
أتبع القول الفعل	أتبع القول بالفعل
امثل أمره	امثل لأمره
حاز فلان الشهادات والأموال	حاز فلان على الشهادات والأموال

TABLE 3.6 – Exemple de verbe transitif (bi nafsihi) [Mahmoud I,1998]

3.4.3 Erreurs lors de la liaison :

-L'erreur الواو قبل أسماء الصلة

Phrases corrigées	Phrases erronées
ذهبت الفتاة التي كانت عندنا	ذهبت الفتاة والتي كانت عندنا
أحب الشعر خاصة الشعر العمودي	أحب الشعر وخاصة الشعر العمودي
الحياة ممتعة لكنها متعبة	الحياة ممتعة ولكنها متعبة

TABLE 3.7 – Exemple d'erreurs lors de traduction [10]

3.4.4 Erreur de pluriels :

-Erreurs de pluriels جمع التفسير

Corrections	Erreurs
مشروعات	مشاريع
منشورات	مناشير
مكتوبات	مكاتيب
محال	محلات

TABLE 3.8 – Exemple d’erreurs de pluriels [Adnani M, 1983]

-L’exemple d’erreur de pluriel selon l’architecture : محلات بيع الأواني

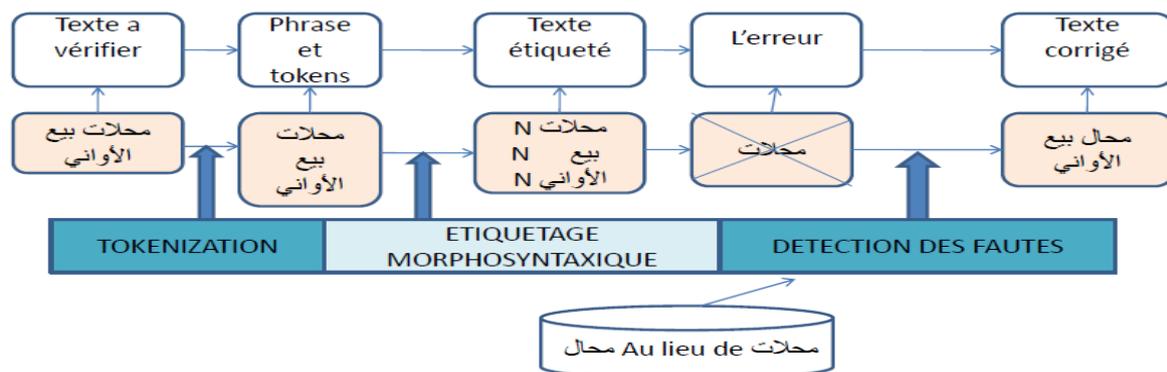


FIGURE 3.9 – Exemple d’erreur de pluriel

Cette catégorie contient des exemples hétérogènes dont on ne peut pas les traiter tous ensemble.

Et de même manière, la démarche reste ressemblable aux autres cas précédents, mais au niveau de réalisation, chaque règle aura une syntaxe différente.

3.4.5 Erreurs de (EL NAWASIKH) :

كان و أخواتها-

إذا دخلت كان و أخواتها على الجملة الاسمية (مبتدأ و خبر) يرفع المبتدأ و ينصب الخبر

Phrases corrigées	Phrases erronées
كان الجو صافيا	كان الجو صافي
أصبح الماء راكدا	أصبح الماء راكد
كان الطالبان مجتهدين	كان الطليين مجتهدين

TABLE 3.9 – Exemple d'erreurs (EL NAWASIKH)

3.4.6 Erreurs de prononciation :

(التحريك والتسكين وإبدال الحركات بعضها مكان بعض) -

Le tableau ?? suivant exprime les erreurs agrammatical exactement les erreurs de la prononciation que les locuteurs font généralement.

Phrases corrigées	Phrases erronées
قطع الشيء إِرْبًا إِرْبًا	قطع الشيء إِرْبًا إِرْبًا
(دُولية (بضمّ وفتح	(منظمة دُولية (بفتح وسكون
يرميل النَّفط	يرميل النَّفط
زادَ الطَّيْنَ بِلَّة	زادَ الطَّيْنَ بِلَّة
(الحراك (بفتح الحاء	(الحراك (بكسر الحاء
(مدينة جدة (بضم الحيم	(مدينة جدة (بفتح الحيم

TABLE 3.10 – Exemple d'erreurs de la prononciation [Adnani M, 1983]

3.4.7 Erreurs sémantiques :

Une erreur sémantique peut être une erreur contextuelle ou une erreur de choix de mot erronée. Lorsqu'un mot mal saisi est un mot réel dans la langue, il n'est pas détecté

comme une faute d'orthographe, mais il ne correspond pas au contexte donné.
de telles erreurs sont appelées erreurs contextuelles.

Explications	Corrections erronées	Erreurs
اعتبر معناه أخذ العبرة والعظة، والتعجب، والتأمل	أعدُّ الأمر متهياً	اعتبر الأمر متهياً
شيوغ استخدام الفعل أحنى بمعنى أخفض لأن أحنى من الحنو أي العطف والحنان	حنى الرجل رأسه احتراماً	أحنى الرجل رأسه احتراماً
أنَّ السوي في اللغة هو المستقيم المعتدل	ذهبنا معاً	ذهبنا سويّاً
الصواب استخدام مصادفة لأنها مصدر الفعل صادف الذي يعني الالتقاء، أما الصدفة فهي من الفعل صدف ومعناه أعرض عنه	مصادفة	صدفة
يشيع على الألسنة استخدامه بمعنى شارك وهذا خطأ شائع لأنَّ الفعل ساهم معناه اقترح	أسهم الطلاب في تنظيف المدرسة	ساهم الطلاب في تنظيف المدرسة

TABLE 3.11 – Exemple d'erreurs sémantiques [10]

3.5 Conclusion

Nous avons pu à travers ce chapitre présenté notre conception globale dédiée par l'étude des différentes catégories d'erreurs commune en langue arabe en commençant par la segmentation des phrases en mots (tokens) avant de passer à l'étiquetage de ces tokens et en fin on passe à la phase de détection des erreurs de grammaire.

Nous consacrons maintenant le chapitre suivant à l'implémentation des différentes procédures de notre projet, ainsi à l'évaluation des résultats obtenus.

L'implémentation

4.1 Introduction

Dans ce chapitre, nous allons décrire les différentes fonctionnalités de notre approche de modélisation des erreurs grammaticales pour le logiciel libre « Language Tool ».

Nous présentons dans ce chapitre l'implémentation, le prototype informatique suivi, et l'expérimentation de notre approche, ainsi les résultats d'évaluation de notre application que nous avons développée.

4.2 Implémentation de l'approche

Dans cette partie, nous décrivons d'abord notre choix de logiciel ensuite nous citons les principales scripts développées et décrire les résultats de l'application tout en détaillant leurs fonctionnalités.

4.2.1 Choix du logiciel

La programmation d'un logiciel de correction grammaticale est un travail complexe nécessite plusieurs années de développement. Dans notre recherche nous avons choisis un logiciel libre "LanguageTool" tel qu'il permet :

- L'accès aux sources et la licence permissive permettent une intégration rapide et simple.
- L'ajout des règles facilement par les non informaticiens à travers la page web communauté.

- De l'utiliser soit en ligne soit depuis l'ordinateur sans avoir besoin de connexion à Internet.
- De l'installer sous forme d'applications pour OpenOffice et LibreOffice.

4.3 L'implémentation des règles

Languag tool permet de modéliser des erreurs sous forme d'un fichier XML utilisé pour vérifier les erreurs dans une phrase. Si un modèle déclaré dans la règle associe la phrase entrée, une erreur est signalée à l'utilisateur.

Le fichier de règles, insensible à la casse par défaut, est composé de plusieurs catégories de règles qui peuvent couvrir mais ne se limitent pas aux erreurs d'orthographe, de grammaire, de style et de ponctuation.

Chaque catégorie de règle est composée d'une ou de plusieurs règles ou groupes de règles. Chaque règle est composée de différents éléments et attributs.

voile le schéma suivant exprime les éléments de base d'une règle en XML .

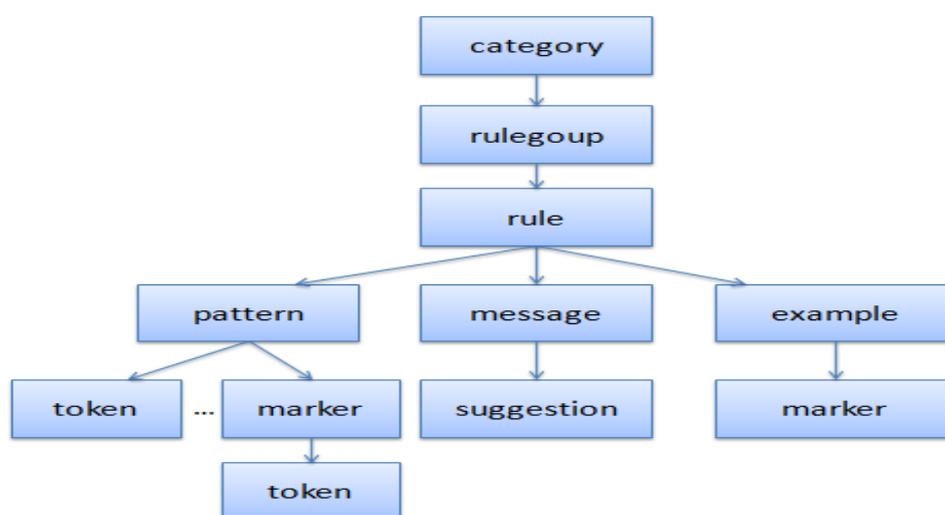


FIGURE 4.1 – Les éléments de base d'une règle en XML.

Les trois (03) éléments de base d'une règle sont :
modèle, message et exemple.

1. L'élément de modèle est l'endroit où l'erreur à faire correspondre est déclarée.
2. L'élément d'exemple est l'emplacement où des exemples incorrects et corrects sont déclarés.

3. L'élément de message est l'endroit où le retour d'informations et la suggestion, le cas échéant, sont déclarés.

L'élément racine de l'ensemble de règles est «rules». L'élément racine «rules» a deux enfants nommés «rule» et «rulegroup».

«rule» comporte trois sous-éléments : «**pattern**», «**message**» et «**exemple**».

- <**rulegroup**> contient un ensemble de règles contenant l'ID du groupe de règles.
- <**rules**> est l'élément racine des éléments à l'intérieur du XML qui contient les règles de grammaire.
- <**rule**> est le sous-élément de l'élément racine. Il définit chaque modèle de règles avec son message et ses exemples. Il a un identifiant et un nom d'attribut.

L'identificateur identifie de manière unique chaque règle et permet d'activer et de désactiver les règles séparément, le cas échéant. L'attribut name décrit la règle.

Dans le formalisme XML, l'élément <rule> se présente ainsi :

```
<rule id="inshalla" name="إن شاء الله">
...
</rule>
```

L'identifiant et le nom sont utilisés pour un usage interne.

- <**message**> est un autre sous-élément de <**rule**> il contient une description de la règle actuelle déclarée comme parent du <**message**>.

En revanche il peut contenir des éléments <**suggestion**>, dont les balises entourent des mots qui seront suggérés en correction.

```
<message>قل جمادى<suggestion>الأولى</suggestion> مؤنث</message>
```

- <**example**> Il s'agit d'un élément servant à donner des exemples d'énoncés corrects ou incorrects, en rapport avec l'erreur détectée par le motif de la règle. Il contient un attribut type indiquant le type d'exemple qu'il donne et qui a donc

pour valeur soit "correct" soit "incorrect".

L'exemple peut également contenir un ou plusieurs autres éléments **<marker>**, qui permettent, lors de l'affichage, de mettre en évidence le ou les mots que les balises de cet élément encadrent [A. Souque, 2014].

```
<rule >
<pattern>
<token >...</token>
</pattern >
<example correction="الأخرة">شهر جمادى <marker>الأخر</marker></example>
</rule >
```

- **<token >** Les tokens du motif sont décrits dans autant d'éléments **<token >**. Chacun contient plusieurs attributs facultatifs : [A. Souque, 2014]

```
<rule >
<pattern>
<token >قطع</token>
<token >الشيء</token>
</pattern >
</rule >
```

- **<regexp>** signifie expression régulière (regular expression). Cet attribut, s'il a la valeur "yes", permet de mettre plusieurs mots dans un même **<token >**, en les séparant avec la barre verticale "|", [A. Souque, 2014] .

```
<rule >
<pattern>
<token regexp="yes">المراجع|البنز|الريح|الساق|الفخذ|الكتف</token>
</pattern>
...
</rule>
```

4.4 Développement des règles d'erreurs

Dans cette partie nous avons cité quelques exemples en XML d'implémentation des règles modéliser dans le chapitre de conception.

4.4.1 L'accord de genre et nombre

Les exemples suivants concernent la catégorie des règles d'accord de genre et nombre. Nous avons implémenté selon les règles d'accord suivants : de pronom relatif, pronom démonstratif, de l'adjectif et de nombre et l'énuméré avec langage XML.

a). L'accord de Pronom Démonstratif (الاسم الإشارة)

la règle : si $Nom \rightarrow NFS$ Alors $Det \rightarrow DFS$

nous aide à implémenter l'exemple suivant, qui montre l'accord de chaque déterminant avec son nom qu'il convient et l'utilisation des expressions régulières qui nous permet de regrouper plusieurs tokens en une seule règle.

```

<category id="ERREUR_accord" name="الربط" >
  <rulegroup id="demanstratif" name="اسم الإشارة" >
    <rule>
      <pattern>
        <marker><token postag="DMS">هذا </token></marker><token
        regexp="yes">المراجع |الريح |الفخذ |الغاس |الكأس |البئر |البيتر |جهنم |الكف |
        </token>
      </pattern>....
      <message>هذه <suggestion>هذه </suggestion>؟</message>
      <example correction="هذه"><marker>هذا </marker></example>
    </rule>
  </rulegroup>
</category>

```

FIGURE 4.2 – Exemple d'implémentation d'un pronom démonstratif.

Le 2ème token contient plusieurs mots qu'on doit changer leur déterminant « هذا » par

« هذه ».

Donc l'expression régulière sert à réduire le travail en rassemblant les mots dans une seule balise <token>.

La balise <marker> fait un trait sous les tokens qu'elle contient lors de l'exécution, généralement sont les tokens erronés qu'on doit les remplacer avec ses corrections.

Cette règle fait pouvoir le correcteur d'afficher un message descriptif du motif d'erreur, plus une suggestion et enfin un exemple de type « correct » pour remplacer les fautes .

•**Résultat :**

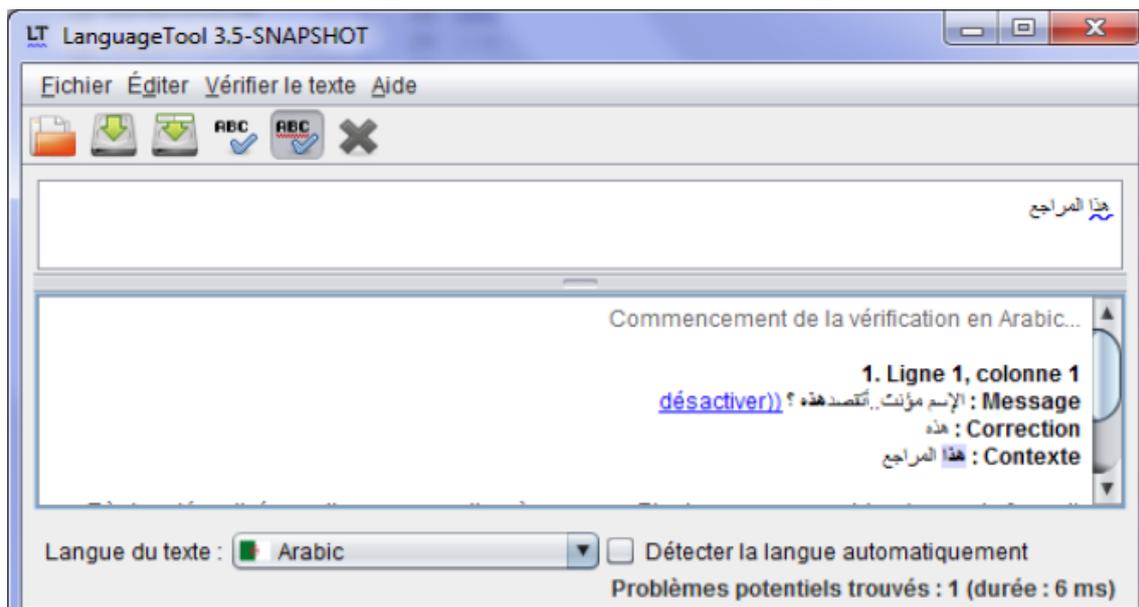


FIGURE 4.3 – Résultat d'un pronom démonstratif.

b). L'accord de pronom relatif (اسم الموصول)

La règle suivant : Si Nom \rightarrow NMS Alors Det \rightarrow DMS.

montre que le pronom relatif s'accord en genre et en nombre .

l'erreur dans cet exemple se trouve dans le token « التي », le token « النشاط » est un nom masculin singulier et le token « التي » est un déterminant féminin singulier, donc la correction est de remplacer le mot « التي » par le mot « الذي » .

```

<category id="ERREUR_accord" name="التطابق " >
<rulegroup id="PRONOMRELATIF" name="الاسم الموصول " >
<rule>
<pattern>
<token postag="NMS" >النشاط </token>
<marker><token postag="DFS" >التي </token></marker>
</pattern>
<message>أنت قصد <suggestion>الذي </suggestion>؟</message>
<example correction="الذي" >النشاط <marker> التي </marker> /example>
</rule>
</rulegroup></category>

```

FIGURE 4.4 – Exemple d'implémentation d'un pronom relatif.

•Résultat :



FIGURE 4.5 – Résultat d'implémentation d'un pronom relatif.

c).L'accord d'adjectif (النعته)

Plusieurs écrivains et locuteurs de la langue arabe pensent que le mot « **خنصري** » est un nom masculin, par contre, les grammairiens arabes le classifient comme un nom féminin, et comme la règle dit : Si NomQualifie —>féminin Alors Adjectif —>féminin donc l'adjectif doit compatible avec le nom qui la précède la correction de l'erreur de cet

exemple « خنصري مجروح » est « خنصري مجروحة » .

```
<category id="ERREUR_accord" name="التطابق" >
<rulegroup id="Adjectif" name="التعت">
<rule>
<pattern>
<token>خنصري</token>
<marker><token>مجروح</token></marker>
</pattern>
<message>أنتقصد <suggestion>مجروحة</suggestion>؟</message>
<example correction=مجروحة> خنصري <marker>مجروح</marker></example>
</rule>
</rulegroup></category>
```

FIGURE 4.6 – Exemple d'accord d'adjectif en XML.

•Résultat :

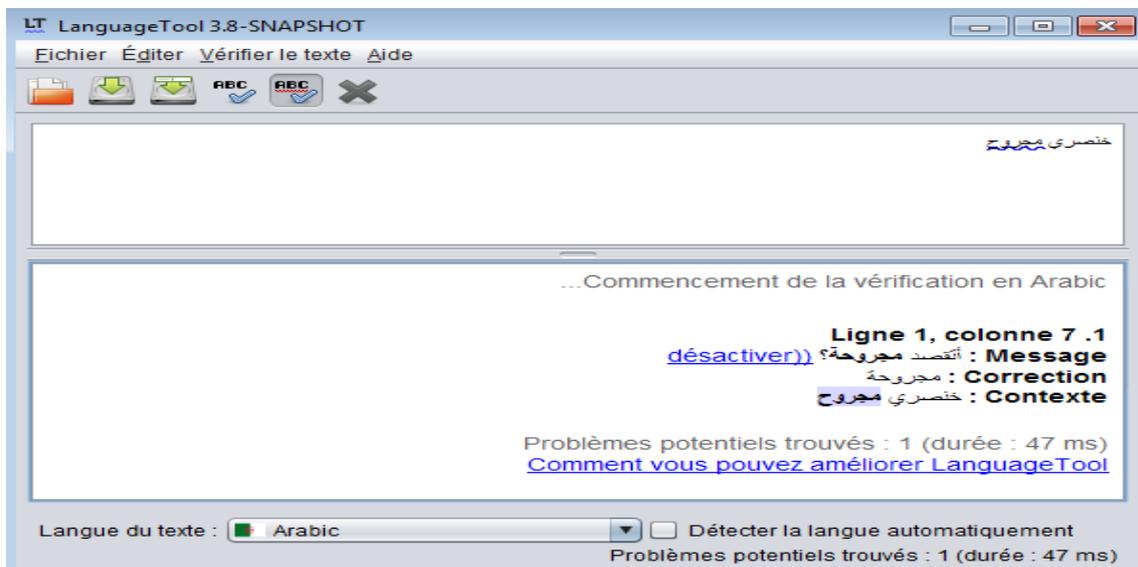


FIGURE 4.7 – Résultat d'accord de l'adjectif.

d).L'erreur de nombre et l'énuméré

La règle de (العدد و المعدود) dans la catégorie de (3 à 10) dit :

si l'énuméré —->masculin alors le nombre —->féminin.

Dans cet exemple l'énuméré « شهور » masculin et le nombre « ثلاث » aussi masculin. alors d'après la règle le token « ثلاث » est une erreur on doit le remplacer par le nombre féminin « ثلاثة » .

```

<category id="ERREUR_accord" name="التطابق" >
<rulegroup id="NUMBERSERRORS" name="الأخطاء المتعلقة بالأعداد" >
<rule>
<pattern>
<marker><token>ثلاث</token></marker>
<token postag="NMP">شهور</token> </pattern>
<message>أنت قصد <suggestion>ثلاثة</suggestion>?</message>
<example correction="ثلاثة"> <marker>ثلاث</marker></example>
</rule>
</rulegroup></category>

```

FIGURE 4.8 – Exemple d'implémentation de nombre et l'énuméré en XML.

•Résultat :



FIGURE 4.9 – Résultat d’implémentation de nombre et l’énuméré.

4.4.2 Le verbe transitif

Le verbe transitif est suivis d’un complément d’objet, ce dernier peut être : ”harf jar” , d’un verbe en lui-même et d’un ou plusieurs complément d’objet direct.

Cet exemple montre l’erreur dans le cas « فعل متعدي بنفسه » où l’erreur se trouve dans le token « لأمره » d’où la correction est de supprimer le caractère « ل » par ce que le verbe « امثل » « يتعدى بنفسه، لا باللام »

```

<category id="ERREUR_verbe " name=" فعل متعدي " >
<rulegroup id="Vtransitif" name=" فعل متعدي بنفسه ">
<rule>
<pattern>
<token inflected="yes ">امتثل</token>
<marker><token inflected="yes ">لأمره</token></marker>
</pattern>
<message> فالفعل "امتثل" يتعدي بنفسه، لا باللام <suggestion>أمره</suggestion> قل يمتثل
</message>
<example correction="أمره">امتثل<marker>لأمره</marker></example>
</rule> </ rulegroup >
</category>

```

FIGURE 4.10 – Exemple d'implémentation d'un verbe transitif.

•Résultat :



FIGURE 4.11 – Résultat d'implémentation d'un verbe transitif.

4.4.3 L'erreur de pluriel

Plusieurs écrivains et locuteurs de la langue arabe pensent que le mot «إطارات» est le pluriel du mot «إطار» par contre, les grammairiens arabes le classifient dans le pluriel (جمع التفسير), cet exemple pour faire corriger l'erreur de token connu «إطارات» par «أطر».

```
<category id="PLURIEL" name="الجمع">
<rulegroup id="PLURIELENTERN " name=" جمع التفسير ">
<rule>
<pattern>
<marker><token>إطارات</token></marker>
</pattern>
<message>صيغة الجمع غير صحيحة: جمع إطار من جمع التفسير.. أنتصد < suggestion>أطر</suggestion> ?</message>
<example correction="أطر"> <marker>إطارات</marker></example>
</rule>
</rulegroup>
</category>
```

•Résultat :



FIGURE 4.12 – Résultat d'implémentation de l'exemple de pluriel

4.4.4 L'erreur sémantiques

(الخطأ الدلالي)-

Cet exemple montre l'erreur sémantique qui rend la phrase insensée ou absurde, une erreur sémantique dans le token «سَاهم» peut être une erreur contextuelle ou une erreur de choix de mot erronée. Le verbe «سَاهم» veut dire «اقترع» Le mot est mal placé, il ne correspond pas au contexte donné. Donc on remplace le token «سَاهم» par «أسهم» .

```

<category id="Erreurs_Semantique " name="خطأ دلالي " >
<rule>
  <pattern>
    <marker><token>سَاهم</token></marker>
    <token>في</token>
  </pattern>
  <message>الفعل (سَاهم) معناه (اقترع)</message>
  <example correction="أسهم"><marker>سَاهم</marker> </example>
</rule>
</category>

```

FIGURE 4.13 – Exemple d'implémentation d'erreur sémantique

•Résultat :



FIGURE 4.14 – Résultat d’implémentation d’erreur sémantique

4.4.5 L’erreur de prononciation

التحريك والتسكين وإبدال الحركات بعضها مكان بعض -

Cette catégorie concernant les erreurs agrammaticale qui sont des mots qui s’écrivent de la même manière, mais leurs prononciation est différente . Dans l’exemple suivant le token « حراك » est marqué comme une erreur par ce que plusieurs locateurs pensent que le mot « حراك » est prononcé avec (كسر الحاء), la correction est de le prononcer avec (فتح الحاء).

```

<category id="tachekile " name="التحريك والتسكين وإبدال الحركات بعضها مكان بعض" >
  <rule>
    <pattern>
      <marker><token>حراك</token></marker>
    </pattern>
    <message>حَرَكَ (فتحتين مُتتاليتين)</message>
    <example correction="حَرَكَ " > <marker>حراك</marker> </example>
  </rule>
</category>

```

FIGURE 4.15 – Exemple d'implémentation d'erreur de prononciation.

•Résultat :



FIGURE 4.16 – Résultat d'implémentation d'erreur de prononciation.

4.5 Conclusion

La correction grammaticale est une fonctionnalité très demandée par les utilisateurs de traitement de texte.

Nous avons effectué un travail de recherche très intéressant sur la correction grammaticale et ses outils. Nous avons modélisé les erreurs grammaticale à l'outil "languageTool" , et

de proposé des corrections permettant d'améliorer la langue arabe par l'étude de certaines erreurs communes.

Conclusion générale

La langue est l'outil de communication par excellence. Pour nous comprendre, nous utilisons un vocabulaire unique, plus ou moins large selon les locuteurs, et nous construisons nos phrases selon une logique identique, selon une syntaxe. Ne pas respecter la syntaxe ou l'orthographe, c'est courir le risque d'être mal compris, de ne pas être lu ou écouté. Nul ne rédige un texte, qu'il s'agisse d'une note de service, d'un courrier, d'un article de journal ou d'un essai littéraire, sans respecter du mieux possible les règles de la grammaire.

Notre projet s'insère dans le cadre de travail sur les logiciels open source dont le code source est accessible au public et que les utilisateurs peuvent utiliser tel quel ou modifier. Notre objectif est donc de modéliser des erreurs courantes de la langue arabe dans ce type de logiciel qui est « LanguageTool » pour la correction grammaticale de cette langue. Les applications telles que les traitements de texte et autres outils d'écriture incluent généralement un vérificateur de grammaire. Le vérificateur de grammaire a pour but d'identifier les phrases grammaticalement incorrectes en fonction de la syntaxe du langage. Le vérificateur de grammaire qui a été proposé est un système basé sur des règles permettant d'identifier les phrases les plus susceptibles de contenir des erreurs. L'ensemble de règles est automatiquement généré à partir d'une partie d'un corpus étiqueté.

Tout d'abord, nous avons à travers une recherche étudiée en détail les différentes phases de cette démarche comme la segmentation, l'étiquetage morphosyntaxique et en fin la détection des erreurs qui basée sur des règles d'erreurs. Le résultat de cette recherche, nous a permis de dessiner un schéma pour la conception de notre approche de vérification.

Ensuite nous avons implémentée notre stratégie qui combine tous ces phases de vérification grammaticale.

Et pour Le développement d'un tel outil consistera alors également à compléter la base de règles. Dans ce but, il faut réaliser, de manière approfondie, l'analyse d'un corpus d'erreurs constitué. Nous allons extraire des fautes, dont nous définirons la typologie, et à partir desquelles nous créerons de nouvelles règles.

Et nous proposons d'ajouter le support et de modéliser les erreurs courantes de la langue thamazight pour logiciel "language tool" .

Bibliographie

a. Bibliographie :

[Adnani M ,1983] كتاب معجم يعالج الأخطاء اللغوية الشائعة و يبين صوابها مع الشرح و الأمثلة

[Augustin B, 2013] Implémentation d'un correcteur orthographique automatique pour l'extraction d'information, UNIVERSITE LIBRE DE BRUXELLES.

[Araibi & Abdlmoumene, 2016] L'automatisation de la segmentation thématique de textes arabes p11.

[Abou obeida, 2007] Les règles de grammaire du premier Livre de Médine.

[Boubekeur Y, 2016] Identification automatique de mots clés dans les textes arabes p 24,32.

[Charnet C, 1998] Le correcteur grammatical : un auxiliaire efficace pour l'enseignant ? Quelques éléments de réflexion, Université Paul Valéry, Montpellier III, France p3

[Chouchaoui & Brahimia, 2016] Détection Automatique De La Cohésion Lexicale Entre Phrases Dans Les Textes Arabes, Université de Djilali BOUNAAMA KHEMIS MILIANA

[Cheragui M, 2012] une analyse morphologique de la langue arabe basée sur l'aide multicritère à la décision université d'Adrar, p1.

[Dilekh T, 2011] Implémentation d'un outil d'indexation et de recherche des textes en arabe, thèse de magister, université Batna, p47, 49.

[Djoudi M, 1991] Contribution à l'étude et à la reconnaissance automatique de la parole en Arabe standard, Université de Nancy1, 12 Nov 1991.

[Echeteia A, 2009] العلاقات النحوية بين الخبر والصفة والحال

[Gardent C, 2011] traitement des langues naturelles CNRS/LORIA Nancy(France).

[Jacquet-Pfau C, 2001] correcteurs orthographiques et grammaticaux p4.

[Jensenetal, 1993] Karen Jensen, George E. Heidorn, Steph en D.Richards on (Eds.): Natural language processing: the PLNLP approach, Kluwer Academic Publishers, 1993.

[Atwell, 1987] Eric Atwell, Stephen Elliott: Dealing with ill-formed English text in: The computational analysis of English, Longman, 1987.

[Kali & Tchoketch, 2015] Correcteur open source des erreurs grammaticales communes en langue arabe UNIVERSITE DR. YAHIA FARES DE MEDEA.

[Madhvi S, 2018] A Systematic Review of Automated Grammar Checking in English Language

[Naber D, 2003] A Rule-Based Style and Grammar Checker

[Mahmoud I, 1998] كتاب الأخطاء الشائعة في استعمالات حروف الجر

[Motasem, Ghassan & Djou, 2004] Filtrage sémantique de textes en arabe en vue d'un prototype de résumé automatique, Université Paris – Sorbonne, CNRS 96, Bd Raspail 75006 Paris – France p2.

[Shaalán K, 2005]. Arabic GramCheck: a grammar checker for Arabic.

[Souque A, 2014] Modèle de vérification grammaticale automatique gauche-droite. Linguistique, Université Grenoble Alpes, P22, 27, 31.

[Souque A, 2007] CONCEPTION ET DÉVELOPPEMENT D'UN FORMALISME DE CORRECTION GRAMMATICALE AUTOMATIQUE, UNIVERSITÉ STENDHAL – GRENOBLE 3, LABORATOIRE LIDILEM, Juin 2007

b. Webographie :

[2] <https://www.cairn.info/revue-francaise-de-linguistique-appliquee-2001-2-page-81.html>
consulter le 25/06/2019

[1] <http://www.technolanguage.net/imprimer.php3?Le> 25/05/2019

[3] <https://extensions.openoffice.org/en/project/lightproof-grammar-checker-development-framework> le 15/09/2019

[4] <https://tahadz.wordpress.com/2011/06/20/alightproof/> consulter le 10/10/2019

[5] : <https://languagetool.org/fr/> consulter le 15/11/2019

[6] : <https://www.languagetool.org/usage/> consulter le 15/15/2019

[7] <https://fr.slideshare.net/othmendhifallah/etiquetage-morphosyntaxique-de-larabe-avec-nooj> Consulté le 15/07/2019

[8] <http://al-dirassa.com/le%3%A7on/la-phrase-nominale-et-la-phrase-verbale-en-arabe/>
Consulté le 15/07/2019

[9] <https://blogs.transparent.com/arabic/arabic-sentence-structure-nominal-and-verbal-sentences/> Consulté le 15/07/2019

[10] <https://mahmoudqahtan.com>

Résumé

L'arabe est une langue sémitique riche en morphologie et en syntaxe. Les règles grammaticales très nombreuses et complexes de la langue peuvent être déroutantes pour l'utilisateur moyen d'un traitement de texte. Dans ce mémoire, nous rapportons notre tentative de développer un vérificateur grammaticale open source pour l'arabe standard moderne, appelé "Language Tool".

"Language tool" peut aider l'utilisateur moyen en vérifiant dans son écriture certaines erreurs de grammaire courantes. il décrit le problème pour lui / elle et propose des suggestions d'amélioration. L'utilisation d'un vérificateur grammatical arabe peut augmenter la productivité et améliorer la qualité du texte écrit en arabe. "Language Tool" a été implémenté avec succès à l'aide des règles écrite en langage XML. Des tests réussis ont été effectués à l'aide d'un corpus.

Mots clés : Traitement du langage naturel arabe, vérification grammaticale, erreurs courantes de grammaire arabe, vérificateurs de grammaire ...

Abstract

Arabic is a Semitic language rich in morphology and syntax. The very numerous and complex grammatical rules of the language can be confusing for the average user of a word processor. In this memory, we report our attempt to develop an open source grammar checker for modern standard Arabic, called "Language Tool".

"Language tool" can help the average user by checking some common grammar errors in their writing. he describes the problem for him / her and offers suggestions for improvement. Using an Arabic grammar checker can increase productivity and improve the quality of text written in Arabic. "Language tool" has been successfully implemented using the rules written in XML. Successful tests were performed using a corpus.

Key words : Arabic natural language processing, grammatical verification, common mistakes of Arabic grammar, grammar checkers. ...