A Morphological - Syntactical Analysis Approach For Arabic Textual Tagging

Shihadeh Alqrainy

Thesis submitted in partial fulfillment
of the requirements for the Degree of
Doctor of Philosophy
in Computer Science

School of Computing
Faculty of Computing Sciences and Engineering
De Montfort University

July, 2008

Abstract

Part-of-Speech (POS) tagging is the process of labeling or classifying each word in written text with its grammatical category or part-of-speech, i.e. noun, verb, preposition, adjective, etc. It is the most common disambiguation process in the field of Natural Language Processing (NLP). POS tagging systems are often preprocessors in many NLP applications.

The Arabic language has a valuable and an important feature, called *diacritics*, which are marks placed over and below the letters of the word. An Arabic text is partially-vocalised[1] when the diacritical mark is assigned to one or maximum two letters in the word.

Diacritics in Arabic texts are extremely important especially at the end of the word. They help determining not only the correct POS tag for each word in the sentence, but also in providing full information regarding the inflectional features, such as tense, number, gender, etc. for the sentence words. They add semantic information to words which helps with resolving ambiguity in the meaning of words. Furthermore, diacritics ascribe grammatical functions to the words, differentiating the word from other words, and determining the syntactic position of the word in the sentence.

[1] Vocalisation (also referred as diacritisation or vowelisation).

This thesis presents a rule-based Part-of-Speech tagging system called AMT - short for Arabic Morphosyntactic Tagger. The main function of the AMT system is to assign the correct tag to each word in an untagged raw partially-vocalised Arabic corpus, and to produce a POS tagged corpus without using a manually tagged or untagged lexicon (dictionary) for training. Two different techniques were used in this work, the *pattern-based technique* and the *lexical and contextual technique.*

The rules in the *pattern-based technique* technique are based on the pattern of the testing word. A novel algorithm, Pattern-Matching Algorithm (PMA), has been designed and introduced in this work. The aim of this algorithm is to match the testing word with its correct pattern in pattern lexicon.

The *lexical and contextual technique* on the other hand is used to assist the *pattern-based technique* technique to assign the correct tag to those words not have a pattern to follow. The rules in the *lexical and contextual technique* are based on the character(s), the last diacritical mark, the word itself, and the tags of the surrounding words.

The importance of utilizing the diacritic feature of the Arabic language to reduce the lexical ambiguity in POS tagging has been addressed. In addition, a new Arabic tag set and a new partially-vocalised Arabic corpus to test AMT have been compiled and presented in this work. The AMT system has achieved an average accuracy of 91%.

Contents

Abstract i

Dedication xii

Acknowledgments xiii

Publications xv

Index of Transliteration xvi

1 Introduction 1

1.1 Overview 1

1.2 Motivation 6

1.3 Research Hypothesis, Aims and Objectives 6

1.4 Significant Research Contributions 8

1.5 Outline of the thesis 10

2 Related Concepts and Literature Review 12

2.1 Part-of-Speech (POS) Tagging Problem 13

2.2 Applications of Part-of-Speech Tagging 17

2.3 Corpus-based Linguistics 19

2.3.1 Introduction 19

2.3.2 Existing Corpora : English and other languages 20

2.3.3 Arabic language corpora 23

2.4 Part-of-Speech tag set 25

2.4.1 Introduction 25

2.4.2 English and other languages POS tag sets 26

2.4.3 Arabic POS tag sets 29

2.4.4 Justification for a new Arabic tag set 32

2.5 Part-of-Speech Tagging Approaches 36

2.5.1 Rule-based Approach 37

2.5.2 Statistical Approach 44

2.5.3 Advantages and disadvantages of rule-based and statistical approaches 52

2.5.4 Hybrid and Other approaches 53

2.5.5 Arabic POS Tagging Systems 55

2.6 Chapter Summary 63

3 Arabic Language and POS tagging 65

3.1 Introduction 65

3.2 Arabic script and diacritics feature 68

3.2.1 Brief history 68

3.2.2 Arabic Diacritical Marks 70

3.3 Importance of the diacritic feature in Arabic POS tagging 71

3.4 Arabic Major grammatical part-of-Speech 74

3.4.1 Verb 75

3.4.2 Noun 76

3.4.3 Particle 77

3.5 Arabic Grammatical System 78

3.5.1 Morphology System 79

3.5.2 Syntax System 80

3.6 Chapter Summary 82

4 Tag set Design **83**

4.1 Tag set design criteria 83

4.2 Arabic Inflectional Features 87

4.2.1 Gender 88

4.2.2 Number 89

4.2.3 Person 89

4.2.4 Mood 89

4.2.5 Case 90

4.2.6 State 90

4.3 ARBTAGS-The developed Tag set 90

4.3.1 ARBTAGS Hierarchy 90

4.3.2 Tag design of ARBTAGS 99

4.3.3 Detailed and general tags in ARBTAGS tag set 102

4.4 Chapter Summary 107

5 Design and Implementation of AMT **109**

5.1 AMT Characteristics 109

5.2 Rule-based - the developed approach 110

5.2.1 Justification for using the rule-based approach 111

5.3 Pattern-based technique - A novel technique 112

5.3.1 Pattern-based Rules 116

5.3.2 Pattern-matching algorithm 120
5.4 Lexical and Contextual technique 126
5.4.1 Lexical Rules 127
5.4.2 Contextual Rules 128
5.5 A description of the tagger system 132
5.5.1 Tagger Modules 132
5.5.2 Tagging Process 133
5.6 Chapter Summary 135
6 Evaluation of Results obtained from AMT 137
6.1 Testing Data sets 137
6.2 AMT Experiments and accuracy measurement 139
6.2.1 Experiment-1 140
6.2.2 Experiment-2 141
6.2.3 Experiment-3 142
6.2.4 Experiment-4 143
6.3 Experimental results Analysis 145
6.3.1 The Quran text experiment 151
6.4 Summary of results obtained from the AMT system 153
6.5 Chapter Summary 154
7 Conclusion 156
7.1 Importance of diacritic feature 158
7.2 Contributions 159
7.3 Future Works 160
7.4 Summary 161

Bibliography 161

Appendices 178

A Tagset Appendices 179

A.1 General Tags 179

A.2 Detailed Tags 180

B The Arabic Language Orthography 191

B.1 Arabic words and the Roman alphabet 191

B.2 Arabic alphabet and other diacritical marks 192

C Lexical and Contextual Rules 195

C.1 Names and description of lexical rules 195

C.2 Lexical Rule Examples 195

C.3 Names and description of contextual rules 196

C.4 Examples used contextual rules 196

D Permission for Collecting Testing Corpus 197

List of Figures

2.1 Ambiguity types 14

2.2 The possible values of tag 15

2.3 ARBTAGS tag set hierarchy 33

2.4 Khoja tag set hierarchy [89] 34

2.5 Transformation-Based Error-Driven Learning. 43

2.6 How APT performs tagging 57

3.1 The origin of the Arabic script 68

3.2 The Arabic grammatical system 78

4.1 Categories of Arabic verb 91

4.2 Categories of Arabic noun 92

4.3 Categories of Arabic particle 99

4.4 Verb sub-classes and their inflectional features 102

4.5 Noun sub-classes and their inflectional features 103

4.6 Particle sub-classes 103

5.1 the word ويصَلخونَ and its pattern ويفَاعلونَ 115

5.2 the word غَافل and its pattern فَاعل 116

5.3 The identical letters between the word فكتبتنَ and the pattern فتفعلنَ . 122

5.4 Matching the word فكتبتنَ with the pattern ففعلتنَ 123

5.5 Matching the word يسمعونَ with the pattern يفَاعلنَ 124
5.6 Matching the word يسمعونَ with the pattern يفتعلنَ 125
5.7 Matching the word يسمعونَ with the pattern يفعلونَ 125
5.8 Matching the word ومنهَا with the pattern فعلهَا 131
5.9 Matching the word كتبهَا with the pattern فعلهَا 131
5.10 An overview of AMT 132
5.11 How AMT performs tagging 134
5.12 Tagging process for simple part of text 136

6.1 Success rate of experiment-1 141
6.2 Detailed and general tags ratio in experiment-1 141
6.3 Distribution of POS classes in experiment-1 141
6.4 Success rate of experiment-2 142
6.5 Detailed and general tags ratio in experiment-2 143
6.6 Distribution of POS classes in experiment-2 143
6.7 Success rate of experiment-3 143
6.8 Detailed and general tags ratio in experiment-3 144
6.9 Distribution of POS classes in experiment-3 144
6.10 Percentage of rules applicability based on type 144
6.11 Matching the word نموهَا with the pattern فعلهَا 148
6.12 Detailed and general tag ratio overall in the correctly tagged corpus . 150
6.13 Success rate for unvocalised sample text which contains 1500 words . 150
6.14 A sample of Quran text 151
6.15 The result of the Quran text 152

List of Tables

2.1 Ambiguous words in Arabic sentence 16

2.2 Sample of Brown tag set 26

2.3 Sample of LOB tag set 27

2.4 Sample of Penn Treebank tag set 28

2.5 Sample of Khoja tag set 30

2.6 The LDC POS tagset 31

2.7 Sample of Alshamsi and Guessoum tag set 31

3.1 Arabic short vowels diacritics 70

3.2 Nunation (Tanween) Vowels diacritics 70

3.3 Sukun and Shadda vowels 71

3.4 Vocalisation state of the Arabic word 71

3.5 Unvocalized Arabic sentence and its POS tags 72

3.6 The possible last diacritical mark (case ending) of the word حضرت . . 74

3.7 Partially-vocalised Arabic sentence and its correct POS tag 74

3.8 Samples of imperative verbs and their inflectional features 76

3.9 Samples of past tense (perfect) verb forms 79

3.10 Samples of present (imperfect) and imperative verb forms 80

3.11 Samples of additional forms such as verbal, diminutive, Adjective nouns created from the same simple root كتب 80

4.1 Personal pronouns between Arabic and English 96

4.2 Abbreviation symbols of the main POS classes 100

4.3 Abbreviation symbols of the sub-classes of class verb 100

4.4 Abbreviation symbols of the sub-classes of class noun 100

4.5 Abbreviation symbols the sub-classes of class particle 100

4.6 The possible value of the inflectional feature (Gender) 101

4.7 The possible value of the inflectional feature (Number). 101

4.8 The possible value of the inflectional feature (Person). 101

4.9 The possible value of the inflectional feature (Mood). 101

4.10 The possible value of the inflectional feature (Case). 101

4.11 The possible value of the inflectional feature (State). 102

4.12 Abbreviation symbols used in ARBTAGS tag set 104

4.13 ARBTAGS tag set vs. Penn Treebank tag set 106

4.14 ARBTAGS general tags 106

4.15 Sample of detailed tags in ARBTAGS 107

5.1 Derived forms from the ground form (root) 114

5.2 Sample of prefixes, forms, suffixes for some imperfect verb words . . 117

5.3 Sample of pattern lexicon shows the patten for some imperfect verb words 118

5.4 Sample of prefixes, forms, suffixes for some perfect verb words 118

5.5 Sample of pattern lexicon shows the patten for some perfect verb words 118

5.6 Number of identical letters between the word فكتبنَ and its patterns . 120

5.7 Number of identical letters between the word يسمعونَ and its patterns 124

5.8 Classification of Proper noun 130

5.9 Sample of prefixes, particle word, suffixes for some particles words . . 131

5.10 Sample of particles lexicon 131

6.1 Some of Quran words VS MSA words 153

B.1 Arabic Alphabet 193

B.2 Hamza (glottal stop) with Alif, waaw, and yaay consonants 194

B.3 Arabic short vowels 194

B.4 Other diacritical marks (Nunation,Sukun,gemination) in Arabic 194

Dedication

To my lovely wife, *who gave me unchanged affection, endless love, and constant encouragement over the years.*

To my children, *Ramzi, Dou'a, Iman, Ala'a and Malak for their patience, love, and for enduring the ups and downs during the completion of this thesis.*

This thesis is dedicated to them.

Acknowledgments

First of all, I would like to thank my God, who gave me the strength to finish this thesis. This thesis would not have materialised without the aid and collaboration of many people whom I wish to thank.

I would like to start with acknowledging my first supervisor, Dr. Aladdin Ayesh. He was always available with an accurate advice, an interesting suggestion or an encouraging word and a listening ear. His patience, guidance and help, both personally and professionally, have been greatly appreciated. I also deeply appreciate the dedicated support of the members of my supervision team, Prof. Robert John and Dr. John Cowell who guided the completion of this thesis with many helpful insights and valuable comments.

I am thankful to the members of my family, my mother, brothers and sisters for their long support. I also thankful my brothers in law. I would like to acknowledge the financial support afforded me by my brothers, Fathi Alqrainy, Dr.Saleh Abu-Soud and Husain Dolat during a very tough period of my life. The only word I remember at this point is *gratitude*.

I would also like to express my best thanks to my brothers, Walid Alqrainy and

Ibrahiem Abu farah for his continuing support. I would like to express my deepest gratitude to my closed friend, Hasan Alserhan, for his kind assistance with so many things through all the good times and especially through the bad times that we have spent together during our trip.

I am also thankful to all my friends in UK and Jordan. Last but not least, without the financial support of Albalqa'a Applied University in Jordan, my stay in the United Kingdom would have been even harder and this research would not have happened. Their contribution was certainly very helpful and to them I wish to extend my special thanks.

List of Publications

1. Shihadeh Alqrainy and Aladdin Ayesh. Developing a Tagset for Automated POS Tagging in Arabic .*WSEAS TRANSACTIONS on COMPUTERS* , 5(11):2787-2792, 2006.

2. Shihadeh Alqrainy and Aladdin Ayesh. Word Class Tagger and Tagset design for partial-vocalized Arabic Text. In *proceedings of 2nd Jordan International Conference on Computer Science and Engineering (JICCSE 2006), Albalqa'a Applied University, JORDAN, December 2006.*

3. Shihadeh Alqrainy and Aladdin Ayesh. Rule-based Part-of-Speech Tagger for Arabic. Submitted to *(ACM) Transactions on Asian Language Information Processing.*

Index of Transliteration

Arabic Alphabets[2]

No	Name	Con	Trans
1	Alif	ا	A
2	baa	ب	b
3	taa	ت	t
4	thaa	ث	th
5	jiim	ج	j
6	Haa	ح	H
7	khaa	خ	kh
8	daal	د	d
9	dhaal	ذ	dh
10	raa	ر	r
11	zaay	ز	z
12	siin	س	s
13	shiin	ش	sh
14	Saad	ص	S

No	Name	Con	Trans
15	Daad	ض	D
16	Taa	ط	T
17	DHaa	ظ	DH
18	ayn	ع	E
19	ghayn	غ	gh
20	faa	ف	f
21	qaaf	ق	q
22	kaaf	ك	k
23	laam	ل	l
24	miim	م	m
25	nuun	ن	n
26	haa	ه	h
27	waaw	و	w
28	yaay	ي	y

continued

[2] In Arabic Alphabets table : Con=Consonant, Trans=Transliteration

Hamza (glottal stop) and Ta Marboota Consonants.

Name	Consonant	Transliteration
Hamza	ء	'
hamza above Alif	أ	O
hamza below Alif	إ	I
hamza above waaw	ؤ	W
hamza above yaay	ئ	}
Ta Marboota	ة	p
Alif Maqsoura	ى	Y

Short Vowels Marks

Name	Mark in consonant	Transliteration	Pronunciation
Fatha sign	كَ	a	/a/
damma sign	كُ	u	/u/
kasra sign	كِ	i	/i/

Other diacritical marks (Nunation,Sukun,gemination)

Name	Mark in consonant	Transliteration	Pronunciation
Tanween fath	دً	an	/an/
Tanween damm	دٌ	un	/un/
Tanween kasr	دٍ	in	/in/
Sukun	دْ	x	
Shadda	بّ	=	

Chapter 1

Introduction

1.1 Overview

Natural Language Processing (NLP) is one of the Artificial Intelligence (AI) fields that deals with analysing, understanding and generating the human languages in order to interface with computers in both written and spoken contexts using natural human languages (e.g English, Arabic, French, etc.) instead of computer languages (e.g Java, C++, etc.)[1]. Understanding human languages is not an easy task for a computer that lacks the human knowledge of the world and the human experience with linguistic structures.

Multiple levels of knowledge are required to process the human language. The list below summaries some of the different form of knowledge relevant for natural language understanding([23], p.10) :

- *Phonological knowledge* : how words are related to the sounds that realise them.
- *Syntactic knowledge* : how words can put together to form sentences.

[1] For more : http://www.webopedia.com/TERM/N/NLP.html

- *Semantic knowledge* : the assignment of meaning to words in a sentence.
- *Morphological knowledge* : how words are constructed a smallest meaning units called morphemes. For example, the English word "*cats*" has two morphemes (*cat* and *s*).
- *Pragmatic knowledge* : how sentences are used in different situations .

This information is extremely necessary to resolve any type of ambiguity that may arise. In NLP a word, a phrase, or a sentence is called ambiguous if it can be reasonably interpreted in more than one way [33]. The ambiguity is arguably the single most important problem in NLP [66]. Natural language has a huge number of ambiguities at every level of description, such as, *lexical* (many words tend to have multiple lexical category[2] or senses), *syntactic* or *structural* (words having different structural functions in a sentence), and *semantic* (some sentences can have multiple interpretations) [40]. Ambiguity types in NLP is discussed in more detail in chapter 2.

The main goal of the NLP field is to resolve the ambiguity that may found in human language. Whether the ambiguity is lexical, syntactic or semantic, the disambiguation process is a central first step in most NLP tasks, such as machine translation, information retrieval, etc. [43].

The most common disambiguation process which has received extensive attention from NLP research community is Part-Of-Speech (POS) tagging. POS tagging is the process of labeling or classifying each word in written text with its part-of-speech, i.e. noun, verb, preposition, adjective, etc. It concerns with lexical ambiguity resolution. For example, the sentence " **He will table the motion.**" is tagged as follows :

[2] Also called grammatical class or part-of-speech

He / PPS will / MD table / VB the / AT motion / NN . /.

The descriptive symbols or notations, **PPS**, **MD**, **VB**, **AT**, **NN**, **BEZ**, and **JJ** are called POS tags. Each symbol or tag indicate that the word belongs to a particular grammatical class. For example, **PPS** = subject pronoun; **MD** = modal; **VB** = verb (no inflection); **AT** = article; **NN** = noun; **BEZ** = present 3rd sg form of " to be "; **JJ** = adjective.

Many words in languages are ambiguous : they may be assigned more than one POS tag [114]. For example, the English word *round* may be a noun, an adjective, a preposition or an adverb, or a verb. It is well-known that part-of-speech depends on context. The word "**table** " in the above context is tagged as a verb while it can be a noun in other context (e.g., " **The table is ready** ") [44].

Resolving these lexical ambiguities constitutes the main challenge and the ultimate goal of POS tagging system[3]. Lexical information includes not only the part-of-speech of the word but also the inflectional features of the word, such as, tense, person, number, mood, case and gender. In general, this information is extremely necessary to be available to the tagging system. It is encoded in a descriptive symbol called *a tag* and typically stored in a lexicon or a dictionary [73].

POS tagging is a very important intermediate step toward building many NLP applications, such as, text-to-speech synthesis, speech recognition, information retrieval (IR), spelling correction, and parsing system. In addition, the most prominent and largely developed field where the POS tagging used is a corpus linguistics [73, 114]. NLP applications which need POS tagging system as important intermediate step and corpus linguistics are discussed in more detail in section 2.2 and section 2.3 respectively.

[3] Also called tagger system

POS tagging can be done manually by linguists or automatically by computer. Since the size of text corpus is increasing, it is becoming very difficult for the human tagger to annotate the text in the corpus accurately. Furthermore, it requires great effort, cost, and time. So, the development of an automatic POS tagger is highly desirable.

The main task of concern for this thesis is POS tagging over the Arabic language. The current literature in the field of Arabic NLP shows that little research has been done in POS tagging for Arabic. Very few attempts were made to develop the POS tagger for Arabic such as the work done by Abuleil [15] in 1999. The aim of his tagger is to use it as a first step in parsing Arabic newspaper text. Also El-Kareh and Al-Ansary [54] presented the semi-automatic POS tagger in 2000. The first tagger for Arabic appeared in 2003 by Khoja [87] since the aim of this tagger was to produce a tagged corpus. A few taggers later appeared, such as, the work done by Habash and Rambow [71], Diab et al. [51] and Marsi et al. [102] in 2005. Also, Alshamsi and Guessom [127] and Harmin [75] presented a tagger system for Arabic in 2006. This brief literature shows that the work in POS tagging for Arabic has been done in recent years, while it was done for English, as an example, three decades ago.

Many reasons lie behind the lack of research on the Arabic language. A richly inflected and a complex morphological system that Arabic exhibits on one hand, and the lack of resources such as the availability of large manually tagged Arabic corpus on the other hand may constitutes the main reason behind the lack of research on the Arabic language. In addition, the actual deployment of the use of computers and Internet in the Arab world began in the mid-nineties and grows continuously.

The current taggers were built to tag unvocalised Arabic text using a lexicon or dictionary that was tagged manually and used as a training corpus containing all possible tags (lexical information) for each word. At this point, the main task of the tagger is to resolve the lexical ambiguity and to determine the proper tag of ambiguous words based on the context of the sentence.

The training corpus should be very huge for two reasons. The first reason is to achieve very good accuracy like the taggers accuracy (98%-99%) used for English because a very large amount of data was used to train them (e.g., hundreds of million words) while the accuarcy of Khoja tagger as an example is 86% since her tagger was trained on a very small training corpus (10,000 word) [87]. At the same time, Khoja state that "***Of course, having a tagger that did not require a tagged corpus was valuable to languages other than English, where there was no tagged corpus available***"([88], p.29).

The second reason is to avoid the most important problem in POS tagging: unknown words. Unknown words are words not appearing in the training corpus. Neither the testing corpus nor the training corpus has lexical information and tags for these words.

In case the tagger system deals with unvocalised Arabic text, a huge lexicon or training corpus is required to be available to the tagging system. Unlike English, Arabic still lacks a huge manually tagged corpus from which large amounts of training data can be extracted. At the same time, it is desirable in the authors' opinion to construct a POS tagger that needs as little training data as possible. Therefore, developing a POS tagger for unvocalised Arabic text using a statistical approach as one of the two major approaches (rule-based and statistical) that achieves reasonable accuracy seems very difficult at the present time.

1.2 Motivation

As mentioned earlier in section 1.1 the taggers were built for Arabic are based on a lexicon or dictionary that was tagged manually for training and used to tag unvocalised Arabic text. However, the Arabic language has a valuable and an important feature, called *diacritics*, which are marks placed over and below the characters of the word.

An Arabic text may be written with diacritics or without. The text that appears without diacritics is called unvocalised text. While written Arabic text with full representation of diacritics marks is called fully-vocalised text. An Arabic text is a partially-vocalised text when the diacritical mark assigned to one or maximum two letters in the word.

In addition, Arabic language has many signs that indicate the class of the word. Patterns, grammatical rules, affixes[4], and ending case, are examples of these signs. Based on these distinctive characteristics of the Arabic language, a set of questions deserve answer regarding the field of Arabic NLP. These questions and the objectives of this research are described in more detail in the following section.

1.3 Research Hypothesis, Aims and Objectives

This research begins with the following four questions :

1. **Does an automatic POS tagger system deals with partially-vocalised Arabic text exist ?**

2. **Do diacritics play an important role to resolve the lexical ambiguity that may arise in Arabic text ?**

[4] affixes in Arabic are those letters which precede the root of the word (prefixes), follow the root (suffixes) or placed inside the root with which it is associated (infixes).

3. **Does a standarised and comprehensive Arabic tag set exist ?**

4. **Does an Arabic corpus which contains partially-vocalized Arabic text exist ?**

The literature carried out on the Arabic NLP shows that the answers to the previous questions have not yet been. As mentioned above, the current taggers were built to tag unvocalised Arabic text. A tagger system that deals with partially-vocalised Arabic text does not yet exist. In addition, the importance of utilising the diacritic feature of Arabic language to reduce the ambiguity in POS tagging has not been addressed. A raw or a hand tagging corpus which contains partially-vocalised Arabic text also does not exist. Finally, despite the current taggers were used a set of tag sets as described in chapter 2, most of these tag sets were compiled to represent the general tag of the word (the general part-of-speech) without including more linguistic attributes of the Arabic word. In addition, these tag sets were not cover the most grammatical classes of Arabic language and the inflectional feature of Arabic word as well. Therefore, a standarised and a comprehensive Arabic tag set does not exist.

The aim of this research is multifaceted :

- to create a POS tagger system deals with partially-vocalised Arabic text without using a lexicon of Arabic words (tagged or untagged) especially for words belong to verb or noun classes, and at the same time achieves very good accuracy.

- to investigate the role of diacritic feature, especially at the end of the word (ending case) in reducing the ambiguity and providing semantic information that helps to determine the correct tag of each word in the testing corpus.

- to explore the possibility of using a novel technique to assign the correct tag to each word in testing corpus based on the pattern of the word instead of the word itself.

- to present a comprehensive theoretical study of the diacritic and inflectional features over the Arabic language.

1.4 Significant Research Contributions

This research provides a new contributions to the field of the Arabic NLP in different ways, these contributions can be summarised as follows.

- **AMT: Arabic Morphosyntactic Tagger**

  The ultimate contribution of this research is to develop the POS tagger system called AMT (short for Arabic Morphosyntactic Tagger). AMT deals with partially-vocalised Arabic text . The main aim of AMT is to annotate the testing corpus, that is, adding POS tag or label to each word in the testing corpus and produce a POS tagged partially-vocalised Arabic text. It is also used as a prerequisite tool for many NLP tasks, such as, parsing and informational retrieval systems. Chapter5 in this research show the design and implementation of AMT.

- **A new Tag set for Arabic**

  The fundamental component of any tagger system is the POS tag set that is used in the tagging process [98]. The development of a tag set is an extremely necessary step in building the tagging system. The need for a tag set comes from the fact that there is no standardised and comprehensive Arabic tag set that covers the grammatical classes of Arabic language. Chapter 4 describe the steps of designing a new Arabic tag set. The developed tag set follows the Arabic grammatical system, based upon POS classes and inflectional morphology that Arab grammarians describe. During the course of developing this tag set, two Arabic linguists were consulted : Prof. Ali Alhamad[5] [20]-Yarmouk University-Jordan and Mr. Walid Alqrini[6]- Ministry of Education - Jordan. The consultation was extended to cover other related issues such as, the rules of the Arabic language and the testing corpus.

- **Partially-Vocalised Arabic Corpus**

  A raw corpus which contains partially-vocalised Arabic text is needed to test the AMT tagger system. Such this corpus does not exist. This research provides a new partially-vocalised Arabic corpus. The corpus is not limited to a particular domain; it covers a wide range of topics such as scientific and literary topics. The detail of the corpus can be seen in chapter 6.

- **Pattern-Based Technique - A novel technique**

  This thesis represents a substantial starting point for developing a rule-based part-of-speech tagging system. The research present two different techniques: *Pattern-Based Technique* and *Lexical and Contextual technique*.

  The basic idea of *Pattern-Based Technique* is to generate automatically a lexicon of patterns instead of using manually tagged. The rules in this technique are based on the pattern of the word in testing corpus instead of the word itself. In addition, this research introduce a novel algorithm; Pattern-Match Algorithm(PMA). The aim of this algorithm is to match the inflected word in the testing corpus with its correct pattern in pattern lexicon.

  The *Lexical and Contextual Technique* is used to assist the *Pattern-Based Technique* to assign the correct tag to the words not tagged by *Pattern-Based Tech-*

[5] Prof. Ali Alhamad site can be found at : http://www.yu.edu.jo/ArtsArabicDeptStaff/tabid/56/Default.aspx
[6] Walid Alqrini email : walidalqrini123@yahoo.com

nique. The rules in *Lexical and Contextual Technique* are based on the character(s), affixes, the last diacritical mark, the word itself, and the surrounding words or on the tags of the surrounding words. However, chapter 5 describe these techniques in more detail.

1.5 Outline of the thesis

The rest of the thesis is organised as follows :

Chapter 2 : Related Concepts and Literature Review

The necessary background material of this research is presented in chapter 2. It is divided into five sections. Section 2.1 introduces the problem of part-of-speech tagging while some of its applications are introduced in Section 2.2. Section 2.3 discusses corpus-based linguistics. The most important approaches used to solve the problem of POS tagging are briefly examined in Section 2.4. The last section (section 2.5) defines the POS tag set and also describes the previous work on POS tag sets.

Chapter 3 : Arabic Language and POS tagging

Section 3.1 introduces an overview of the Arabic language. A brief history of the Arabic script and the diacritic feature is presented in Section 3.2. The importance of the diacritic feature in POS tagging for Arabic is discussed in Section 3.3. Section 3.4 briefly defines the Arabic grammatical system.

Chapter 4 : Tag set Design

Chapter 4 is concerned with the development of the tag set design presented in this work and contains three main sections. Section 4.1 describes the criteria to take into account while developing the POS tag set. Arabic inflectional features are explained in Section 4.2. The last section (section 4.3) introduces the developed Arabic POS tag set hierarchy and design.

Chapter 5 : Design and Implementation of AMT

This chapter is concerned with an implementation of the AMT system presented in this work. It contains five main sections. The characteristics of the AMT tagger system are defined in Section 5.1. The rule-based approach is described in Section 5.2. Section 5.3 explains the pattern-based technique used in this work while the lexical and contextual technique is explained in Section 5.4. A description of the tagger system and the tagging process is described in section 5.5.

Chapter 6 : Evaluation of Results obtained from AMT

Chapter 6 is devoted to the evaluation of results obtained from AMT. It contains three main sections. Testing data is described in Section 6.1 while the details of each experiment is done to evaluate the AMT tagger presented in Section 6.2. Finally, experimental results analysis is introduced in Section 6.3.

Chapter 7 : Conclusion

This chapter contains the main conclusion yielded by this work and future research.

Chapter 2

Related Concepts and Literature Review

Objectives

- To present the problem of part-of-speech tagging.
- To discuss some of its applications.
- To define the corpus linguistic in NLP.
- To define Part-of-Speech tag set.
- To describe the previous work on POS tag sets.
- To justify the need for a new Arabic tag set.
- To briefly examine the different approaches used to solve the problem.

2.1 Part-of-Speech (POS) Tagging Problem

To illustrate what part-of-speech tagging[1] is about, let us begin with a simple example representing an English text ([73], p.4) :

the Fulton County Grand Jury said Friday an investigation of Atlanta's recent primary election produced no evidence that any irregularities took place.

The goal of part-of-speech tagging consists of labeling or tagging each word in the text, including punctuation marks, with its correct part-of-speech. The following results are expected as the output of the tagging process.

the/**AT** Fulton/**NP** County/**NP** Grand/**NP** Jury/**NP** said/**VBD** Friday/**NR** an/**AT** investigation/**NN** of/**IN** Atlanta's/**NP$** recent/**JJ** primary/**NN** election/**NN** produced/**VBD** no/**AT** evidence/**NN** that/**CS** any/**DTI** irregularities/**NNS** took/**VBD** place/**NN**. / .

For the simple text shown, the words in the sentence are followed by a tag, where the slash "/" separates the word from the tag or part-of-speech symbol. The tag here is taken from a predefined inventory of labels called a tag set. The tag **AT** indicates that the word belongs to the grammatical class of articles; **NP** represents proper nouns; **VBD** for verbs; **IN** for prepositions; and so on.

One of the most difficult problems which affects the POS tagging is text ambiguity. Something is ambiguous when it can be understood in two or more possible senses or ways [43]. Ambiguity is the most significant problem in processing natural language. In NLP a word, a phrase, or a sentence is called ambiguous if it can be reasonably interpreted in more than one way [33]. Unlike grammars for computer programming languages, grammars for natural languages like English, as an example, are usually

[1] Also called morphosyntactic categorisation or syntactic wordclass tagging. (see ref [73])

ambiguous. Figure 2.1 shows the main ambiguity types in a natural language.

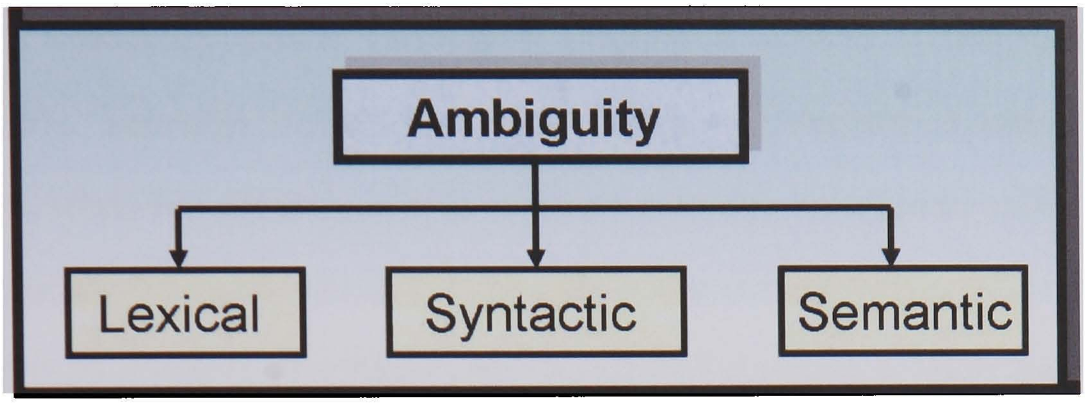


Figure 2.1: Ambiguity types

The list below describe the ambiguity types in more detail.

- **Lexical Ambiguity**

  Lexical ambiguity occurs when a word has several meanings. For instance, the word "*Lie* " = "Statement that you know it is not true" or "present tense of lay". Words like "*light*", "*note*", "*bear*" and "*over*" are lexically ambiguous [40].

- **Syntactic (structural) Ambiguity**

  Syntactic (structural) ambiguity occurs when a given sequence of words can be given more than one grammatical structure, and each has a different meaning. In other word, when there are different possible syntactic parses for a grammatical sentence. For example, the sentence "*Visiting relatives can be so boring*" is structurally ambiguous (Who is doing the visiting?). Another example of such ambiguity is the problem of attachment of modifiers to the proper constituents. Consider the sentence "*Fasten the assembly with the lever*". This may be either an instruction to fasten the assembly using a lever, or an instruction to fasten the assembly, which has a lever attached to it. With the former interpretation, the prepositional phrase "*with the lever*" is attached to the verb, and with the latter, it is attached to the noun phrase object [33].

- **Semantic Ambiguity**

  Semantic ambiguity occurs when a sentence has more than one way of reading it within its context although it contains no lexical or structural ambiguity [40]. Semantic ambiguity refers to the broad category of ambiguity which arises when the meaning of the sentence must be determined with the help of greater knowledge sources. The problem of resolving simple pronominal reference is an example of semantic ambiguity. In the sentence *"Start the engine and keep it running"*, the fact that *it* refers to the engine is not inferable from the single clause *"keep it running"*. Knowledge of the prior clause is necessary to resolve the pronoun [33].

POS tagging is the most common type of lexical disambiguation. POS Tagger system is typically used to resolve the lexical ambiguity (ambiguity in a single word) based on context using the surrounding words and grammar rules. For example, in the following English sentence[2]: **"Book that flight"** the word **"Book "** as shown in figure 2.2 is ambiguous regarding its part-of-speech, it can be a verb `[V]` or a noun `[N]`. Similarly, the word **"that"** can be a determiner `[DET]`, or a complementiser `[C]`.

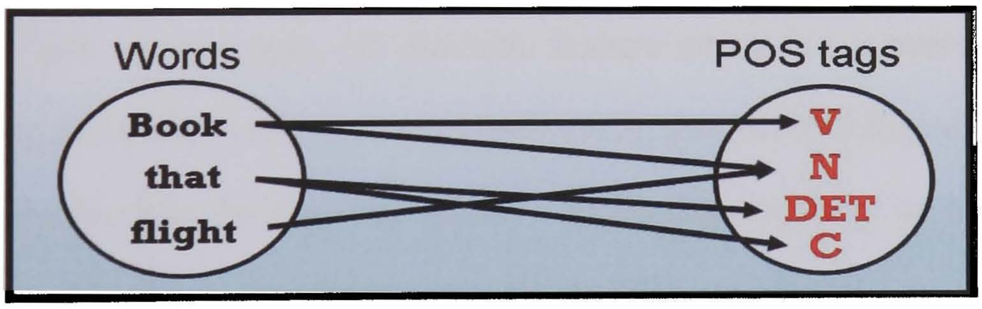


Figure 2.2: The possible values of tag

[2] from: www.cse.ttu.edu.tw/chingyeh/courses/nlp/slides/ch8WordClassesAndPOSTagging.ppt

In Arabic, the same problem is faced. For example, in the Arabic[3] sentence[4] shown in Table 2.1, the word دخل, *dkhl* is ambiguous with regard to its part-of-speech. It can be a verb if it means “**entered** ”, or a noun “**income** ”.

Arabic Sentence :	دخل رمزي البيت
Transliteration :	*Albyt rmzy dkhl*
Translation :	“ **the house** ” “ **Ramzy** ” “ **entered** ”, **but it really means**
	“ Ramzy entered the house ”

Table 2.1: Ambiguous words in Arabic sentence

Since many words in languages are POS ambiguous, the lexical ambiguities become the main problem that POS tagging system faces. Resolving these ambiguities constitutes the main challenge in POS tagging. The tagger system should choose the best tag for each word in the text which has more than one part-of-speech. It is clear and well known that part-of-speech depend on context [45]. The word “**table** ” as another example, can be a verb in some contexts (e.g., “He will **table** the order”) and a noun in others (e.g., “The **table** is too big”). Therefore, adequate context and/or adequate semantic information knowledge are required to resolve the problem of POS tagging [109].

The Arabic language differs from English in terms of characteristics and grammatical system as well, for example, (1) *diacritic* feature which is not present in English, and (2) *the root and pattern structure* on which the Arabic morphological system based is on. While Arabic has the diacritic feature, the Arabic text may be written without diacritics (unvocalised) or with them (partially or fully-vocalised).

When the text is written in an unvocalised form, resolving the lexical ambiguities in

[3] Since a cursive system from right to left is used in written Arabic, the sentence is read from right to left. More details of the Arabic language are discussed in chapter 3

[4] Transliterated Arabic words throughout this thesis are in italics while English translations are in double quotes. All separated by commas.

this case resembles English language which is based on the context. But the case is different if the text is partially or fully-vocalised.

The testing corpus in this work is a partially-vocalized Arabic text. The diacritical mark is assigned only to the last letter of each word in the testing corpus. There are two reasons for choosing a partially-vocalised Arabic text as a testing corpus. The first one is to investigate the importance of the last diacritical mark in reducing the lexical ambiguity of the word and helping the POS tagger to resolve this ambiguity and to assign the correct tag to the words in the testing corpus regardless of the context in most cases. The second reason is to explore the possibility of applying pattern-based rules to tag the testing words based on the pattern of the word instead of the word itself.

The importance of the diacritic feature in POS tagging and pattern-based approach is described in more detail in Chapter 3 and Chapter 5 respectively. On the other hand, ambiguity identification is crucial not only for the part-of-speech tagging, but also for any other text processing dealing with content, such as, speech processing or semantic annotation [33].

2.2 Applications of Part-of-Speech Tagging

POS tagging is a preliminary stage for many NLP applications. The most prominent and largely developed field where the POS tagging is used is corpus linguistics [73]. Corpus linguistics is described in more detail in Section 2.3. It is also a useful and an important practical problem with potential NLP applications in many areas [45], such as :

- **IR : Information Retrieval**

  POS tagging system can enhance an IR application by selecting nouns or other important words from a document (e.g sequences of proper nouns or common nouns) [50]. The user of World Wide Web will appreciate the importance of accurate information retrieval. POS tagging removing the lexical ambiguity and identifying the syntactic class of words. For example, the word " Cooking " can be used either as a noun (" Cooking is fun ") or a verb (" he is cooking lamb "). By identifying the syntactic role of the word " Cooking " within documents, the results of searching for " Cooking fish ", as an example, would not include documents where the word is used as a noun [43].

- **Parsing system**

  POS tagging system can be an important first step and an integral part for any parsing system [50]. Since the parser needs lexical information for each word before performing the parsing process, such this information usually obtained from the output of a POS tagger.

- **Word Processing**

  Since most word processors attempt to provide a check not only on spelling, but also grammatically, knowing the category of misspelled word helps in reducing the number of corrections [73].

- **Speech synthesis system**

  Knowing POS can produce more natural pronunciations in the speech synthesis system and more accuracy in the speech recognition system [50].

- **Machine Translation**

  A tagged version of each corpus on parallel corpora[5] (text in different languages)

[5] corpora is a Latin plural of corpus. Next section defines corpus linguistic in more detail.

in machine translation research facilitates the automatic identification of translation equivalents on words and phrase level [73].

- **Building Dictionaries**

  A tagged text has a great benefit also in building dictionaries. It has information which can be of help to users of the dictionary such as language learners and teachers in acquiring or identifying a core vocabulary [73].

2.3 Corpus-based Linguistics

2.3.1 Introduction

Over the last decade, many efforts have been devoted to compile a large raw text corpora. Corpus Linguistics is the study of linguistic phenomena through large collections of machine-readable texts [9] : corpora. A corpus is defined by Leech [92, 140] as " ***a large collection of natural language material stored in machine readable form that can be easily accessed, automatically searched, manipulated, copied and transferred*** ".

The usability of corpus can be extremely enhanced by adding POS class to every word in the corpus or any other relevant linguistic information which may be needed by the linguist or other developers in NLP. Once the corpus is analysed, it constitutes a kind of database that contains information about the linguistic structure and statistics of language usage [140].

Since the fast development of computers with huge memory capabilities and software on the one hand and the availability of large documents, books and publications on a machine readable format, all these factors have made the compilation of these corpora no longer a difficult problem.

Corpus linguistics can be considered as an independent field; it is a methodology rather than an aspect of a specific language [86]. Many POS tagging systems built earlier used different approaches especially for English language since it is the first language of corpus linguistics [30]. The majority of these systems are designed to annotate the text corpora, that is, they contain not only word, but also linguistic information on them, such as, part-of-speech. A tagged corpus has a higher linguistic value, it provides specific linguistic information which is very useful for developing lexical resources, inducing grammatical structure and estimating parameters of statistical model [101].

2.3.2 Existing Corpora : English and other languages

The history of corpus linguistics started at the beginning of the sixties when the first printed American English corpus was compiled, which is known as Brown corpus. Summarised below is a list, although not exhaustive, of some well-known computerised English corpora:

1. **Brown Corpus**

   The Brown Corpus [60, 68] was Compiled by W. Nelson Francis and Henry Kucera in Brown University and contains 500 samples, each about 2,000 words of continuous written American English, from texts published in the US in 1961. The original edition of the text corpus was completed in 1964. It was revised twice in 1971, and then revised and annotated with word tags[6] in 1979.

2. **LOB: Lancaster-Oslo-Bergen corpus**

   The LOB Corpus [28, 82] contains also approximately one million words of

[6] POS tag sets are described in chapter 4

British English from publications of the year 1961. It is a British counterpart of the Brown corpus resulting from research collaboration between the University of Lancaster, the University of Oslo, and the Norwegian Computing Centre for the Humanities. The text corpus was published in 1978, and its tagged edition in 1986.

3. **LLC: London-Lund Corpus**

   The London-Lund Corpus [131] was compiled at Lund University. It contains about 500,000 words of spoken British English collected from broadcast and recorded materials. The texts were collected between 1959 and 1975.

4. **Penn Treebank Corpus**

   The Penn Treebank corpus [99, 124] was developed at the University of Pennsylvania. The Penn Treebank-I project ended in 1992 with 4.5 million words of text, including the entire Brown corpus text, the Wall Street Journal Corpus, and some other genres. The texts were tagged with POS tags. The data produced by the Treebank is released through the Linguistic Data Consortium (LDC).

5. **ICE: International Corpus of English**

   The International Corpus of English [67] was compiled by research teams (15 researchers) from different English speaking countries, such as USA, UK, Australia, and New Zealand. It contains about one million words with regional varieties of English for each component. For example, the ICE-GB[7] consists of one million words completed in 1998. The texts in the corpus were published or recorded between 1990-1996.

6. **BNC: British National Corpus**

   The British National Corpus [65] began in 1991 and published in 1994. It con-

[7] For more information: http://www.ucl.ac.uk/english-usage/projects/ice-gb/index.htm

tains Over 100 million words of written and spoken modern British English (90% written, 10% spoken). The corpus is encoded with SGML to represent POS tags and automatically tagged.

7. **SUSANNE Corpus**

   The SUSANNE Corpus [121] was created by Geoffrey Sampson with the sponsorship of the Economic and Social Research Council (UK). It contains about 130,000 words of American English based on a subset of the million-word Brown Corpus. It is a modification of the Gothenburg Corpus and is freely available without formalities for use by researchers.

8. **TOSCA Corpus**

   TOSCA Corpus [12] has been compiled at the University of Nijmegen in 1986. It contains about 1.5 million words of British English and consists of written texts on education, history, philosophy, etc.

In addition, many other computerised English corpora have been developed, such as : SEC: Spoken English Corpus [133], PoW: Polytechnic of wales corpus [128], SCRIBE: Spoken Corpus Recordings In British English [29], COLT: Corpus of London Teenager English [25] and IPSM: Industrial Parsing of Software Manuals [130].

Lastly in this list, the multi-tagged corpus, which is known as AMALGAM corpus [30] (short for Automatic Mapping Among Lexico-Grammatical Annotation Models). This corpus has been developed in Leeds University within the AMALGAM[8] project by Atwell et al. [31]. It contains texts from different genres of English corpora such as, COLT, SEC and IPSM.

[8] http://www.comp.leeds.ac.uk/amalgam/amalgam/amalgover.html

It becomes clear that English has been the productive field of research in corpus linguistics and it stands out as the most computerised language in the world due to hundreds, if not thousands of different corpora which have been developed and are being developed.

The success of the English language in the field of natural language processing and corpus linguistic encouraged other researchers to build their own corpora, such as : Chinese (The UCLA Chinese Corpus), Czech (Czech National Corpus), Danish (Danish Corpus), Spanish (LEXESP corpus,), German (NEGRA corpus), French (TLF corpus), Swedish (Bank of Swedish corpus), Catalan (CTILC corpus), Basque (EEBS corpus), Basnian (Oslo corpus of Bosnian Texts), and many other languages [10].

2.3.3 Arabic language corpora

Unlike English, Arabic has been much less fortunate in the field of research in corpus linguistics as well as POS tagging for Arabic. A useful survey on existing resources for Arabic corpora can be found in work done by Latfia Alsulaiti and Eric Atwell [18]. However, a number of electronic unvocalised Arabic text raw corpora have been compiled, such as:

1. **An-Nahar Newspaper Text Corpus**

   An-Nahar Corpus [2] comprises articles in written Arabic collected from the articles published between 1995 to 2000. The total size of the complete files in this corpus is 806 MB.

2. **Al-Hayat Corpus**

   Al-Hayat Corpus [1] has been compiled at the University of Essex, in collaboration with the Open University. It contains 18,639,264 distinct tokens in 42,591 articles covering several subjects, such as, General, Car, Computer, News, Economics, Science, and Sport. The size of the total file is 268 MB.

3. **Buckwalter Arabic Corpus**

   The Buckwalter Corpus [4] was compiled by Tim Buckwalter between 1986-2003. It contains around three million written Arabic words collected from public resources on the Web.

4. **Nijmegen Corpus**

   Nijmegen Corpus [6] was compiled at Nijmegen University in 1996. It contains Over 2M words of written Arabic words collected to build an Arabic-Dutch/Dutch-Arabic dictionary.

5. **Arabic Newswire corpus**

   The Arabic Newswire corpus [3] was compiled by David Graff and Kevin Walker at University of Pennsylvania (Linguistic Data Consortium (LDC)) in 2004. It contains 76 million tokens (869 MB) covering written Arabic texts collected from Agence France Presse, Xinhua News Agency, and Umma Press from 1994 to 2000. The source material in this corpus was tagged using TIPSTER-style SGML and was transcoded to Unicode (UTF-8).

6. **CCA : Corpus of Contemporary Arabic**

   The Corpus of Contemporary Arabic [18] was compiled by Latifa Alsulaiti during her MSc research project with Eric Atwell at University of Leeds in 2004. It contains around 1M words covers written and spoken Arabic text collected from websites and online magazines. It is the only corpus available free for public.

7. **Penn Arabic Treebank corpus**

   The Penn Arabic Treebank [7] project started in 2001 at the University of Pennsylvania to develop an Arabic corpus containing one million words. The project began with 734 files representing 166K words of written Modern Standard Arabic newswire from the Agence France Presse corpus, which was released as Arabic Treebank: Part 1. The second part was released as the 168K word corpus, Arabic Treebank: Part 2. The Arabic Treebank: Part 3 corpus was released in 2005 and it consists of 600 stories from the An-Nahar corpus.

In addition, there are other Arabic corpora have been compiled [8], such as, CLARA, Egypt. DINAR, Leuven, and other corpora. Unfortunately, these corpora are not available to researchers free of charge except CCA corpus. However, some of these Arabic corpora can be acquired from the Linguistic Data Consortium (LDC) and the European Language Resources Association (ELRA).

2.4 Part-of-Speech tag set

2.4.1 Introduction

The POS tag set is a list of all the word classes that will be used in the tagging process. It is the fundamental component of any tagger system and the first step for the annotation of corpora [89]. A tag is a code or descriptive symbol that represents some features or set of features attached to the word in a text [73, 105]. Thus, a POS tag set is an inventory of labels used to classify and mark up words of a target text [74].

A new Arabic tag set called (ARBTAGS) has been developed. The justification behind developing a new Arabic tag set is explained in section 2.4.4 while the previous work in POS tag sets for English and other languages is described in section 2.4.2. Previous work in Arabic POS tag sets is introduced in section 2.4.3.

2.4.2 English and other languages POS tag sets

Since English corpora have been tagged by several POS tagging systems, numbers of popular tag sets have been built also to support these POS systems. The list below summarises some of these tag sets which can be found at the site of AMALGAM[9].

- **Brown tag set**

  The Brown tag set started with a set of 77 tags, and enlarged to about 226 tags used to tag and enhance the coverage of Brown corpus. Sample of Brown tag set can be seen in Table 2.2

Tag	Description	Example(s)
ABN	*determiner/pronoun, pre-quantifier*	all, half, many, nary
ABX	*determiner/pronoun, double conjunction or pre-quantifier*	both
BED	*verb "to be", past tense, 2nd person singular or all persons plural*	were
CD	*numeral, cardinal*	two, one, 1
CS	*conjunction, subordinating*	that, as, after, whether
DOD	*verb "to do", past tense*	did, done
IN	*preposition*	of, in, for, by, at
MD	*modal auxiliary*	should, may, might, will
HVN	*verb "to have", present participle*	had
JJ	*adjective*	failure, burden, court
NN	*noun, singular, common*	did, done
JJS	*adjective, semantically superlative*	top, chief, principal
NPS	*noun, plural, proper*	hases, Aderholds, Chapelles

Table 2.2: Sample of Brown tag set

- **LOB tag set**

  The LOB tag set was based on the Brown corpus tag set, but revised for fine

[9] AMALGAM project : http://www.comp.leeds.ac.uk/amalgam/amalgam/amalghome.htm

granularity. The tag set contains 135 tags used to tag LOB corpus. Table 2.3 shows a sample of LOB tag set.

Tag	Description	Tag	Description
CS	*subordinating conjunction*	**NN**	*singular common noun*
CD	*Cardinal number*	**NNP**	*singular common noun with word initial capital*
NP	*singular proper noun*	**JNP**	*adjective with word initial capital*
MD	*modal verb*	**OD**	*ordinal number*
NPS	*plural proper noun*	**PPL**	*singular reflexive personal pronoun*
NR	*singular adverbial noun*	**QL**	*qualifier*
VB	*base form of lexical verb*	**VBD**	*past tense of lexical verb*
VBG	*present participle of lexical verb*	**VBN**	*past participle of lexical verb*
WPA	*nominative wh-pronoun*	**ZZ**	*letter(s) of the alphabet*
TO	*infinitival TO*	**NPLS**	*plural locative noun with word initial capital*
NPL	*singular locative noun with word initial capital*	**JJ**	*general adjective*

Table 2.3: Sample of LOB tag set

- **LLC tag set**

  The LLC tag set was used to tag London Lund Corpus. It contains about 210 tags.

- **ICE tag set**

  The ICE tag set was used to tag the International Corpus of English . It contains about 205 tags.

- **SEC tag set**

  The SEC tag set was used to tag the Lancaster/IBM Spoken English Corpus. It is based on the LOB corpus tag set. In contrast, LOB tag set differentiates between relative and interrogative **WH**-pronouns whereas SEC tag set does not. For example, in SEC tag set, the tag **WP** used to cover **WH**-pronouns, interrogative, nominative or accusative and **WH**-pronouns, relative, nominative or accusative whereas LOB used separate tags [30].

- **Penn Treebank tag set**[10]**:** The Penn Treebank tag set was used to tag Penn Treebank corpus. It contains about 36 tags used. Sample of Penn Treebank tag set can be seen in Table 2.4.

Tag	Description	Tag	Description
CC	*Coordinating conjunction*	**NNS**	*noun, plural*
CD	*Cardinal number*	**NP**	*Proper noun, singular*
EX	*Existential "there"*	**NPS**	*Proper noun, plural*
FW	*Foreign word*	**PDT**	*Predeterminer*
IN	*Preposition or subordination conjunction*	**POS**	*Possessive ending*
JJ	*Adjective*	**VB**	*verb, base form*
JJR	*Adjective, comparative*	**VBD**	*verb, past tense*
JJS	*Adjective, superlative*	**VBN**	*verb, past particle*
LS	*List item marker*	**VBG**	*verb, present participle*
MD	*Modal*	**VBP**	*verb, non 3rd person singular present*
NN	*noun, singular or mass*	**VBZ**	*verb, 3rd person singular present*

Table 2.4: Sample of Penn Treebank tag set

In addition, several English tag sets have been built and used to tag other corpora, such as : URCEL C7 tag set, SUSANNE corpus tag set, TOSCA corpus tag set and PoW corpus tag set [50, 74].

Other tag sets have been designed for languages other than English, such as : Urdu [74], French [41], African languages [80], Czech [79, 83], Hungarian [136], Slovene [53], German [94], Persian [104], Swedish [115], Hebrew [118], Italian [34], Span-

[10] http://www.comp.leeds.ac.uk/amalgam/tagsets/upenn.html

ish [27] and Turkisk [112]. However, a useful resources on comparison of some of the above tag sets can be found on [30,46].

2.4.3 Arabic POS tag sets

Since there has not been much work done in POS tagging for Arabic, a very small number of tag sets had been built. The list below summarises well-known tag sets that have been built for Arabic.

- **Khoja tag set**

  Khoja [89] describes an Arabic tag set that has been built based on POS classes and inflectional morphology system and used for her tagger system APT : An Automatic Arabic Part-of-Speech Tagger. The tag set contains 177 detailed tags. Each tag represents the name of the three main class (verb, noun, particle) and their sub-classes including the inflectional features such as , gender, number and person. For example, her tag set covers 57 type of verbs, 103 type of nouns, 9 type of Particles, 7 residual and 1 punctuation. Sample of Khoja tag set can be seen in Table 2.5.

- **El-Kareh and Al-Ansary tag set**

  El-Kareh and Al-Ansary [54,87] described an Arabic tag set used for their semi-automatic tagger system. Their tag set contains 72 tags covering 3 sub-classes of the main class verb, 46 sub-classes of the main class noun and 23 sub-classes of the main class particle.

- **Linguistic Data Consortium (LDC) tag set[11]:**

  The LDC tag set was created by the Linguistic Data Consortium (LDC) team and contains 24 tags used to tag Penn Arabic Treebank corpus. It is also used by

[11] For more : http://www.ling.ohio-state.edu/b̃romberg/postags/posproject.html

Tag	Description	Tag	Description
NCSgMNI	*Sing. Masc. Nom. Indef. common noun*	**NCSgMND**	*Sing. Masc. Nom. Def. common noun*
NCSgFND	*Sing. Fem. Nom. Def. common noun*	**NCPlMNI**	*Plu. Masc. Nom. Indef. common noun*
NP	*Proper noun*	**NPrRSDuM**	*Dual Masc. Spec. relative pronoun*
NNuCaSgM	*Sing. Masc. cardinal number*	**VPSg2M**	*2nd, Sing. Masc. perfect verb*
VPSg3M	*3rd, Sing. Masc. perfect verb*	**VPDu3M**	*3rd, Dual Masc. perfect verb*
VPPl3F	*3rd, Plu. Fem. perfect verb*	**VISg3MJ**	*3rd, Sing. Masc. Juss. imperfect verb*
VIDu3MI	*3rd, Dual Masc. Ind. imperfect verb*	**VIDu3MJ**	*3rd, Dual Masc. Juss. imperfect verb*
VIPl2MI	*2nd, Plu. Masc. Ind. imperfect verb*	**VIPl3MJ**	*3rd, Plu. Masc. Juss. imperfect verb*
VIvSg2M	*2nd, Sing. Masc. imperative verb*	**VIvPl2M**	*2nd, Plu. Masc. imperative verb*
PPr	*Prepositions*	**PE**	*Exceptions*
RF	*Residual, foreign*	**PU**	*Punctuation*

Table 2.5: Sample of Khoja tag set

other works in POS tagging for Arabic, such as, the SVM tagger done by Mona Diab [51] and Egyptian dialect POS tagger done by Duh and Kirchhoff [52]. The LDC tag set can be seen in Table 2.6.

- **Alshamsi and Guessom tag set**

  Alshamsi and Guessom [127] described an Arabic tag set used for their HMM POS tagger system. It contains 55 tags. As Alshamsi and Guessom point out, that since the main use of their tagger is intended to be for Named Entity extraction, their tag set is not a fine-grained tag set. For example, they used the following tags : **NOUN** (noun), **ADJ** (adjective), **PNOUN** (proper noun), **PRON** (pronoun), **INDEF** (indefinite noun) and **DEF**(definite noun) to represent the noun category and its subcategories. On the other hand, **PVERB** (perfect verb),

Tag	Description	Tag	Description
CC	*Coordinating conjunction*	**DT**	*Determiner*
CD	*Cardinal number*	CONJ+NEG	*Conjunction, Negation Particle*
FW	*Foreign word*	**NPS**	*Noun, plural*
NN	*Noun, singular or mass*	**IN**	*Preposition or subordinating conjunction*
JJ	*Adjective*	**NNP**	*Proper noun, singular*
NNPS	*Proper noun, plural*	**PRP**	*Personal pronoun*
PRP$	*Possessive pronoun*	**PUNC**	*Punctuation*
RB	*Adverb*	**RP**	*Particle*
UH	*Interjection*	**VBD**	*Verb, past tense*
VBN	*Verb, past participle*	**VBP**	*Verb, present*
WP	*Wh-pronoun*	**WRB**	*Wh-adverb*
NO_FUNC		**NUMERIC_COMMA**	

Table 2.6: The LDC POS tagset

IVERB (imperfect verb), **CVERB** (imperative verb), **MOOD_SJ** (subjunctive or jussive), **MOOD_I** (indicative), **SUFF_SUBJ** (suffix subject) and **FUTURE** (future/Imperative) tags were used to represent the verb category and its subcategories. For particle, **INTERROGATE, NEGATION, CONJ** and **PREP** tags were used to represent interrogation, negation, conjunction and preposition particles. In addition, some inflectional features, such as, person, number and gender were added to their tag names to show the morphology analysis of the word. For example, **PRON_2S** tags means second person singular number feminine/masculine gender pronoun. Table 2.7 shows a sample of their tag set.

CONJ	**DPRON_MP**	**NOUN**	**PRON_2MP**
CVERB	**DPRON_MS**	**PNOUN**	**PRON_2S**
IV3	**PRON_1S**	**PPRON_2FP**	**PPRON_3FP**
PVERB	**CVERB**	**PREP**	**PRON_3FS**
DEF	**INTERROGATE**	**PRON**	**PRON_3MP**

Table 2.7: Sample of Alshamsi and Guessoum tag set

2.4.4 Justification for a new Arabic tag set

In this work, an Arabic tag set called (ARBTAGS) has been developed. The rationale behind developing our tag set comes from the fact that there is no standardised and comprehensive Arabic tag set covering the most common types (sub-classes) of the three main Arabic word classes.

The developed tag set differs from the tag sets which have been built for Arabic. The main difference is a tag set hierarchy which is described in figure 2.3 and shows the way that the Arabic word has been classified.

As shown in the tag set hierarchy, noun class is classified into sixteen sub-classes (common, proper, Adjective, etc.), verb class into three sub-classes (perfect, imperfect, imperative), particle class into seven sub-classes (preposition, vocative, conjunction, etc.), and one punctuation. In addition, one more general tag is added to the above general tags; this tag is used to represent the foreign word (Arabised word); it's [Fw]. So, the total size of general tags becomes 28 tag.

These general tags represent the names of main classes and the sub-classes without inflectional features, the developed tag set hierarchy is differs from the tag sets hierarchy which have been built for Arabic. For example, Khoja (see figure 2.4 which is reproduced from the original figure from [89]) was classified noun class into five sub-classes (common, proper, pronoun (personal, relative, demonstrative), numeral, adjective), while particle class was categorised into nine sub-classes (prepositions, adverbial, conjunctions, interjections, exceptions, negatives, subordinates, answer, explanations).

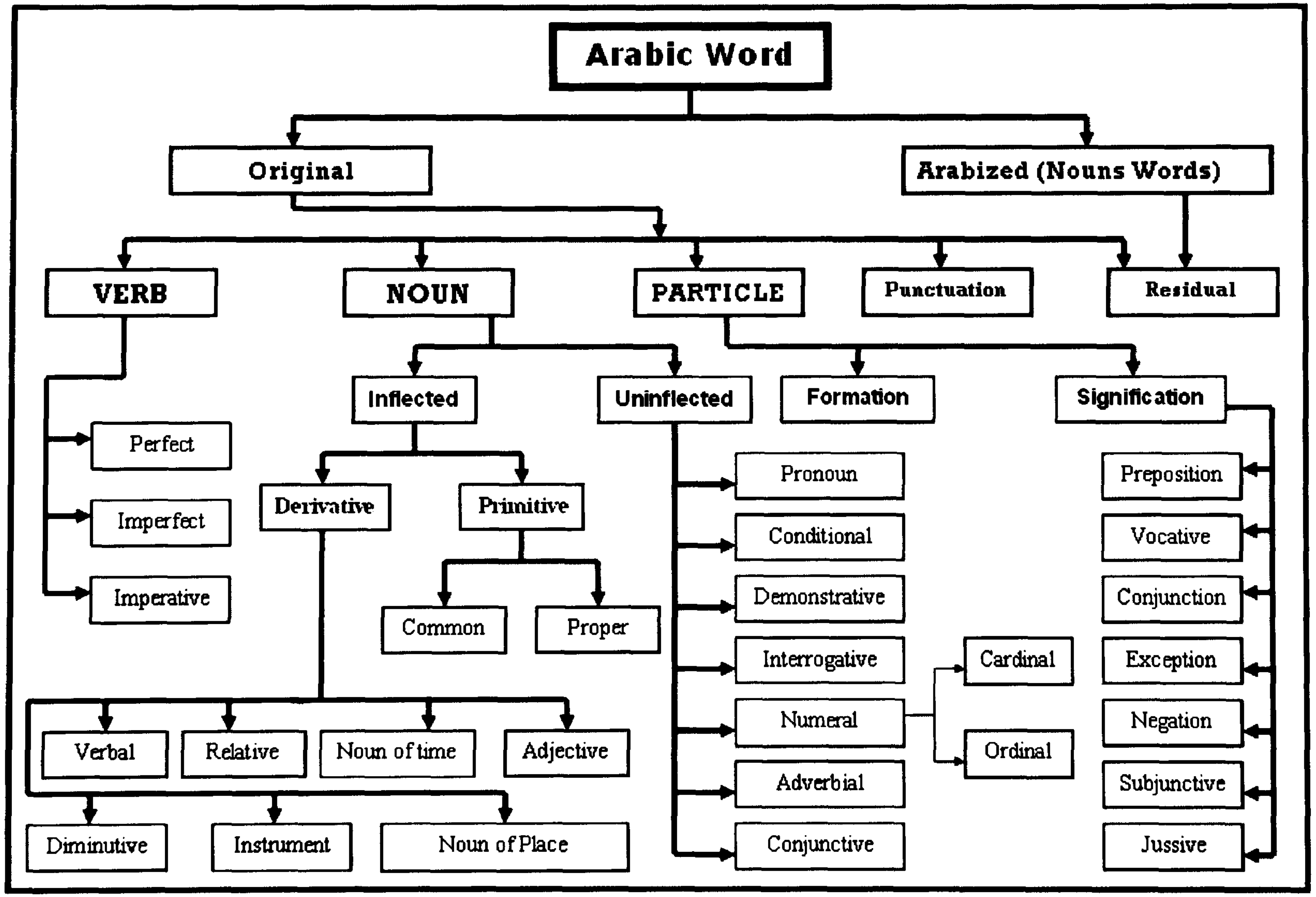


Figure 2.3: ARBTAGS tag set hierarchy

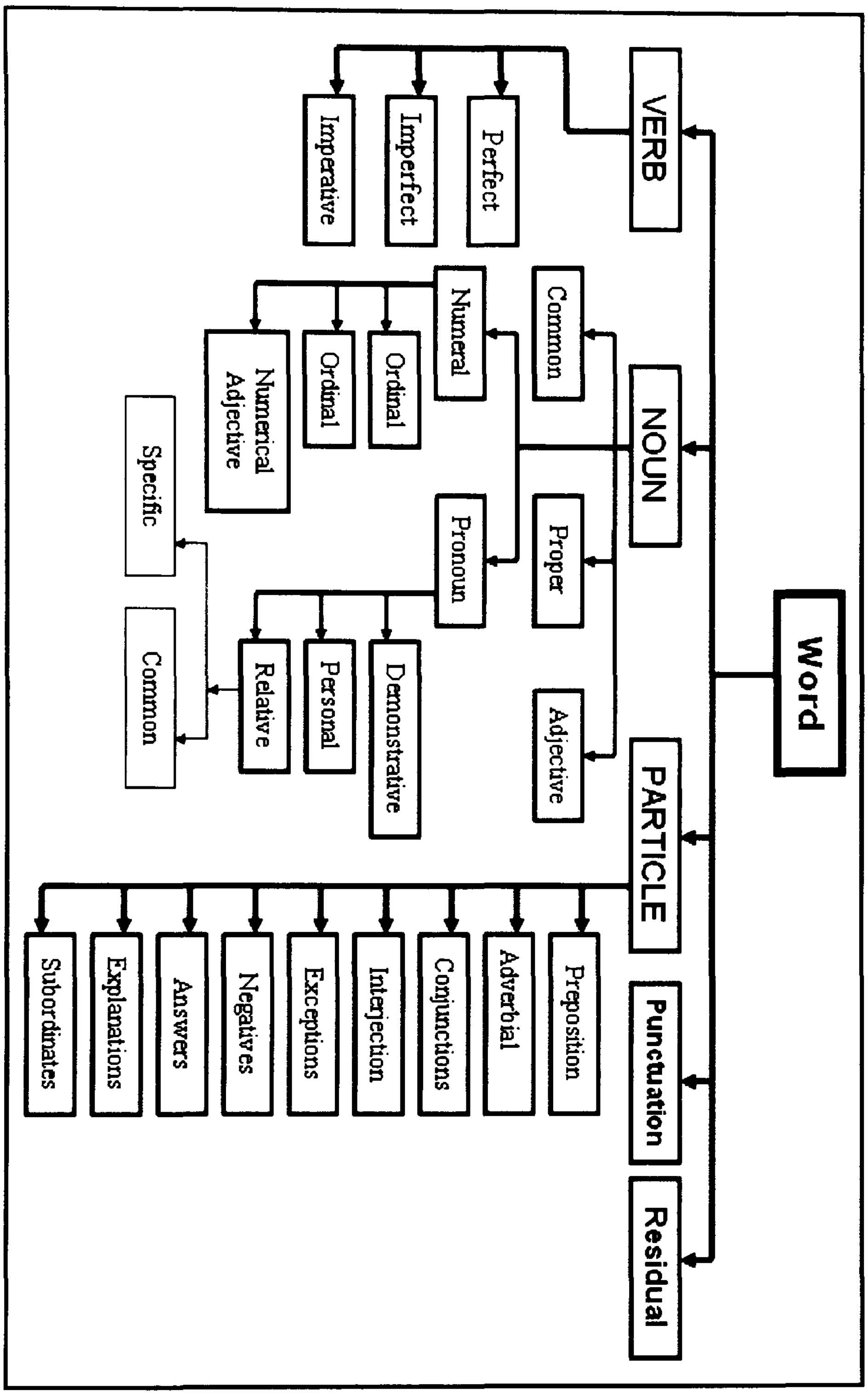


Figure 2.4: Khoja tag set hierarchy [89]

Alshamsi and Guessom [127] were classified noun class into four sub-class, particle class into four subclass (see Table 2.7). They point out, there is no need to have fine-grained a tag set, since their tagger was intended to be for Named Entity extraction([127], p.34). LDC tag set as another example was mapped from English tag set and not rich enough to cover Arabic POS classes [102].

The subclasses of the developed tag set, such as : verbal, diminutive, instrument, noun of place, noun of time, conditional and interrogative, which belong to the noun class . In addition, vocative, subjunctive and jussive subclasses which belong to the particle class. These subclasses have not been mentioned before. Thus, using one of the tag set has been built before will not capture all the subclasses shown in the developed tag set hierarchy. In addition, the testing corpus in this work is a partially-vocalised text which leads to use more inflectional features than described in the other tag set.

The developed tag set is based upon POS classes and inflectional morphology [24]. The tag names in the developed tag set uses terminology from Arabic tradition rather than English grammar. For example, in Khoja tag set, the tag [VPP12M] is verb perfect plural second-person masculine. As Atwell [30] point out, since Khoja Arabic tag set came from the Lancaster URCEL tradition of Corpus Linguistics, she was influenced by the English tag sets, such as, CLAWS heritage of tag set for LOB and BNC corpora. Therefore, she has used terminology from English grammar rather than Arabic tradition in naming categories and features. The author agrees with Atwell. It seems that, not only khoja tag set uses terminology from English grammar rather than Arabic tradition, but also the tag sets have been built for Arabic and described above used the same terminology in naming categories and features.

The tag names in the developed tag set uses terminology from Arabic tradition rather than English grammar. For example, `VePiMaPlThDc`, which means *[Imperative verb, masculine gender, plural number, third person, subjunctive mood]*. Details about the developed tag set design are provided in chapter 4.

ARBTAGS tag set developed in this work contains 161 POS detailed tags, 101 nouns, 50 verbs, 9 particles, 1 punctuation; these tags are enriched with inflectional features information. However, the general and detailed tags with examples have been described in full in Appendix A.1 and Appendix A.2.

On the other hand, the usability of ARBTAGS has been tested in manual tagging and built up a set of tagged text to serve as a goal corpus used to compare it with the results obtained from the AMT tagger. Despite that Khoja built 177 detailed tags, but she actually used five main general tags (noun, verb, particle, punctuation and residual) and a simplified version of the tagset (30 detailed tags) to make the training of POS tagger computationally feasible([88], p.71). While most of the tags used in the developed tagger are detailed tags due to the main aim of the developed tagger, that is, to provide a tagged corpus more useful for linguists and NLP developers to extract more linguistic information from it.

2.5 Part-of-Speech Tagging Approaches

The existing literature shows that there are two main approaches to POS tagging studied so far, these are : the *Rule-based Approach*[12] and the *Statistical Approach*[13]. Many

[12] also called linguistic approach or Knowledge-Based Approach

[13] also called Probabilistic Approach or Stochastic Approach

POS tagging systems have been implemented using these approaches. The majority of these systems were used to tag text corpora.

We categorise these systems based on whether the tagger systems are adopting the rule-based approach or the statistical approach. On the other hand, some systems adopt a hybrid approach (rule-based and statistical) and some other systems use other approaches, such as, neural networks, machine learning algorithms and decision trees, which have also been addressed. Here the focus is on the two main techniques. More detail is provided and some well-known systems described. In addition, we also discuss the advantages and disadvantages of each of the two main approaches. A useful and good survey on the POS tagging approaches can be found in the work done by Abney [13].

2.5.1 Rule-based Approach

The rule-based approach is based on incorporating a set of linguistic rules in the tagger [49]. This approach uses the linguist-written language model that contains rules ranging from a few hundreds to several thousands. The approach adopted here in this work is based on the rule-based approach. The tagger presented in this work has two main rule components, these are : *pattern-based rules* and *lexical and contextual rules.*

The pattern-based technique is a novel technique presented in this work. The basic idea of this technique is to generate automatically a lexicon of patterns instead of using manually tagged or untagged lexicon or training corpus. The triggers in pattern-based technique depend on the patterns of text words. A novel algorithm to match the Arabic word in the testing corpus with its correct pattern in patterns lexicon has also been built. In addition, a small amount of hand-written rules and constraint rules (*lexical and contextual rules*) have been used to assist the main technique to assign the correct tag to those words not tagged by pattern-based technique. The tagger system and the proposed approach are fully described in chapter 5.

The rule-based approach was the earliest approach for automated POS tagging. It dates back to the 1960's and 1970's when automated POS tagging was initially explored by Klein and Simmons [90] in 1963 and the work done by Greene and Rubin [63, 68] in 1971 which considered the most representative of such pioneer taggers. Afterward, a number of rule-based systems have been developed, such as, work done by Hindle [77], Brodaa [38], Paulussen and Martin [113], Karlsson [85], Voutilainen [135] and Brill [36, 37]. Some of these systems have been built to tag corpora while other systems were built for developing the parsing system.

A rule in the rule-based approach may be represented as follows [77]:

$$\textbf{[PREP + TNS]} \rightarrow \textbf{TNS [N + V]}$$

where PREP = preposition, TNS = tense, N = noun, and V = verb.

This rule implies that a word that can be a preposition or a tense marker (i.e. the word *"to"*) should be tagged with the word TO (tense marker) when it precedes a word that can be a noun or a verb.

Some of the well-known rule-based systems are now briefly discussed.

- **CGC: Computational Grammar Coder system**

  Klein and Simmons [90] developed a larger question-answering system which contains a syntactic analysis program that needs a part-of-speech tagger as a necessary component. They developed a Computational Grammar Coder (CGC) which itself a part-of-speech tagger. Their tagger uses several smaller English dictionaries with a total of 20,000 words, such as function-word dictionary. This dictionary containing articles, prepositions, pronouns, conjunctions, auxiliary verbs, adverbs, etc. It comprises 500 words all of which have unique grammar codes (tags). Their CGC program performs several tests such as a suffix test using several different types of morphological information, and the context frame rule test. Garside and Smith [63] define context frame rule as "***a rule designed by a linguist based on observation of data, which specified some information on a potential tag in the context of up to three tags on either side or that the potential tag was impossible in this context***". Furthermore, there are about 1,500 content word dictionaries containing those nouns, verbs, and adjectives that are exceptions to the computational rules used in suffix tests. They ran an experiment on samples of science writing and reported that their system correctly tagged 90% of the words.

- **TAGGIT system**

  Greene and Rubin developed [63, 68] the first pioneering tagger system for English, which is known as, TAGGIT. It was the first tagger which introduced the idea of providing a text corpus annotated with part-of-speech information as a useful tool for linguistic research. TAGGIT was used to initially tag the one million words of the Brown Corpus grammatically [63]. A small dictionary or lexicon containing a bout 3000 words was used in their TAGGIT program. The lexicon was tagged manually, that is, each word in lexicon was assigned its tag(s) . They used 3,000 context frame rules to disambiguate those words have more than one tag. Each word is initially checked to see if it is found in the lexicon.

If the word is found on the lexicon and has one tag, this tag is extracted and assigned to the word. If it has more than one tag, a set of context frame rules have been applied to assign the best tag to the word. In addition, a suffix list of 450 strings has been used to tag the word not found on lexicon. If the word is not found on the suffix List, the NN, JJ, and VB tags arbitrarily given to the word. A set 77 tags was used. The authors reported that TAGGIT system correctly tagged 77-78% of the words. Cutting et al. [47] point out, that the rest was done manually over a period of several years.

- **Fidditch system**

  Hindle [77] developed a tagger system to resolve the lexical disambiguation problem within a deterministic parser called Fidditch. It is designed to provide a syntactic analysis of text and to build phrase structure trees. Fidditch has the following components :

  - a lexicon of about 100,000 words listing all possible parts of speech for each word, along with root forms for inflected words.
  - a morphological analyzer to assign part of speech and root form for words not in the lexicon.
  - a complementation lexicon for about 4000 words.
  - a list of about 300 compound words, such as, *of course*.
  - a set of about 350 regular grammar rules to build phrase structure.

  a set of about 350 rules to disambiguate lexical category. Fidditch has a set of 46 tags (incltlding 8 punctuations), mostly enriched with inflectional features. Hindle tried to acquire a new set of disambiguation rules automatically from the tagged text of Brown corpus. The author claims that the performance of the acquired rule set is much better than the set of rules for lexical disambiguation written for the parser by hand over a period of several rules; the error rate is approximately half that of the hand written rules.

- **ENGCG: ENGlish Constraint Grammar system**

  Voutilainen [135] developed a tagger system called ENGCG (ENGlish Constraint Grammar) for ambiguity resolution and a finite-state syntactic parser, which is known as, the Finite-State Intersection Grammar. ENGCG tagger consists of two main rule components. The first component is a grammar specifically developed for resolution of part-of-speech ambiguities while the second rule component is a syntactic grammar. This syntactic grammar is able to resolve the pending part-of-speech ambiguities as a side effect. It uses only linguistic distributional rules. Their tagger consists of the following sequential components:

  - Tokeniser
  - ENGCG morphological analyser consists of Lexicon and Morphological heuristics rules.
  - ENGCG morphological disambiguator
  - Lookup of alternative syntactic tags
  - Finite state syntactic disambiguator

  The morphological analyzer assigns part of speech tags by looking each word up in the lexicon contains about 80,000 and then applying heuristic rules for still unrecognized words. The default tagging is noun when none of the rules apply.

A set of 139 tags was used. The author was tested the ENGCG system against a test corpus of 38,000 words and he reported it correctly tagged 99% of the words.

- **TBL: Transformation-Based error-driven Learning**

The most remarkable feature of Brills's tagger system [36,37] which makes it differs from other rule-based systems is that it automatically infers rules from a training corpus [106]. Brill's rule-based tagger is based on a learning algorithm called Transformation-based error driven learning (TBL). It is a technique for acquiring the rules automatically. Rules are learned by iteratively collecting errors and generating rules to correct them. Figure2.5 which is reproduced from the original figure from [37], illustrates the learning process of Brills's tagger system.

First, unannotated text is passed through an initial-state annotator. In this step, the system assigns to every word its most probable POS tag, as estimated from the small annotated training corpus. The training set is used here to determine the most likely (frequent) tag for each word. For unknown words, the most probable tag was guessed based on information such as the initial capital letter or suffix analysis. For example, xxxxxxx*ion* (where x represent any letter) would be tagged as a noun because this is (presumably) the most common tag for words ending in "*ion*". The tag in the second process compared to the true annotation as indicated by the annotations assigned in the manually annotated training corpus. A transformation can then be learned, which can be applied to the automatic annotated text to make it better resemble the manual annotation. The tagger has a small set of rule templates. The templates are of the form:

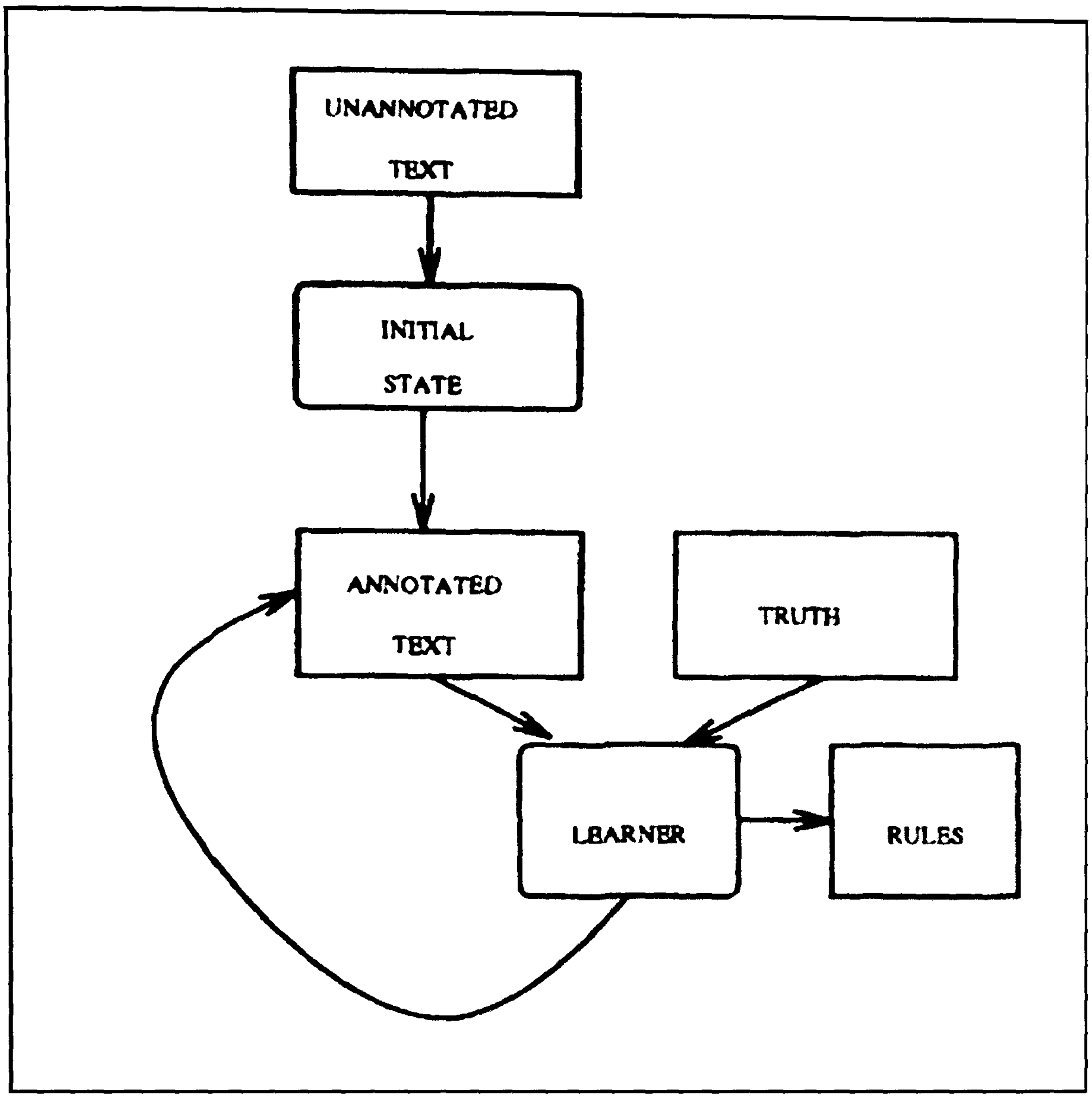


Figure 2.5: Transformation-Based Error-Driven Learning.

Change tag **a** *to tag* **b** *when the preceding (following) word is tagged* **z**

A maximum of three words preceding or following the inflected word in his transformation rule have been considered. The author also considered contextual transformation templates. These templates used to capture the relationships between words. The templates are of the form:

Change tag **a** *to tag* **b** *when one of the two preceding (following) word is* **w**

For example, one automatically acquired contextual transformation template is as follows:

Change the tag from **preposition** *to* **adverb** *when the word two positions to the right is* **as**. Based on the remarkable accuracy the system achieved (97%), the author showed that rule-based approach can achieves a high accuracy in comparison to systems that are based on a statistical approach.

2.5.2 Statistical Approach

The statistical approach is based on collecting statistics from existing corpora. Since it requires much less human effort than the rule-based approach, it is the most popular approach. Graside and Smith [63] point out, the general idea of this approach is that, when a sequence of words, each with one or more potential tags is given, the most likely sequence of tags can be chosen by calculating the probability of all possible sequences of tags, and then choosing the sequence with the highest probability. A statistical model of language is used to disambiguate the word sequence. A successful approved has been to model the sequence of tags in a sentence as a Hidden Markov Model (HMM). To obtain a statistical language model, one needs to estimate the model parameters, such as the probability that a certain word appears with a certain tag (lexical probability)[14], or the probability that a tag is followed by another (contextual probability)[15]. These probabilities are trained on a manually tagged corpus [78]. Also, this estimation is usually done by computing unigram, bigram or trigram (N-gram model)[16] frequencies on tagged corpora.

In order to define the goal of part-of-speech tagging systems with HMM models in a little more detail, we consider the problem in its full generality[17]. Let $w_{1...N} = (w_1, w_2, w_3, w_4, ...w_N)$ be a sequence of words, where N is the length of word sequence, $c_{1...N} = (c_1, c_2, c_3, c_4, ...c_N)$ be a sequence of part-of-speech or lexical

[14] probability of a part of speech given the word.

[15] probability of a part of speech given k previous parts of speech

[16] N-gram model using information about both lexical probabilities and contextual probabilities

[17] see ref [98] for more details

categories. When a word sequence is given, the goal of the part-of-speech system is to find the sequence of part-of-speech or lexical categories that maximizes the probability of a sequence of tags $c_{1...N}$ given a sequence of words $w_{1...N}$, that is :

$$T(w_{1...N}) = arg_{c1...N} max P(c_{1...N}|w_{1...N}) \tag{2.1}$$

where W_i denotes the i_{th} word in the word string and c_i denotes a part-of-speech tag assigned to the i_{th} word. After applying Bayes' rule approximation technique to approximate equation 2.1, it becomes as follows:

$$P(c_{1...N}|w_{1...N}) = P(c_{1...N}) * P(w_{1...N}|c_{1...N})/P(w_{1...N}) \tag{2.2}$$

After using further simplifying methods and approximation (see ref [23] for detailed explanations) to reduce equation 2.2, the final form of formula becomes as follows:

$$T(w_{1...N}) = arg_{c1...N} max P(c_i|c_{i-1})P(W_i|c_i) \tag{2.3}$$

The term $P(w_i|c_i)$ in formula 2.3 is called the lexical probability that can be estimated from a corpus of text labeled with a part-of-speech tag simply by counting the number of occurrences of each word by tag. It represents the probability that a given tag is realised by a specific word.

The term $P(c_i|c_{i-1})$ in formula 2.3 is called a bigram probability; it indicates the likelihood of a tag given only the preceding tag. It can be estimated simply by counting the number of times each pair of tags occurs and computing this to the individual tag counts. For example, the probability that a verb (V) follows a noun (N) can be calculated as follows :

$P(c_i = \mathbf{V}|c_{i-1} = \mathbf{N}) \approx \frac{Count(\mathbf{N}\ at\ position_{i-1}\ and\ \mathbf{V}\ at\ position_i)}{Count(\mathbf{N}\ at\ position_{i-1})}$

The equation below described the general formula for N-gram language model which bigram and trigram models could be simply derived from this general formula:

$$P(w_1^k) = \prod_{k=1}^{N} P(w_k|w_1^{k-1}) \tag{2.4}$$

where w_1^{k-1} denotes the word sequence $w_1, w_2, w_3, w_4, ...w_{k-1}$.

- Bigram Model can be derived as follow: $P(w) = \prod_k P(w_k|w_k - 1)$
- Trigram Model : $P(w) = \prod_k P(w_k|w_k - 2, w_k - 1)$

In HMM, we directly observe the sequence of words only, while the sequence of tags is hidden from the observer of the text; hence the term “Hidden Markov Model” is appropriate [63]. In addition, when the estimates used for the tag transition probabilities are derived from bigrams; that is, we have estimated the likelihood of tag given the knowledge that a particular other tag precede it, this model is called first-order HMM. A second-order HMM would uses tag transition estimates derived from trigrams, that is, we estimated the likelihood of a particular tag given the knowledge that two particular other tags precede it([63], p.105). The simplest model would be a most-likely-tag choice for each word.

Although the peak use of the statistical approach in part-of-speech tagging appeared in the eighties, the first attempt to use the statistical approach started with the work done by Stolz et al. [129] in 1965. Afterward, many researchers presented valuable tagging systems using the statistical approach. The seminal work is the CLAWS system using HMM [63–65, 93]. Merialdo [108] developed a POS tagger for English based on the probabilistic trigram model. Brants [35] proposed TnT, a statistical POS tagger. Many other systems were built using statistical approach such as, DeRose [117], Church [45], Cutting et al. [47], Weischedel et al. [137], Bahl and Mercer [32], Samuelsson [122, 123], and Kupiec [91]. However, A useful resource on the statistical approach can be found in the work done by Merialdo [108].
Some of the well-known statistical systems are now briefly discussed.

WISSYN system

The WISSYN grammatical coder is the earliest known POS tagger developed by Stolz et al [129] that uses probabilities to determine the grammatical classes (tags) of words. It has four component phases: the dictionary, morphology, ad hoc and probability phases. The first three phases accomplish the identification of the more frequently occurring words, and the last performs the prediction of remaining words.

- *Dictionary phase* : a small dictionary was used contains 300 words represent the most frequent words in English. It has not only the four main classes, noun, verb, adjective and adverb, but also many further categories, such as, pronouns, prepositions, negatives, determiners and other closed classes. Each word of the input text is checked against the dictionary entries. If a word is located in the dictionary, a tag is retrieved and assigned to the word. If a word is not found in this dictionary it is considered to be of one of the four main classes (noun, verb, adjective and adverb). At this stage the authors reported that an average of 60%-70%, of the words in a passage have been identified by this phase.

- *Morphology Phase* : this phase was constructed to deal with those words not located in dictionary during the previous phase. A small suffix dictionary contains 63 suffixes was used in this phase to determine the grammatical class of a word (morphological characteristics). For example, one such suffix test scans the word for its last four letters to determine if they match the *-ship* suffix, the *-ment* suffix, or any of a number of other four-letter endings. When a match is found, that word is assigned its appropriate tag.

- *Ad Hoc Phase* : the first two phases of WISSYN system operate on each word of the input sequentially as it is isolated. In other word, the context has no role. In this phase and the next one, the context of the remaining words has been taken into account. At this phase WISSYN system try to attempt clarification of some of those words identified in either of the first two phases but which remain ambiguous. For example, the word that, being a function word, is in the initial dictionary phase, but happens to have multiple class membership in different contexts (e.g., as in *that dog*, *that the dog jumped*, *the dog that jumped*, etc.).

An Ad Hoc Phase uses rules to determine the most likely tag. This phase can identify eight ambiguities, and include the various forms of that, and the verb to be. The authors point out, they used the same principle to that employed by Klein and Simmons [90] more generally, in that a specified set of frames is provided as diagnostic for particular identifications. For example, a routine which processes certain preposition-adverbs to determine their exact usage either as prepositions (e.g., *in* in *in the house* or as adverbs *come in from the cold.* The authors reported that also this stage identifies 10% of words on average.

- *Probability Phase* : this phase was constructed to use set of conditional probability tables to predict the four main grammatical classes for those words not tagged by the previous three phases. These probabilities were calculated from a manually tagged corpus that contained about 28,500 words. At this phase, the previous three tags, and the following three tags of a given word were examined. The authors state that this phase correctly tagged around 20% of words in texts.

A test set contains 1916 words was used to test the tagger system. The tagging of the text that occurred used a tag set consisting of 18 tags. An overall accuracy of WISSYN system, the authors reported that the system correctly tagged 92.8% of words.

CLAWS system

The original CLAWS(Constituent-Likelihood Automatic Word-Tagging System) system (version 1) [63–65, 93] was developed by Marshal I, Garside R, Leech G, and Atwell E over the period 1981 to 1983 at the Unit for Computer Research on the English Language (UCREL) at the University of Lancaster [111]. It was used to tag about one million words of (LOB) Corpus with 96-97% accuracy [103]. LOB tag set contains 133 tags was used with CLAWS version 1.

CLAWS system has five phases : pre-editing, tag assignment, idiom-tagging, tag disambiguation, and post-editing. Furthermore, it composed of four separate programs : PREEDIT, WORDTAG, IDIOMTAG, and CHAINPROBS. These programs associated with pre-editing, tag assignment, idiom-tagging, and tag disambiguation phases respectively([103], p.64).

- *PREEDIT* : concerned the preparation of text for processing by system.

- *WORDTAG* : assigns to each word a list of all possible tags for that word by using knowledge base or a set of rules to deduce the candidate word classes. WORDTAG program has a knowledge base contains 7200 words which are stored with a list of their candidate tags. This program was constructed not to disambiguate the lexical ambiguity of the word, but merely to assign a list of all possible tag(s) to each word. At this stage, if the word has only one tag, then the tag associated with the word and the word is assumed to be correctly tagged.
- *IDIOMTAG* : designed to assign a single tag to the compound symbol (composed of more than one word). For example, "such that" assigned one tag.
- *CHAINPROBS* : designed to choose one of the candidate tags to those words still have more than one tag at the end of the WORDTAG execution. CHAINPROBS program uses statistical analysis (bigram). Garside [64] estimated that 35% of LOB corpus words had more than one tag associated with them.

CLAWS was developed based on TAGGIT, except that CLAWS adopts a statistical technique for figuring out cases with ambiguous categories. It uses a table of probabilities of predecessor and successor tags to calculate the likelihood (probabilities) of all paths for each sequence of ambiguous words and eliminate sequences with low probability. The predecessor/successor probabilities of tags are extracted from a large proportion of the tagged Brown corpus. If tagging fails in ambiguous cases, context-dependent disambiguation is carried out based on the context frame rules of TAGGIT.

CLAWS version 2 on the other hand developed over the period from 1983-1986 to reduce the manual and automated pre-editing required by the system before any text could be analysed. It differs little from CLAWS version 1. The main difference is the automation of tag analysis itself. In addition, an extended tag set was used in CLAWS

2 containing 166 tags. Also, some change made in the WORDTAG program used in CLAWS 1 as part of the overall goal of removing any manual pre-editing. For example, WORDTAG program in CLAWS 2 dealt with capitalisation and abbreviation([103], p.78). The current version of CLAWS (version 4) was began in 1988 to undertake the enormous task of tagging the 100 million word British National Corpus (BNC). In this version of CLAWS, the authors separated the tagger from the tag set. They used BNC tag set[18]. In addition, they added a component that will enable it to handle SGML tags since the BNC was marked up with these tags([88], p.25). The authors reported that CLAWS 4 achieved an overall accuracy of 96-97% of BNC corpus words. However, it seems that CLAWS is used a hybrid technique since it has a rule based and statistical components.

PARTS system

Church [45] has also implemented a statistical tagger called PARTS. It used the lexical probability, which is the probability of observing part of speech **i** given word **j**, and the contextual probability, which is the probability of observing part of speech **i** given **k** previous parts of speech. The author calculates the product of the lexical probabilities and the contextual probabilities for each combination of ambiguous word sequences. The tag sequence that gets the highest probability is selected as the proper tagging result. PARTS differs from CLAWS in terms of the statistical model they used. The former used a trigram model while the later used a bigram model. Furthermore, PARTS does not have a rule based component. The author reported that PARTS achieved an overall accuracy of 95-99%.

[18] For more : http://www.scs.leeds.ac.uk/amalgam/amalgam/corpus/tagged/edited/ipsm_bncc5.prf.html (see also URCEL C7 tag set)

2.5.3 Advantages and disadvantages of rule-based and statistical approaches

1. **Rule-based Approach**

   Rule-based approach has some features and advantages which can be summarized as follows [36]:

   - A vast reduction in the amount of stored information because this approach represents knowledge in form of rules rather than stored data records. The model not need a huge manually tagged corpus to calculate probabilities.
   - The language model is written from a linguistic point of view and explicitly describes linguistic phenomena.
   - The model may contain many complex kinds of knowledge.
   - The written rules are easy to understand and to maintain.
   - High portability from one kind of text corpus to another.
   - Allow the construction of an extremely accurate system.

   While the disadvantage(s) of rule-based approach consist of:

   - Less transporting of the language model to other language.
   - The language mode requires a high labour of work and cost.
   - Usually the language models do not consider frequency information.

2. **Statistical Approach**

   The advantages of using a statistical approach [108] can be summarized as follows:

   - When a huge manually tagged corpus in the desired language is available, model transportation from other language becomes much easier.

- Language models consider frequency information.
- The probabilities can be estimated automatically from data.

While the disadvantage(s) of statistical approach consist of:

- Can't deal with unknown words.
- Model needs a huge manually tagged corpus to calculate probabilities.
- Model needs a huge matrix to represent the information.

Samuelsson and Voutilainen [58] and Chanod and Tapanainen [42] show that a rule-based tagger for English and French respectively can achieve better results than a statistical tagger.

2.5.4 Hybrid and Other approaches

Some implementations combine the statistical approach with the rule-based to build a hybrid POS tagger. Chanod and Tapanainen [42] developed a tagger that use a combination of both statistical and rule-based approaches for French. Kuba et al. [26] built a hybrid tagger for Hungarian. Schneider and Volk [126] trained the Brill tagger to German, French respectively.

Additionally, some different approaches have been used for building text taggers. Schmid [125], Marques and Pereira [100], Antonio et al. [114] developed a POS tagger using Neural Networks. They train a single-layer perceptron to produce the POS tag of a word. They reported an overall accuracy of 96.2%, 92.7% 92% respectivally. Schmid trained his tagger on 2 million words of the Penn Treebank corpus, and tested on 100,000 words of the corpus. Marques and Pereira trained their tagger on a very small Portuguese training corpus (15,000) words, and tested on 2229 words. While

Antonio et al. trained their tagger on 46,461, and tested on 47,397 words of the Wall Street Journal corpus.

Daelemants et al. [48] developed a memory-based part of speech tagger-generator. Memory-based systems are basically a form of k-nearest neighbor systems where set of cases (the training data) are kept in memory, and each test sample uses a distance metric to determine which training samples are closest. Then, the test sample is classified as the same class as the training samples. The set of cases in this approach usually consist of a word, its preceding and following context, and the POS of that word in the context. The author trained his tagger using a tagged corpus. To tag a new sentence. for each word and its context, the most similar case(s) where kept in memory are selected and extracting the POS tags from these cases. The tagger was trained on two different set size (two million words and 500,000 words). The author reported an average accuracy of 96.4%.

Decision trees have also been used to implement part-of-speech. A decision tree is a tree such that each internal node is a feature test and the leaves are classes to be assigned to the tested individual. The trees are constructed using statistical information. Marquiz and Rodriguiz [101] have implemented a POS tagger using decision tree that has been tested and evaluated on the Wall Street Journal corpus. The authors reported an overall accuracy of 96.16%.

Maximum-Entropy on the other hand is another technique used for building text tagger. This technique uses a statistical model can be classified as a Maximum Entropy model. It uses many contextual "features" to predict the POS tag. The Maximum Entropy model trains from a corpus annotated with Part-Of-Speech tags and assigns them to previously unseen text.

Ratnaparkhi [116] developed Maximum-Entropy POS tagger. The tagger was trained on 962687 words taken from Wall Street Journal and it has been tested on 133805 words. The author reported an overall accuracy of 96.6%.

2.5.5 Arabic POS Tagging Systems

As mentioned in chapter 1, the research in the Arabic computational linguistics in general and specifically POS tagging is growing significantly in recent years. The list below summarise most of the work done in POS tagging for Arabic.

- El-Kareh and Al-Ansary [54] used a statistical approach to describe semi-automatic POS tagger for Arabic.
- Shereen Khoja [87, 88] describe a hybrid tagger system that uses both morphological rules and statistical techniques in the form of hidden Markov models.
- Abuleil and Evens [15] describe a system for building an Arabic lexicon automatically by tagging Arabic newspaper text using some rules and morphological analysis.
- Andrew Freedman [62] implemented Brill's POS tagger for Arabic.
- Diab et al. [51] present a Support Vector Machine (SVM) based approach to automatically tokenized, part-of-speech tag in Arabic text.
- Habash and Rambow [71] presented a morphological analyser based on Support Vector Machine (SVM) based approach for tokenisation, part-of-speech tagging, and morphological disambiguation in Arabic.
- Alshamsi and Guessom [127] described HMM POS tagger system.

- Marsi et al. [102] employed MBT, a memory-based tagger-generator and tagger developed by Daelemans et al. [48] to produce a POS tagger for Arabic.
- Harmin [75] described a web-based Arabic tagger based on Buckwalter morphological analyser [39].
- Buckwalter [11] presented a morphological analyser for Arabic using lexicon rules.

All the systems described above were built to tag unvocalised Arabic text. Some of these systems are discussed below in more detail.

- **Semi-automatic Arabic tagger system**

  El-Kareh and Al-Ansary [54, 87] described a semi-automatic tagger to tag unvocalised Arabic text that waits for the user of the system to either confirm and accept the output of their system or change it. Their system used statistical techniques (HMM) and morphological rules. A small set of words were stored in its lexicon with its class and subclass as well as some inflectional features.

  Morphological rules was used to remove affixes and particles words from the testing text. The analysis result of morphological component represent the main class the word belong to or sub-class and inflectional feature which already stored in lexicon. The analysis result passed to the user. At this stage, the user may accept or reject the system result. In case the user reject the result, the word analysed once a gain and passed to the user. When the system completes its analysis without an accepted result from the user, the user in this case has an option to store his correct analysis.

  Statistical component in their system used to calculate statistics collected throughout the use of the system and stored later in lexicon. This component used to select the tag with the highest frequency without any intervention from the user. Testing corpus collected from Egyptian Al-Ahram newspaper was used to test their tagger. The authors report an accuracy of 90%.

- **APT : Arabic part-of-speech tagger system**

  Shereen Khoja [87, 88] developed the APT system that uses statistical and rule-based approaches. In the authors' point of view, the APT is the first tagger system for Arabic, for two reasons. First, it is the first fully-automatic tagger for Arabic. While the second is the aim of this tagger is to produce a POS tagged unvocalised Arabic corpus that may used as a useful tool for linguistic research.

  A manually tagged lexicon containing 50,000 word was used to extract several small lexicons. A training corpus containing about 10,000 word has been used to train her tagger. APT tagger has two main components; a rule-based component (stemmer) and a statistical component. Figure 2.6 which is reproduced from the original figure from ([88], p.78), illustrates how APT performs tagging.

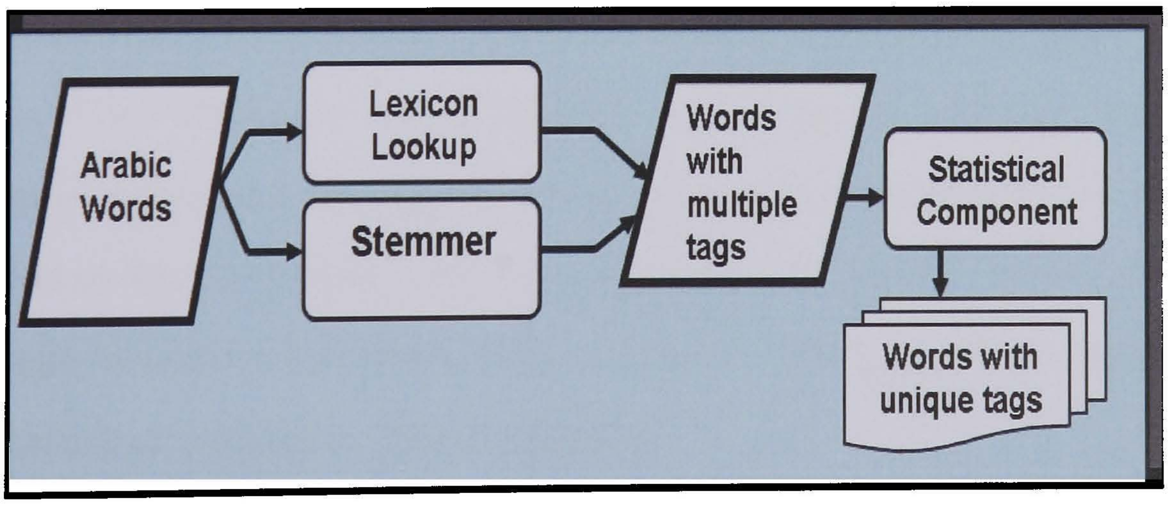


Figure 2.6: How APT performs tagging

APT performs the tagging process as follows. Each word is initially looked up in the lexicon. If the word is found in the lexicon, then it is assigned all the possible POS tags as found in the lexicon. The word is then passed to the stemmer regardless of whether it was found in the lexicon or not. The main function of stemmer is to remove all prefixes, suffixes and infixes to produce the root. The author does not mention the number of strings that were used in her stemmer affix lists.

If a word could not be stemmed, and was not found in the lexicon, then it is given the main tags (noun, verb, particle, residual and proper noun). As Khoja points out, at this point, each word has at least one or more tag. If a word has more than one tag, then this word (and its neighbors) are passed to the statistical component where the most likely tag is selected. APT statistical component uses the contextual and lexical probabilities to determine the most likely tag of the word. A corpus contains 1700 word has been prepared to test her tagger. The author report that APT system correctly tagged 86% of the words.

- **HMM Part-of-Speech Tagger for Arabic system**

  Alshamsi and Guessom [127] presented a Part-of-Speech (POS) Tagger for Arabic. The POS tagger resolves Arabic text POS tagging ambiguity through the use of a statistical language model developed from Arabic corpus as a Hidden Markov Model (HMM). The main goal behind the development of their POS tagger is to use it for Named Entity extraction. The input of the tagger is noun phrase and verb phrase Arabic sentences.

Like Khoja work, their system has two main components; stemmer and statistical component. The authors used Buckwalter's stemmer to stem the training data. A training corpus contains 27594 nouns, 23554 verbs, 5722 adjectives and 5384 proper nouns of Arabic news articles has been used. The training corpus tagged manually with 55 POS tag set developed by the authors.

During their tagging process, after the tokenizer converts the original input text into a list of words using the space as a delimiter, the resulting list is passed to the stemmer. A trigram language model has been constructed and used the trigram probabilities in building their HMM model. Each word has more than one tag been tagged by calculating the lexical and contextual probabilities. A test corpus containing 944 words was used to test their tagger system. The authors report that their tagger achieved 97%. This high level of accuracy is surprising me due to the fact that they have a small training corpus. However, as the author point out, they are in the process of enlarging the size of thier training corpus to reach one million words.

- **Web-based Arabic tagger system**

  Harmin [75] described a web-based Arabic tagger. As the author points out, this tagger was still in early development. The architecture of the tagger is based on 3-tiers; the client tier, the middle tier, and the database tier.

  The client tier is a web browser which sends the user's message to the web server and displays the returned results back to the user. The middle tier consists of a web server, a scripting engine and NLP module which is responsible for analysing the Arabic documents. While the third tier consists of an SQL server and the database used in the tagger system.

The tagger used the Buckwalter dictionary and his morphological analyser [11] distributed by Linguistic Data Consortium (LDC). The author collected about 42,000 HTML Arabic documents mostly from Al-Hayat Arabic newspaper. These documents were translated into XML format to test the tagger. The user can write a sentence and pass it to the tagger. Each word in the sentence is looked up in the dictionary, analysed and segmented into prefix, stem, and suffix. The result returned to the user contains all possibilities for the word. The author did not mentioned any information about the tag set they used and the accuracy their system achieved. However, based on their system snapshot, it seems they used the LDC tag set.

- **MBT:Memory-Based Tagger for Arabic**

  Marsi et al. [102] employed MBT, a memory-based tagger-generator and tagger developed by Daelemans et al. [48] to produce a POS tagger for Arabic. Memory-based tagging is based on the idea that words occurring in similar contexts will have the same POS tag.

  They used Arabic Treebank-1 corpus and LDC tag set. Their training corpus contains 150,966 words. The test set contains 15102 words, with 947 words do not in the training corpus (unknown words).

  MBT tagger has three modules; a lexicon module which stores for all words occurring in the provided training corpus their possible tags, the second module generates two distinct taggers; one for known words and the other for unknown words. The known-word tagger used a lexicon, while the unknown-word tagger attempts to derive as much information as possible from the surface form of the word, by using its suffix and prefix letters as features.

The authors report an accuracy of the tagger using the first two modules on the test corpus is 91.9% correctly assigned tags. They state that on the 14155 known words in the test set the tagger attains an accuracy of 93.1%; while on the 947 unknown words the accuracy is considerably lower: 73.6%. The third module on their tagger has been designed to improve the precision and recall in their system. The tagger integrated with morphological analysis was also built as a separate part in their work to enhanced the accuracy.

- **Buckwalter Arabic Morphological Analyser**

  Buckwalter [11] developed a morphological analyser for Arabic. It was produced by LDC and used for POS tagging Arabic text. The author used three lexicons :

  1. Prefixes lexicon contains 299 entries in the first release, 548 entries in the second release.
  2. Suffixes lexicon contains 299 entries in the first release, 906 entries in the second release.
  3. Stems lexicon contains 82,158 entries in the first release, 78,839 entries in the second release.

In addition, the lexicons are supplemented by three morphological compatibility tables used for controlling prefix-stem combinations, stem-suffix combinations, and prefix-suffix combinations. The data is written using his Arabic transliteration system[19] instead of original Arabic script. The author Morphology Analysis Algorithm (MAA) is based on four assumptions:

- Words are composed of three elements: prefix, stem, and suffix.
- The prefix can have 0-4 characters.
- The stem can have 1-infinite characters.
- The suffix can have 0-6 characters.

Each input word is segmented into three elements : prefix, stem and suffix. Each element is looked up in its respective lexicon. If all three word elements (prefix, stem, suffix) are found in their respective lexicons, then their respective compatibility tables used to determine whether they are compatible or not. Three questions are asked here :

1. Is the morphological category of the prefix compatible with the morphological category of the stem? (i.e., is the combination pair found in the list of compatible prefix-stem morphological categories?)
2. if so, is the morphological category of the prefix compatible with the morphological category of the suffix? (i.e., is the combination found in the list of compatible prefix-suffix morphological categories?)
3. if so, is the morphological category of the stem compatible with the morphological category of the suffix? (i.e., is the combination found in the list of compatible stem-suffix morphological categories?)

[19] For more : http://www.qamus.org/transliteration.htm

If the answer to the last question is "yes" then the morphological analysis is valid. The morphological analyser is produced all the variations of the input word included the short vowel and diacritics. The POS tag (which is stored in lexicons) for each variation also is produced. However, to those who are interested, Buckwalter Arabic Morphological Analyser can be found in [5].

2.6 Chapter Summary

This chapter contained a description of the POS tagging problem and NLP applications that may use POS tagger systems as their first stage. We defined the concept of corpus linguistics and POS tag set. The previous work on corpus linguistics and POS tag set has been discussed. In addition, the different approaches used to solve the problem have been examined and the previous work on POS tagging for English and Arabic has been explored.

This work employs the rule-based approach. AMT tagger presented in this work has two main rule components, these are : *pattern-based rules* and *lexical and contextual rules*. The basic idea of pattern-based technique is to generate automatically a lexicon of patterns instead of using manually tagged lexicon or training corpus which contains a set of Arabic words. The triggers in pattern-based rules depend on the patterns of text words. A novel algorithm to match the Arabic word in testing corpus with its correct pattern in patterns of lexicon has also been built. In addition, a small amount of hand-written rules and constraint rules have been used to assist the main technique to assign the correct tag to those words not tagged by pattern-based technique.

The next chapter will covers some of the basics of the Arabic language, describe the diacritics feature in Arabic and its importance in Arabic POS tagging. The POS tagger design and the main technique is described in chapter 5.

Chapter 3

Arabic Language and POS tagging

Objectives

- To present an overview of Arabic language and its script.
- To describe the diacritic feature in Arabic.
- To explain the importance of diacritic feature in Arabic POS tagging.
- To briefly define the Arabic grammatical system.

3.1 Introduction

The most prominent member of semitic languages family is the Arabic language. This semitic family includes also Hebrew, Amharic, Maltese and Syriac. They all share the pattern based morphology system. Furthermore, these semitic languages have a morphological system based on a root, usually consisting of three consonants, and a pattern structure. The root gives the basic lexical meaning of the word, while the pattern consists of vowels and it signals the grammatical significance of the word[1]. Bar-Haim et at. state that :

"*Semitic languages have rich inflectional systems and a template-based derivational morphology, which are manifested in a large variation of word forms* "([118], p.4).

Arabic is considered as the most widely used member of the semitic languages. It is spoken by more than 300 million Arabs around the world. Furthermore, it is also understood by more than 1.1 billion other Muslims. It has been a literary language since the 6th century A.D, and is the liturgical language of Islam in its classical form [70]. It is exhibiting a rich inflected and morphological system.

Arabic words, like words in other Semitic language, are written with consonants. Arabic language has several varieties, these are : *Classical Arabic*, *Modern Standard Arabic (MSA)* and *Colloquial (spoken) Arabic*. *Classical Arabic* is the language of Qur'an and classical literature. It is used as the language of religious practice throughout the Islamic world. *Modern Standard Arabic (MSA)* is the language of the media, education, and formal communication, which is understood by all Arabic speakers. *Colloquial (spoken) Arabic* is a local dialects of people throughout the Arab world [134].

The principal script used for writing the Arabic language is Arabic alphabet. It is composed of 28 letters. On the other hand, writing in Arabic language is unicase; the concepts that distinguish between Upper/Lower case letters do not exist. Furthermore, a cursive system from right to left is used in written Arabic[2]. The transliteration system of Arabic Alphabet and other diacritical marks used in this thesis are described in

[1] For more information : www.a-z-dictionaries.com/language/Arabic_dictionaries.html

[2] For more details : http://foolswisdom.com/users/sbett/arabic.htm

Appendix B on page 191.

While the Arabic alphabet was originally used to write the Arabic language, it has been adopted by other groups to write their own languages, such as Persian, Pashto and Urdu. A letter in the Arabic language is written in multiple forms, depending on where in a word a letter appears. It may appear in the beginning of a word (initial form), anywhere other than the beginning or the end of a word (medial form) and in the end of the word (final form)[3]. For example, in these Arabic words, (مَدارسَ[4], *mdArsa*, "schools"), (سمعَ, *smEa*, "to hear"), (حلمَ, *hlma*, "to dream"), the letter م *m (miim)* appears in initial, medial and final forms, respectively.

In Arabic language, a word may be an **original** word or **Arabized** word. The **original** words have two subcategories : *Derivative Arabic words* and *Fixed Arabic words*, while the **Arabized** words are nouns borrowed from foreign languages [56].

Derivative Arabic words, which are words belonging to the verb and noun classes, have been built from the same root and obey the Arabic derivation rules [72]. For example, the words, مكتب, *mktb*, "office", كتاب, *ktAb*, "book", كتبَ, *ktba*, "he wrote", are derived from the root كتب, *ktb*, "meaning of writing".

Fixed Arabic words, are words which do not obey the Arabic derivation rules. For example, the particles, "في", *fy*, "in", "من", *mn*, "from".

[3] For more detail : http://www.ancientscripts.com/arabic.html

[4] Since this thesis is written using Latex, sometimes, an additional diacritical mark may be added by the Latex system automatically and appear over some letters other than the last letter of the Arabic word, such as the fatha mark appearing above the second letter د in the word مَدارسَ. These marks have been ignored when dealing with the word or the pattern of the word. In this work, we are concerned only with the last diacritical mark.

3.2 Arabic script and diacritics feature

3.2.1 Brief history

Arabic script as well as latin script were derived from the first alphabet which was created by the phoenicians in 1300 B.C. The phoenician script comprises 22 letters as shown in figure[5] 3.1 and written from right to left without capital letters. Since the Phoenicians were living in Lebanon, Palestine and Syria (middle-east area), their script was born in lebanon.

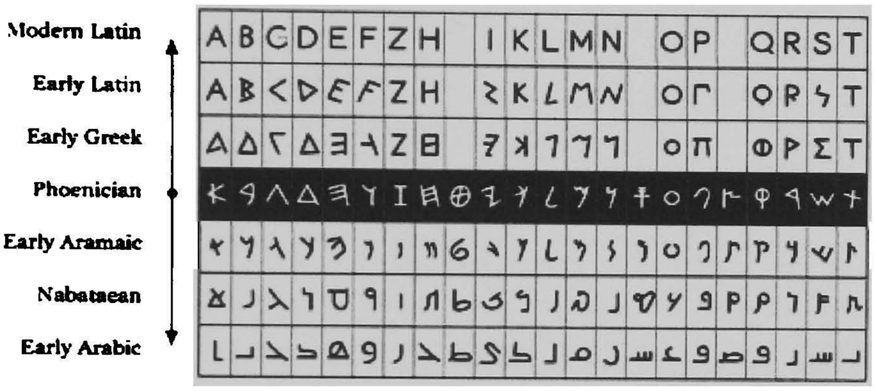


Figure 3.1: The origin of the Arabic script

Later, the Aramaic alphabet originated from the Phoenicians in 1000 BC. Later, the Nabatean script was born in the city of Petra, north of the Red Sea-Jordan in 100 BC, and spread all over the middle-east. The early Arabic alphabet was created in Kufa (Iraq) in the middle of the first century. The old Arabic alphabet consisted of around 17 letter forms without dots or diacritical marks. The calligraphic styles for the old Arabic alphabet was kufi style.

With the birth of Islam, the Quran was written with the Quranic kufi script. Since the missing dots and vowels in the old Arabic script are not clearly indicated, several

[5] Figure 3.1 taken from : http://29letters.wordpress.com/2007/05/28/arabic-type-history/

letters of the Arabic alphabet share the same shapes, for example, the letters ب, ت, ث، have the same shape (without dots), which definitely lead to confusion for Quranic readers. Since the Quran became the reason to reform all the Arabic scripts found in Arabia on one hand, and the number of non-Arab Muslims increased on the other hand, some reform was needed to avoid confusion and facilitating reading and learning of Arabic as well.

The first system of developing the old Arabic script was invented by Abul Aswad al Duali (688 AD) by placing large colored dots in order to help with pronunciation. Later, a uniform system to distinguish letters by using dots (in current usage) was developed by Al Hajjaj ibn Yusuf al Thaqafi. Lastly, Al Khalil ibn Ahmad al Farahidi (786 AD) devised a diacritical system to replace Abu al Aswad system.

By using the dot system, one, two, or three dots to letters with similar phonetic characteristics were added. A total of 28 letters containing three long vowels is obtained. This unified well structured Arabic script was developed for the writing of the holy scripts of the Quran in the 7th century with the development of calligraphic styles as well. Later the Quran was written with the Quranic Naskh style[6].

On the other hand, the Phoenician alphabet was used as a model by the Greeks. letters for vowels were added by the Greeks. Afterwards the Greek model became the model for early latin, and ultimately all Western alphabets[7].

[6] For more: http://sakkal.com/ArtArabicCalligraphy.html

[7] http://www.answering-islam.org/Green/seven.htm

3.2.2 Arabic Diacritical Marks

The Arabic language has two types of vowels (long and short vowels). The long vowels are three letters form a part of Arabic letters (Arabic alphabet)[8]. The short vowels are three small vowel marks (see Table 3.1), which do not form part of the Arabic letters. These marks are placed above and below the Arabic letter.

Fatha /a/ بَ	Damma /u/ بُ	Kasra /i/ بِ
Mark above the letter	*Mark above the letter*	*Mark below the letter*

Table 3.1: Arabic short vowels diacritics

Fatha represents the sound of /a/ in bag, damma represents the sound of /u/ in put and finally, kasra represents the sound of /i/ in sit.

Moreover, there are other five diacritical marks[9]. Three of them as shown in Table 3.2 called nunation (Tanween Fath pronounced /an/, Tanween Damm pronounced /un/, Tanween Kasr pronounced /in/). Nunation is the doubling of the short vowels used at the end of indefinite nouns.

Tanween Fath /an/ بً	Tanween Damm /un/ بٌ	Tanween Kasr /in/ بٍ
Mark above the letter	*Mark above the letter*	*Mark below the letter*

Table 3.2: Nunation (Tanween) Vowels diacritics

Finally, the last two marks in use are sukun (absence of a vowel) which means that the consonant is not followed by a vowel and gemination (Shadda) which means a duplication of the consonant; these marks are shown in Table 3.3.

[8] The three long vowels letters are : ا Alif, و waaw, ي yaa.

[9] Somtimes, researchers distinguish between short vowel marks and diacritical marks. In this thesis, we use the term *diacritics* to represent all marks (including short vowels marks)

Sukun بْ	Shadda بّ
Mark above the letter	*Mark above the letter*

Table 3.3: Sukun and Shadda vowels

In Arabic language, diacritics can be used in Qura'n text, in other religious texts, in classical poetry, in textbooks of children and foreign learners and in complex texts to avoid ambiguity. The diacritic marks may be assigned to each character of the Arabic word, in this case, an Arabic word is called fully-vocalised. When the diacritical marks are assigned to most letters of the word, but not each, an Arabic word in this case is called half-vocalised. An Arabic word is partially-vocalised when the the diacritical marks assigned to one or maximum two letters in the word [56]. Table 3.4 shows an example on each of the vocalisation state of the Arabic word.

Translation :	"I wrote the lecture today evening"
Transliteration :	*ktbtu AlmHADrpa msA'a Alywmi*
Arabic Sentence in :	
Full-vocalized :	كَتَبْتُ المُحَاضَرَةَ مَسَاءَ اليَوْمِ
Half-vocalized :	كَتَبْت المُحَاضَرَة مَسَاء اليَوْم
Partial-vocalized :	كتبْتُ المحَاضرةَ مسَاءَ اليومِ

Table 3.4: Vocalisation state of the Arabic word

3.3 Importance of the diacritic feature in Arabic POS tagging

The base of POS tagging is that many words are ambiguous regarding their grammatical category [109]. For instance, the word "ذهب",*dhb* in the unvocalised Arabic sentence presented in Table 3.5[10], (which either means "has gone" or "gold"), can be

[10] The tags used in sentence presented in Table 3.5 have been described in more detailed in chapter 4.

a verb or a noun. Due to the fact that the sentence is unvocalised, this lexical ambiguity is predictable. Thus, it requires an adequate context or/and an adequate knowledge about the semantic information to be resolved [109].

Adding semantic information knowledge to an unvocalised Arabic text is not an easy task, because it is very difficult to predict the semantic meaning with the missing diacritics (at least one diacritical mark) in Arabic text. Furthermore, removing the ambiguity based on an adequate context requires a more sophisticated technique, such as a statistical technique, which still suffers from many disadvantages, including : needs a manually tagged huge lexicon or training corpus, can't deal with unknown words and needs a huge matrix to represent the statistical information.

Arabic Sentence : POS Tag :	ذهب الطَالب مسرعَا NuAj NuCnNm VePe NuCn NuCnAc NuCn
Transliteration :	msrEA AltAlb dhb
Translation :	The Student has gone quickly

Table 3.5: Unvocalized Arabic sentence and its POS tags

The lack of diacritics in Arabic texts is presented as a major challenge to most Arabic NLP tasks, including parsing [95]. The use of diacritics in Arabic texts are extremely important. The list below summarises the importance of using diacritics in Arabic language :

1. Adding semantic information to the words leads to resolving ambiguity in the meaning of words. For example, adding the short vowel (Fatha mark) to the last letter of the word "ذهب" presented in Table 3.5 to become "ذهبَ" causes the removal of ambiguity in the meaning of the word (has gone).

2. Determining the correct POS tag to the words in the sentence. For example, the

word "ذهب" definitely belongs to the Verb class.

3. Indicating grammatical functions to the words, differentiating the word from other words, and determining the syntactic position of the word in the sentence. For example, short vowels used to indicate mood, aspect and voice endings for verbs and case endings for nouns.

4. Indicating the correct pronunciation of words, correct syntactical analysis which leads to reducing problems for NLP applications such as text-to-speech or speech-to-text, and removing the semantical confusion of Arabic readers [139] [95] [55].

The above list shows that using the diacritics in text is important to differentiate the word from other words and determine the syntactic position of the word in the sentence such as nominative, accusative, and genitive.

In addition, these diacritic marks determine the inflectional features[11] of the sentence words, such as, gender, person, number, noun case, and verb mood.
For example, in the following Arabic sentence :

Arabic Sentence :	حضرت الدرس كله
Transliteration :	*HDrt Aldrs klh*
Translation :	"(I, She) attended all the lesson "

In the above sentence, it is very difficult to determine the inflectional features for the word حضرت, "attended" with the diacritics missing, especially the last diacritical mark (ending case). Neither the context nor the word itself can provide any information on inflectional features for such a word. Thus, the last diacritical mark helps not only in determining the correct part-of-speech (general tag) of the words in the sentence, but

[11] Arabic Inflectional Features described in chapter 4, Section 4.2.

also in providing full information regarding the inflectional features for the sentence words (detailed tag).

The possible last diacritical mark (case ending) of the word حضرت and the inflectional features for each case can be seen in Table 3.6.

Case ending	Inflectional features
(Damma mark) حضرتُ, *HDrtu*	First person, singular number, Masculine gender, indicative mood
(Sukun mark) حضرتْ, *HDrtx*	Third person, singular number, Feminine gender, jussive mood
(Fatha mark) حضرتَ, *HDrta*	Second person, singular number, Masculine gender, subjunctive mood
(Kasra mark) حضرتِ, *HDrti*	Second person, singular number, Feminine gender, jussive mood

Table 3.6: The possible last diacritical mark (case ending) of the word حضرت

The correct tags of the sentence presented in Table 3.5 where the suitable diacritical mark has been added to the last letter of every word in the sentence are shown in table 3.7.

Arabic Sentence: POS Tag:	ذهبَ الطالبُ مسرعاً NuAj NuCnNm VePe
Transliteration:	msrEAF AltAlbu dhba
Translation:	The Student gone quickly

Table 3.7: Partially-vocalised Arabic sentence and its correct POS tag

3.4 Arabic Major grammatical part-of-Speech

A word can be defined as something that is uttered, intelligible, and has a full meaning [69]. According to Arab grammarians, words in Arabic are classified into three Part-of-Speech categories : *Verb*, *Noun*, and *Particle*. Each category has its meaning and its recognisable signs as described below.

3.4.1 Verb

The category of verb is defined as a word denoting an action and may be combined with the particle [134]. In Arabic, traditionally, two verb forms are recognised; the *Perfect* (past) and *Imperfect* (present). The third form, the *Imperative*, has been considered as a variant of Imperfective by Arab grammarians. Each form has its distinguishing signs. Furthermore, an Arabic verb has a temporal aspect inherent in it [69].

- **Perfect Verb**

  The perfect verb indicates a state or a fact in the past [76]. It follows the pattern of the root (ground form)[12] فعل, *fEla*, "do". For example, the root كتب, *ktba* , "wrote", has the basic meaning of writing. It can be suffixed with many letters. For instance, the letter ت, *taa*. The suffix represents more inflectional features to the word, such as, person, gender, number, and mood. For example, the words, كتبتُ, *ktbtu*, "I wrote" (first person, masculine), كتبتَ, *ktbta*, "you wrote" (second person, masculine), كتبتْ, *ktbtx*, "she wrote" (third person, feminine) and كتبتِ, *ktbti*, "you wrote" (second person, feminine).

  The above example shows that adding the diacritical mark on the last letter of the word helps not only in determining the lexical category of the word, but also in defining the inflectional features of the word.

- **Imperfect Verb:**

  The imperfect verb expresses an action still unfinished at the time to which ref-

[12] the ground form and the derived forms are described in more detail in Chapter 5

erence is being made [76]. Also, it can be prefixed with one of the following four letters (called letters of present): أ, ي, ن, ت. For example, the words, أكتبُ, *Aktbu*, "I write", يكتبُ, *yktbtu*, "he write", نكتبُ, *nktbtu*, "we write", تكتبُ, *tktbtu*, "she write". In addition, the imperfect verb can accept a particle. For example, لن يكتبَ, *ln yktba*, "he will not write".

- **Imperative verb :**

  The imperative verb indicates an action demanded to be carried out in the future [76]. It always comes in the second person. For example, the word أكتبْ, *'aktbx*, " write !". Like the perfect and imperfect verbs, the imperative verb can be suffixed with the letters ي, *yaa*, ا, *Alif*, ن, *nuun*, و, *waa* to represent the inflectional features of the word (see Table 3.8).

Arabic Word	Inflectional features
أكتبْ, *'aktbx*, "you (write !)"	second person, singular, Masculine
أكتبي, *'aktbyx*, "you (write !)"	second person, singular, Feminine
أكتبا, *'aktbA*, "you (write !)"	second person, Dual, Masculine/Feminine
أكتبوا, *'aktbwA*, "you (write !)"	second person, plural, Masculine
أكتبنَ, *'aktbna*, "you (write !)"	second person, plural, Feminine

Table 3.8: Samples of imperative verbs and their inflectional features

3.4.2 Noun

The category of noun is defined as a word denoting an essence and may be combined with an article [134]. In Arabic, a noun has no temporal aspect. As Arab grammarians described, a noun has a set of signs that are used to distinguish it from verbs and particles [69]. The list below describes these signs:

1. **Kasra mark**

   A noun can receive a kasra vowel mark when it is in the genitive case. In Arabic, the words which belong to the verb category never receive a kasra mark. For example, مكتبِ, *mktbi*, “office”.

2. **Nunation mark**

   In Arabic language, neither the verb nor the particle receives any nunation mark. A nunation mark appears only on the final letter of Arabic word which belongs to noun category. These marks indicate that these words are indefinite. For example, كتابٌ, *ktabun*, “book”’.

3. **Vocatives**

   A noun in Arabic may be placed in the vocative position, if it follows vocative particle. For example, يا حسن, *ya hasan.*

4. **Definition by an article** ال (*the* in English)

   A noun in Arabic is definite when begin with an article ال, “Al”. For example, الكاتبُ, *AlkAtbu*, “the writer”.

However, it is important to draw attention here, that it is not necessary to find one or all of these signs to define a word as a noun. For example, in the following sentence كَتَبَ مدرسَ الحَاسوبِ الدرسَ, “ the computer teacher wrote the lesson ”, the word مدرسَ, “teacher” is a noun, and none of the above signs is used to distinguish this word. In this case we use the pattern of this word to distinguish it [69].

3.4.3 Particle

The category of particle is included the in remaining words. Particles used to assist other words in their functions in the sentence [134]. In Arabic, the particle does not have a meaning without being attached to a noun or a verb. Furthermore, particles do not accept any of the signs that distinguish between nouns and verbs [69]. An example of Arabic particles is في, *fy*, “in”, من, *mn*, “from” and عن, *about*.

In contrast of Hebrew language as a member of semitic languages, Griess state that : “***Hebrew shows similarity to Arabic in terms of its grammatical constituents of verbs, nouns, and particles. The Hebrew nouns can certainly be in the genitive position, mimated (instead of nunated), defined by ה (instead of ال), and be predicated in the same way as in Arabic. Nevertheless, when contrasted to Arabic, Hebrew enjoys a less complicated particle system***”([69], p.24).

3.5 Arabic Grammatical System

There are two main categories of grammatical analysis in Arabic (see figure 3.2): *Morphology* and *Syntax*. The former is the study of the form of the word while the later is the grammatical arrangement of words in the sentence. On the other hand, Arabic morphology has two subcategories: *Derivational*, how words are formed, and *Inflectional*, how words interact with syntax, such as singular, dual and plural [120].

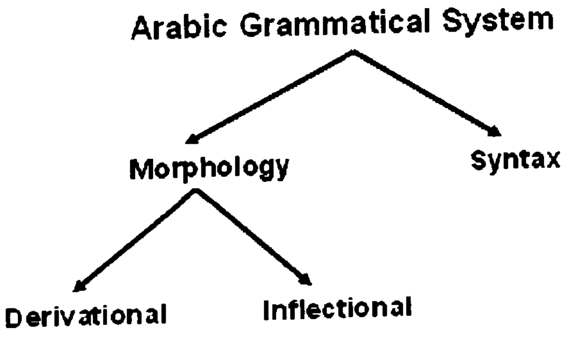


Figure 3.2: The Arabic grammatical system

3.5.1 Morphology System

In Arabic morphology, the Arabic word formation is based on a root [138]. Many affixes can be attached to the root to form Arabic words. Arabic morphology consists of a system of consonant roots which interlock with other consonant and vowels to form word stems. The stem is formed by substituting the characters of the root into certain verb forms [120].

A great number of other forms can be derived from the ground form (root) by inserting a long vowel, lengthening the medial letter of the root, and/or adding consonantal prefixes to produce a new word with a new meaning that still shares the basic meaning of the root [138]. For example, the root " كتب" *ktb* has the basic meaning of writing. The root may be conjugated in many forms[13]. Samples of the words that can be formed and derived from the same root " كتب" *ktb* are shown in Tables 3.9, 3.10, 3.11.

Arabic Word	Transliteration	Translation
كتَب	*ktba*	he wrote
كتبوَا	*ktbwA*	They wrote
كتبْت	*ktbtp*	She wrote
كتبنَا	*ktbnA*	We wrote
كتبْتُ	*ktbtu*	I wrote
كتبتَ	*ktbta*	You wrote

Table 3.9: Samples of past tense (perfect) verb forms

Arabic words are modified not only by number, person, gender and tense, but also by case and mood, definiteness and indefiniteness [22]. According to Arab grammarians, from every verb, a verbal noun (Infinitive), a noun of time, an adjective noun, a noun of place, diminutive noun, an instrument noun, a present (active) participle and past

[13] For more information : http://wahiduddin.net/words/arabic_glossary.htm

Arabic Word	Transliteration	Translation
يكتبُ	*yktbu*	he writes
يكتبونَ	*yktbwna*	they write
تكتبُ	*tktbu*	She wrote
نكتبُ	*nktbu*	we write
أُكتُبْ	*Ouktubx*	write !

Table 3.10: Samples of present (imperfect) and imperative verb forms

Arabic Word	Transliteration	Translation
كاتبٌ	*kAtbp*	writer
مكتوبٌ	*mktwbp*	letter
كتابٌ	*ktAbun*	book
مكتبٌ	*mktbp*	office
كُتيبٌ	*kutybun*	booklet

Table 3.11: Samples of additional forms such as verbal, diminutive, Adjective nouns created from the same simple root كتب

(passive) participle may be derived [120].

3.5.2 Syntax System

The syntax system in Arabic refers to the grammatical arrangement of words. As Arab grammarians described, there are two types of sentences[14]: *Verbal* and *Nominal* sentences.

- **Verbal sentence**

  A verbal sentence is simply one which begins with a verb followed by a subject. The verb in verbal sentence is always in singular form, where the subject may be singular, dual or plural. For example, in the following sentences :

[14] For more information :http://www.multimediaquran.com/quran/arabic/grammar/sentence.html

1. كتبَ الطَالبُ الدرسَ
2. كَتَبَ الطَالبَانِ الدرسَ
3. كَتَبَ الطلَابُ الدرسَ

The above sentences may be translated to English as “wrote the student(s) the lesson”. But, it really means “the student(s) wrote the lesson”. The underlined words in the above sentences represent the subject in each sentence. The subject in the sentence 1, 2, and 3, is singular, dual and plural, respectively, where the verb كَتَبَ, “wrote” is always in the singular form.

- **Nominal sentence**

  A nominal sentence is one which begins with a noun or subject. The verb in an Arabic nominal sentence must agree with the subject in number and gender as shown in the following sentences :

1. الطَالبُ كتبَ الدرسَ
2. الطَالبَانِ كتبَا الدرسَ
3. الطلَابُ كتبوَا الدرسَ

The underlined words in the above nominal sentences “ the student(s) wrote the lesson ” represent the verb in each sentence. The verb is changed to agree with the subject in number and gender.

The above two types of sentences, which are VSO[15] and SVO respectively, are viewed as being independent and neither of them is derived from the other. However, the Arab grammarians assumed that the subject never precedes its verb, and take VSO as the underlying word order for Arabic [19].

[15] V = Verb, S = Subject, O = Object

3.6 Chapter Summary

This chapter briefly described an overview of Arabic language and its script. The diacritic feature and its importance in reducing the lexical ambiguity and providing more semantic information to the word text also addressed. Arabic as other semitic language based on the fact that words are derived morphologically from roots. Many words are derived with a new meaning that still share the basic meaning of the root. The application to the root of a large number of morphological patterns determines the categorical status of the resulting word.

All Arabic words can be theoretically reduced to roots. To deduce a root from the pattern and to decide which pattern has been imposed on the root is a prerequisite skill for using an Arabic dictionary. According to Arab grammarians, there are three major part-of-speech : verb, noun, and particle. Arabic not only has complex morphological system but also exhibits a highly inflectional system as well. In next chapter, an Arabic inflectional features will be describe in more detail beside the tag set which consider as a prerequisite step toward developing a tagger system.

Chapter 4

Tag set Design

Objectives

- To define the tag set design criteria.
- To describe the Arabic inflectional features.
- To explain the developed Arabic tag set hierarchy and design.

4.1 Tag set design criteria

Atwell [30] presented a number of criteria to take into account while developing the POS tag set. These criteria have been taken into account when developed the tag set. The list below summarises these criteria in a little more detail:

1. **Mnemonic tag names**

   This criteria is concerned with the name of the tag. The name of the tag must be chosen in such a way that makes it easy for the user to remember the classes of the text. Since producing a tagged corpus where the text has been enriched with linguistic information to be used in many NLP applications is the main anticipated outcome of ATM tagger presented in this work, the tag names have been chosen to help linguists and NLP developers to remember the lexical class of each word. For example, `Ve` for verb, `Nu` for noun and `Pr` for particle.

2. **Underlying linguistic theory**

   The tag set developer should take into account that the tag set should cover aspects of the theory of language and the characteristics of that language (i.e, inflectional feature). The developed tag set presented in this chapter follows the Arabic grammatical system and is based upon the main three POS classes (verb, noun, particle); these tags are enriched with inflectional features [24].

3. **Classification by form or function**

   Usually, the lexical classes are defined in terms of paradigmatic forms (representative set of the inflections of a noun, verb, etc), and syntagmatic functions (syntactic function of the words). Since the short vowels and other diacritical marks are available in our testing corpus, these vowels can encode the grammatical class or feature information [30]. This criteria is taken into account during the course of developing our tag set.

4. **Idiosyncratic words**

   Arabic like any other language has a number of words with special idiosyncratic behavior. These words do not have patterns to follow, such as words belonging to a particle class. Similarly, the English language has a number of words with special idiosyncratic behavior. These words do not fit into traditional parts of speech. For example, Brown and LOB tag sets analysed "a" as article tag AT, but UPenn tag set analysed it as determiner DT. Our developed tag set analysed these words based on their roles in the text as Arab grammarians classified these words.

5. **Categorisation problem**

   The vowels (last diacritical mark) in our testing corpus add more linguistic information and reduce the ambiguity in categorising words. Most tags in the developed tag set are detailed tags. Each tag being defined clearly and unambiguously. We considered each main POS class or subclass as a unique tag, so that all the words in the testing corpus can be tagged consistently.

6. **Tokenisation issues: what counts as a word?**

   Arabic text like English text needs a tokenisation process. It is responsible for locating an untagged input text and identifying words, punctuation marks, numbers and other marks. Some words need a combined tag. For example, the word ويكتبُ, *wyktbu*, "and he writing" has the following tag `PrCo+VePiMaSnThDc` . This issue has been taken into account when developing our tag set.

7. **Multi-word Lexical Items**

   The Arabic language has very few idiomatic phrases (Multi-word Lexical Items). It may appear in some proper nouns (e.g, Ala'a Alddin) but is treated as one word and has one tag.

8. **Target users and/or application**

   Since one of the aims of the tagged corpus is to use it for developing educational application for teaching purposes, the tags in the developed tag set have been designed to achieve maximum target use and customer satisfaction as well.

9. **Availability and/or adaptability of tagger software**

   Arabic as well as other semitic languages has a morphological system based on a root (usually consisting of three consonants or letters) and a pattern structure. The main technique in this work is based on the pattern of the word. It is interested to note that this technique may also be valid for other semitic languages especially Hebrew language, since this language like Arabic has a morphological system based on a root and a pattern structure on one hand, and has the diacritic feature on the other hand. However, the tag set presented in this work is based upon the three main POS classes, their sub-classes and inflectional morphology. Thus, the guiding principle was compatibility with Arabic grammar tradition [30].

10. **Adherence to standards**

    EAGLES guidelines outline a set of features for tagsets[1]; these guidelines are designed to help standardise tagsets for what were then the official languages of the European Union. EAGLES tags are defined as sets of morpho-syntactic attribute-value pairs (e.g. Gender is an attribute that can have the values Masculine, Feminine or Neuter) [74]. Arabic has its own structure, feature (e.g diacritics), linguistic attributes (e.g dual number and jussive mood), which make this language different from the languages for which EAGLES was designed [89]. In addition, there are other differences in the order of the constituents within the sentence. For example, in Arabic, adjectives follow the noun which they modify [58]. Despite the fact that some classes from traditional Arabic linguistics and grammar have not been compatible with EAGLES guidelines, some of the English translations of class and feature names used in the developed tag set were drawn from standard terminology found in the EAGLES guidelines [30].

11. **Genre, register or type of language**

    This criteria is not fully applied in our developed tag set. The tags were developed to cover written Arabic text. The corpus in this work is partially-vocalised

[1] http://www.ilc.cnr.it/EAGLES96/annotate/

Arabic corpus contains written Arabic text. It does not contain spoken text.

12. **Degree of delicacy of the tag set**

    The tags were developed with a good level of granularity, leading to cover all the sub-classes of the three main POS classes used in Arabic grammar. Each tag is enriched with inflectional features, which seem to help the linguists to develop a robust educational system for learning Arabic as an example. However, the developed tag set contains 161[2] detailed tags, 101 nouns, 50 verbs, 9 particles, 1 punctuation including 28 different POS general tags. Arabic language is characterised by having a rich and an extensive morphological system as well as an inflectional system. Therefore it is natural that the tag set should be a richly articulated tag set, providing distinct codings for all classes of Arabic words. At the same time, as Elworthy [59] point out, if all of the syntactic variations which are realised in the inflectional system for highly inflected languages, such as Arabic or Hungarian were represented in the tag set, there would be a huge number of tags, and it would be practically impossible to implement or train a simple tagger.

4.2 Arabic Inflectional Features

Grammatically, *inflection* is the marking of a word in written text to reflect grammatical information, such as gender, tense, number or person[3]. Arabic is a highly inflected language [97]. It exhibits a rich inflectional morphology system. Inflectional morphology is used to express grammatical relations between words in sentence [16]. The list below describe the Arabic inflectional features in more detail:

[2] detailed tags included inflectional feature while general tags represent the name of the main class and its sub-class without inflectional feature

[3] For more information : http://en.wikipedia.org/wiki/Inflection

4.2.1 Gender

Nouns and verbs in Arabic are morphologically marked for the inflectional feature "Gender". Arabic has two genders: masculine and feminine. Like English, male persons are masculine, female persons are feminine, but things may be masculine or feminine. For example, in English, gender is indicated in the third person singular personal pronouns as the feminine "she", the masculine "he" , and the neuter "it". The personal pronoun "it" can refer to certain creatures of either sex (baby, cat) and to sexless things (beauty, book) [21].

In Arabic a word such فريقٌ, *fryqun*, "team" may refer to (masculine or feminine) gender. Nouns in Arabic may be recognised as feminine singular nouns by their grammatical form. For example, nouns ending by ة (Ta Marbota), such as جنةٌ, *jntpun*, "garden" or ending by اء, such as صحراء, *ShrAp*, "desert". Also, nouns may be recognised as feminine plural nouns which are formed by adding the suffix ات such as جميلاتٌ, *jmylAtun*, "beautiful women". Masculine plural nouns may be recognised by adding the suffix ونَ or ينَ, such as مدرسونَ, *mdrswna*, or مدرسينَ, *mdrsyna*, "teachers".

In terms of Arabic verb, since the verb in Arabic is a combination of a verb and a pronominal suffix or prefix, these pronominal affixes represent inflectioanl features, such as, gender, number, person and mood marker. In general, gender terms and forms in Arabic as well as English do not always refer to biological gender [21, 61]. However, the inflectional feature gender in our tag set has been classified into three genders: *masculine, feminine* and *neuter.*

4.2.2 Number

In Arabic, number is the inflection feature governing nouns and verbs. Unlike English, Arabic has three forms of number:*singular, dual* and *plural*. Singular denotes only one, dual denotes two individuals of a class or a pair of anything and plural denotes three or more [21]. The dual is formed by adding the dual suffix انِ or ينِ. For example, the words ولدٌ, *wldun*, ولدَانِ or ولدينِ, *wldAni* or *wldyni* and أولادٌ, *OwlAduN*, which mean " *a boy* ", " *two boys* " and " *boys* " indicate singular, dual, and plural respectively.

4.2.3 Person

In Arabic, verbs and only personal pronouns inflect for three persons: the speaker (*first person*), the person spoken to (*second person*), and the person spoken about (*third person*). The first person in the singular denotes the speaker. In the plural it denotes the speaker plus anybody else, one or more. The second person denotes the person or persons spoken to. The third person denotes those other than the speaker or those spoken to [132]. For example, the personal noun أنَا, *AnA*, "I", أنتَ, *Anta*, "you" and هوَ, *hwa*, "he" indicate first, second, and third person respectively.

4.2.4 Mood

Arabic Verbs have three moods: *Indicative* , *Subjunctive* and *Jussive* (Imperative). The mood markers are often short vowel marks placed at the end of the word (suffixes) such as fatha, damma and kassra or sukun mark. For example, damma /u/ for indicative and fatha /a/ for subjunctive. On the other hand, mood may be determined by particles which govern or require a certain mood [120]. For example, the negative particle لم, *lm* requires the jussive mood on the following verb such as the words لم يكتبْ, *lm yktbx*, "does not write". The mood of the verb word يكتبْ is jussive.

4.2.5 Case

In Arabic the term "case" refers to inflectional marking. Arabic nouns have three cases: *nominative, accusative* and *genitive*. They indicate the syntactic function of the word and its relationship with other words in the sentence (e.g. singular, dual, masculine plural, feminine plural forms take special case endings) [120]. These cases are indicated by short vowel marks placed at the end of the word (suffixes). For examples, the words الدرسَ *Aldrsa*, الدرسُ, *Aldrsu* and الدرسِ, *Aldrsi* which mean "the lesson", indicate nominative, accusative and genitive respectively.

4.2.6 State

Arabic nouns are marked for *definiteness* or *indefiniteness*. In Arabic the definite article ال, *al* used as a prefix to indicate definiteness. It is not an independent word like "the" in English. In Arabic, "nunation" (tanween) marks are used as suffixes to indicate indefiniteness [120]. For examples, the words الكتَابُ, *AlktAbu*, "the book", كتَابٌ, *ktAbun*, "a book" indicate definiteness and indefiniteness respectively.

4.3 ARBTAGS-The developed Tag set

4.3.1 ARBTAGS Hierarchy

The tag set hierarchy presented in this work follows the tradition of Arabic grammar. Most of the Arabic grammar dictionaries, such as a dictionary of Arabic grammar [57] classified the Arabic words as shown in chapter 2, figure 2.3.

As Arab grammarians described, each Arabic word belongs to one of the three main classes; verb, noun or particle.

1. **Verb**

In Arabic grammar, the main class (verb) comes with three sub-classes shown in figure 4.1 (see also figure 2.3 in chapter 2). These sub-classes are classified according to the tenses of verb in Arabic.

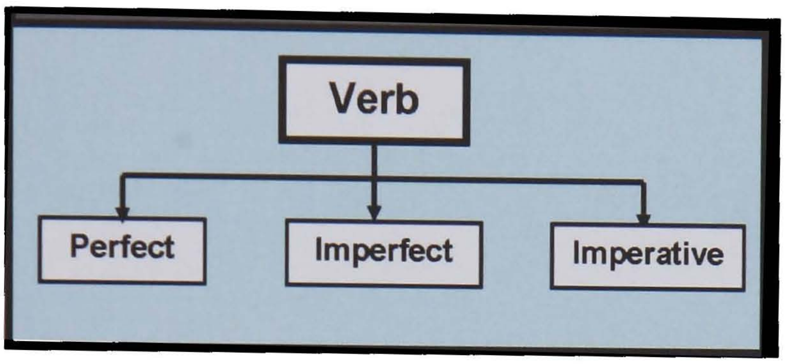


Figure 4.1: Categories of Arabic verb

- The perfect (past) known in Arabic as *Almadi.*
- The imperfect (present) known as *AlmDArE.*
- The imperative (future) known as *AlAmr.*

Practically all semitic scholars agree that the tense of the verb does not express the idea of time, but rather the idea of "finished act" and an "unfinished act". If the act is incomplete or unfinished, the verb is the imperfect. However, the Arab looks at these tenses as expressing the idea of time and not the idea of finished or unfinished acts. In Arabic, to form the imperative verb, a knowledge of the imperfect verb is necessary, because the imperative verb is a form of the imperfect [61].

2. **Noun**

A noun in Arabic indicates a meaning by itself without being connected with the notion of time and refers to a person, place, thing and event [96].

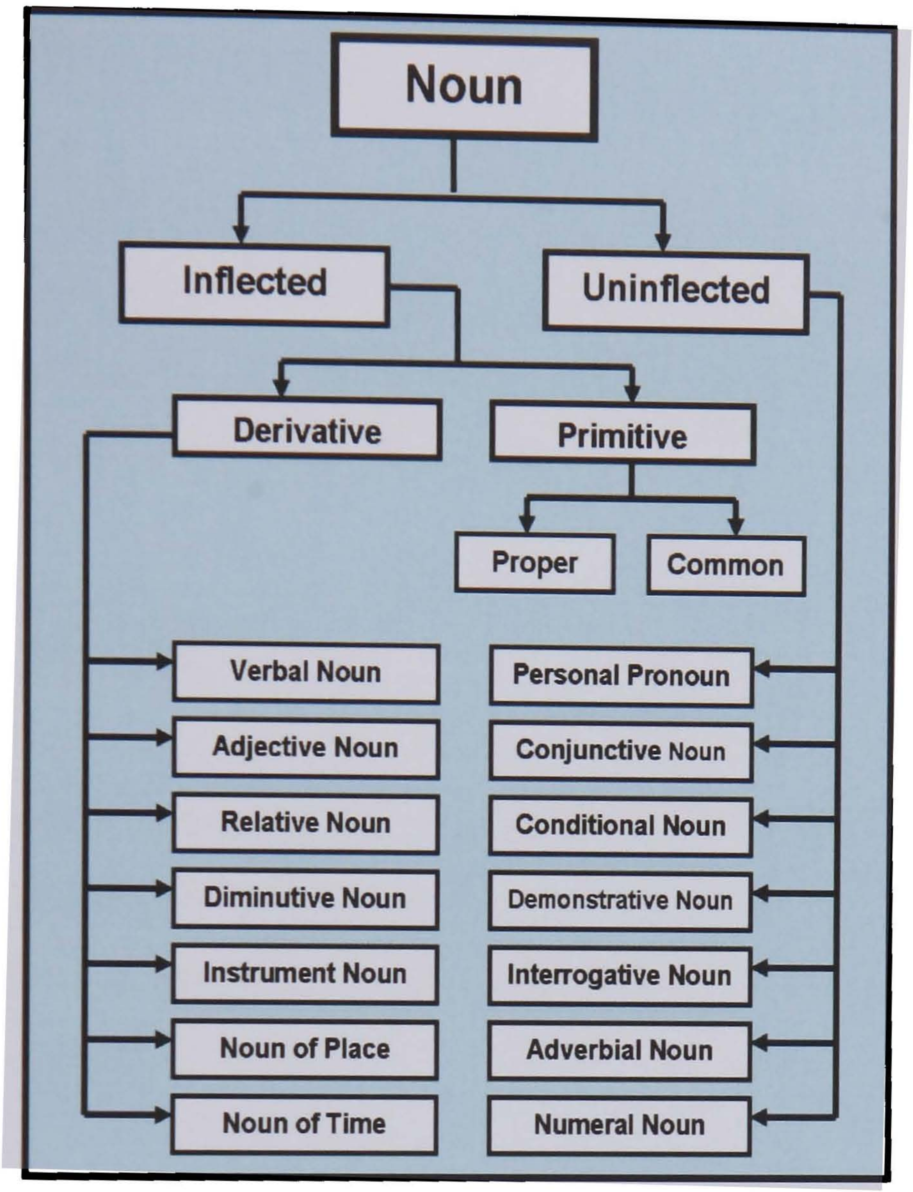


Figure 4.2: Categories of Arabic noun

Grammatically, nouns in Arabic are of two kinds: *inflected nouns*, those nouns that are affected with the inflectional features, such as, Adjecive, Verbal, Relative, etc., and *uninfected*, those nouns appear always in one case and can't affected with the inflectional features, such as, personal, conjunctive, conditional, etc.

The inflected nouns come also in two kinds : *primitive* (not derived from verb or noun) such as رجلٌ, *rjlun*, "a man", أسدٌ, *Osdun*, " a lion", and *derivative* (derived from verb or noun) such as مكتبةٌ, *mktbpun*, "library", derived from the verb كتبَ [61].

In Arabic, nouns can be categorised into the following sub-classes : (*Common, Proper, Verbal, Relative, Noun of time, Adjective, Diminutive, Instrument, Noun of place, Conjunctive, Interrogative, Pronoun, Adverbial, Numeral, Demonstrative, Conditional*). The list below summarises these sub-classes in a little more detail :

- **Common Noun**

  The vast sub-class of the main class (noun) in Arabic is common noun. These nouns may or may not be derived from the ground verb (root). Common nouns may include the definite article ال to indicate definiteness or may not [120]. For example, the words الشجرةَ, *Alshjrpa*, “the tree” and شجرةٌ, *shjrpun*, “a tree”.

- **Proper Noun**

  Like English language, Arabic proper nouns include names of people, places, names of cities, countries, and geographical features. These nouns come from a variety of sources, many of them are Arabic words, but some are non-Arabic (foreign words). These nouns may include the definite article ال or may not [120]. For example, لندن, *lndn* “London”, القَاهرةُ, *AlqAhrpu*, “cairo”.

- **Verbal (Infinitive) Noun**

  The verbal nouns are derived from verb forms[4]. They follow a regular

[4] verb forms described in Chapter 5, Section 5.3

pattern. For example, the words تدريسٌ, *tdrysun*, "instruction", تسامحٌ, *tsAmHun*, "tolerance", معتقدٌ, *mEtqdun*, "belief", follow the patterns تفعيلٌ, *tfEylun*, تفاعلٌ, *tfAElun*, مفتعلٌ, *mftElun* respectively [61].

- **Relative Noun**

  Relative nouns are formed from other nouns by adding the suffix يٌّ (for masculine) or يّةٌ (for feminine) [61]. For example, the words شكليٌّ, *shklyun*, "formal" and اردنيّةٌ, *Ardnypun*, "Jordanian (fem)".

- **Noun of Time**

  In Arabic, to denote the noun of time, some patterns refer to the time when the activity specified by the verb occurs or has been used [57]. For example, the word موعدٌ, *mwEdun*, "appointment" follows the patterns مفعلٌ, *mfElun*.

- **Adjective Noun**

  An adjective in Arabic is placed after the noun it qualifies, and in most cases agrees with it in number and gender. On the other hand, the present participle and past participle are used as adjectives in Arabic language [61]. For example, the words متكبّرٌ, *mtk=run*, "haughty" and معظّمٌ, *mED=mun*, "glorified". Adjective words like many other words in Arabic are derived from the ground verb and each adjective word follows a certain pattern. For example, the words صالحٌ, *SAlHun*, "good man", follows the pattern فاعلٌ, *fAElun*.

- **Diminutive Noun**

  Arabic has a few diminutive forms of nouns which are actually used. They are formed from trilateral noun (noun with three consonant). For example, the word جُبيلٌ, *jbylun*, "a little mound" follows the pattern فُعيلٌ [61].

- **Instrument Nouns**

  Nouns of instrument in Arabic are of two kinds : those which are derived from ground verb (root) such as the word مفتاحٌ, *mftAHun*, " a key", follows the pattern مفعالٌ and derived from the verb فتح, *ftHa*, "he opened", and those which are not derived from ground verb such as سكينٌ, *skynun*, "knife" or جرسٌ, *jrsun*, "bell" [61].

- **Interrogative Noun**

  Usually, the interrogative words (question words) are used at the beginning of an Arabic sentence [76]. For example, the words, كيفَ, *kyfa*, أينَ, *Ayna*, متى, *mtY*, ماذا, *mAdhA* and كم, *km*, which equivalent to the words, "how ?", "where ?", "when ?", "what ?", "how (many/much) ?" in English respectively.

- **Pronoun**

  Pronoun sub-class on our tag set represents the personal pronouns[5]. They refer to persons or entities. On the other hand, the pronoun class in Arabic may come as separate words-independent (subject) or take the form of

[5] the demonstrative, conjunctive and interrogative nouns in some Arabic grammar dictionaries fall also under the pronoun class. However, in our tag set, each sub-class has a different tag to distinguish between those words which belong to these classes in more precises (see [57]).

suffixes (object and possessive pronouns). An example of separate words, the words أنا, *AnA*, أنتَ, *Anta*, نحن, *nhnu*, هو, *hwa*, هي, *hya* and هم, *hm* which equivalent to the words, "I", "you", "we", "he", "she" and "they" in English respectively. In contrast, English has a fewer number of classes of personal nouns than Arabic, because the personal pronouns in Arabic show more difference in inflectional features, such as, gender, number and person [120]. Table 4.1 shows the difference in the gender and number of persons between Arabic and English language. This table shows that for the Arabic first person there is no gender distinction. For the second person, there are five forms of "You". For the third person, there are six verbal distinctions and five pronoun distinctions. Thus, the total number of personal pronouns in Arabic is twelve, as opposed to the eight of English.

	English	Arabic
First Person	I, We	أنا, *"AnA"* , نحن, *"nHn"*
Second Person	You(Fe), You(Ma)	أنتَ, *"Anta"* (Ma/Sn), أنتِ, *"Anti"* (Fe/Sn) أنتما, *"AntmA"* (Du) أنتم, *"Antm"* (Ma/Pl) أنتن, *"Antn"* (Fe/Pl)
Third Person	He She It They	هو, *"hwa"* (Ma/Sn) هي, *"hy"* (Fe/Sn) هما, *"hmA"* ((Ma/Fe)/Du) هم, *"hm"* (Ma/Pl) هن, *"hn"* (Fe/Pl)

Table 4.1: Personal pronouns between Arabic and English

- **Adverbial Noun**

  Grammatically, adverbs may belong to noun class or particle class. Usually, most adverbs in Arabic are words used to answer the questions "when ?", "where ?" and "how ?" such as the words امسِ, *Amsi*, "yestarday", شرقَاً, *shrqAan*, "eastward" and ضَاحكَاً, *DAHkAan*, "in laughing". On the other hand, some adverbs are used as particles such as the words تحتَ, *tHta*, "under", فوقَ, *fwqa*, "over" and قبلَ, *qbla*, "before". However, in our tag set we have used one tag to represent the adverbial words which fall in our tag set under the noun sub-classes [61].

- **Demonstrative Noun**

  The demonstrative words in Arabic are determiners used with other nouns or sometimes instead of nouns to show either distance from or proximity to the speaker. For example, the words هذَا, *hdhA*, ذلكَ, *dhlka*, هؤلَاءِ, *hWlA'*, أولئكَ, *Awl'ka* are equivalent to "this", "that", "these", "those" in English respectively. Arabic has a richer variety of demonstrative words which inflect for gender, number and case [120]. However, the demonstrative words in Arabic do not have a pattern to follow.

- **Conditional Noun**

  In Arabic, the conditional noun is used between two sentences to show that the second sentence depends on the first sentence [57]. For instance, the words, مهمَا, *mhma*, كلمَا, *klma*, لمَّا, *lmma* which either mean "whatever", "whenever", "when", respectively.

- **Noun of place**

  Arabic language has a specifically derived patterns which are used to denote the noun of place. These patterns refer to the place where the activity specified by the verb occurs [120]. For example, the words مركز, *mrkza*, "center", مدرسة, *mdrspa*, "school" follow the patterns مفعل, *mfEla*, مفعلة, *mfElta* respectively.

- **Conjunctive Noun**

  The conjunctive words in Arabic relate to an element in a subordinate relative clause to a noun or a noun phrase in the main clause of the sentence. They may be definite or indefinite. In addition, they marked for gender and number [120]. For example, the words ألذي, من, ما, أي which are equivalent respectively to the words, "who, which", "who, whoever", "that which, whatever", "(he) who, whoever" in English.

- **Numeral Noun**

  Arabic has a complex numeral system. It is one of the complicated features of written Arabic. Numeral nouns in Arabic are of two types : *ordinal numbers*, they usually follow the noun that they modify and agree with it in gender, but sometimes precede it. For example, المؤتمر الثاني, *AlmWtmr Althani*, "the second conference" and عشرون يوما, *Eshrwna ywmA*, "twenty day". The second type refer to *cardinal numbers*; these numbers are rather difficult to categorise due to some characteristics of Arabic language [120]. For example, اثنان, *Athnani*, "two", احدى المدن, *AhdY Almdn*, "one of the cities". Numeral noun also inflect for gender, number and case [76].

3. **Particle**

   Particles are of two kinds : *Formation* and *Signification* as shown in figure 4.3. They are one of the three main POS classes in the Arabic language. *Formation* particles are particles which constitute the characters of Arabic word, while *signification* particles are used with verbs and nouns; they are effective to signal the mood of verb or the case of noun [120]. For example, the particles " لم ", *lm*, "never", " كي" , *ky*, "in order" indicate Jussive and Subjunctive respectively.

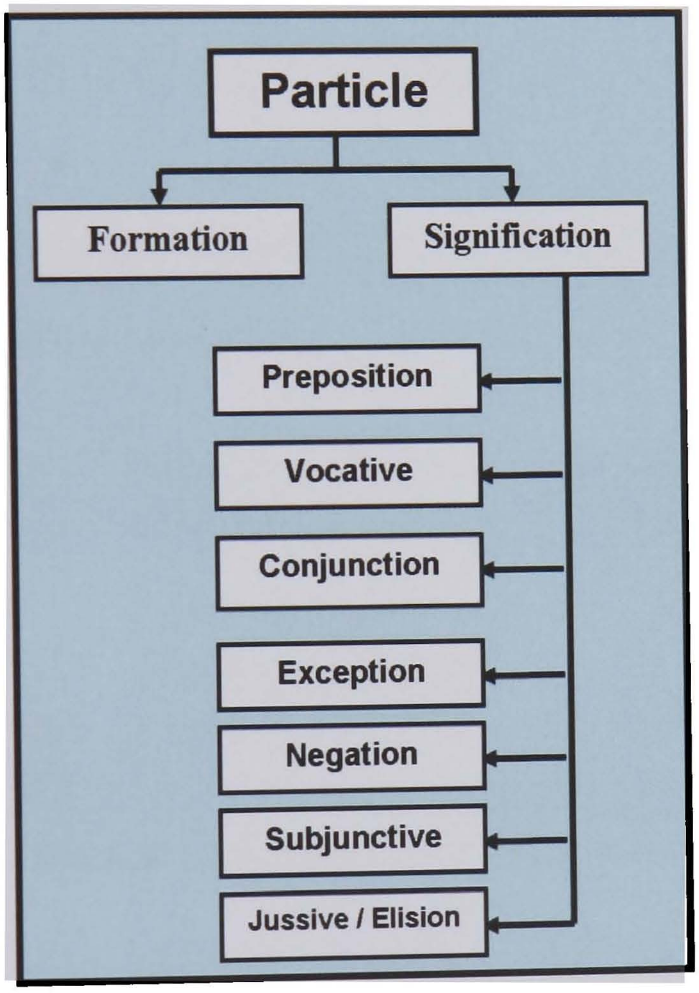


Figure 4.3: Categories of Arabic particle

4.3.2 Tag design of ARBTAGS

ARBTAGS tags have been built based on the following main formula:

[T , S , G , N , P , M , C , F] , Where:

T, represents the name of each main POS class in Arabic. On the other hand, throughout this section, abbreviation symbols representing the name of each main POS class and sub-class as well as the possible value of the inflectional features which have been used to represent the tag in our tag set are shown between square brackets in each table. Table 4.2 shows the abbreviation symbols of the main POS classes in Arabic.

Verb [Ve]	Noun [Nu]	Particle [Pr]

Table 4.2: Abbreviation symbols of the main POS classes

S, represents the sub-classes of each main POS class in Arabic. The abbreviation symbols of sub-classes of verb, noun, and particle class are shown in Tables 4.3, 4.4 and 4.5 respectively.

Perfect [Pe]	Imperfect [Pi]	Imperative [Pm]

Table 4.3: Abbreviation symbols of the sub-classes of class verb

Proper [Po]	Common [Cn]	Adjective [Aj]
Verbal(Infinitive) [If]	Relative [Re]	Diminutive [Dm]
Instrument [Is]	Noun of Place [Pn]	Noun of Time [Tn]
Pronoun [Ps]	Conjunctive [Cv]	Conditional [Cd]
Demonstrative [De]	Interrogative [In]	Adverb [Ad]
Numeral noun [Nn]		

Table 4.4: Abbreviation symbols of the sub-classes of class noun

Preposition [Pp]	Vocative [Vo]	Exception [Ex]
Conjunction [Co]	Negation [An]	Subjunctive [Sb]
Jussive/Elision [Jv]		

Table 4.5: Abbreviation symbols the sub-classes of class particle

G, represents the inflectional feature (**Gender**), used to inflect noun and verb sub-classes. The possible values for the inflectional feature *gender* are shown in Table 4.6.

Masculine [Ma]	Feminine [Fe]	Neuter [Ne]

Table 4.6: The possible value of the inflectional feature (Gender)

N, represents the inflectional feature (**Number**), used to inflect noun and verb sub-classes. The possible values of the inflectional feature *number* can be seen in Table 4.7.

Singular [Sn]	Dual [Du]	Plural [Pl]

Table 4.7: The possible value of the inflectional feature (Number).

P, represents the inflectional feature (**Person**), used to inflect the verb sub-classes. Table 4.8 shows the possible values of the inflectional feature *person*.

First [Fs]	Second [Sc]	Third [Th]

Table 4.8: The possible value of the inflectional feature (Person).

M, represents the inflectional feature (**Mood**), used to inflect the verb sub-classes. Table 4.9 shows the possible value of the inflectional feature *mood*.

Indicative [Dc]	Subjunctive [Sj]	Jussive [Js]

Table 4.9: The possible value of the inflectional feature (Mood).

C, represents the inflectional feature (**Case**) and it is used to inflect the noun sub-classes. The possible value of the inflectional feature case can be seen in Table 4.10.

Nominative [Nm]	Accusative [Ac]	Genitive [Ge]

Table 4.10: The possible value of the inflectional feature (Case).

F, represents the inflectional feature (**State**) and it is used to inflect the noun sub-classes. Table 4.11 shows the possible value of the inflectional feature *state*.

Definite [Df]	Indefinite [Id]

Table 4.11: The possible value of the inflectional feature (State).

However, in Arabic, the first two main POS classes, *verb* and *noun*, can inflect grammatically in the system of inflectional morphology, while the third one (*particle*) can not [16]. For example, verb can inflect for person, number, gender and mood as shown in figure 4.4, while the inflectional features for the noun class can be seen in figure 4.5.

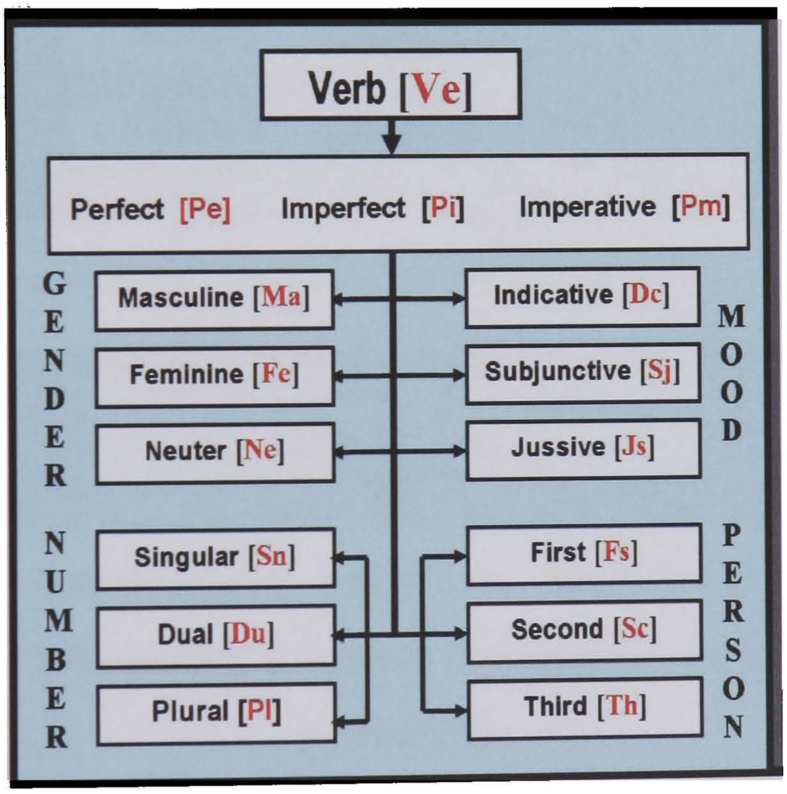


Figure 4.4: Verb sub-classes and their inflectional features

4.3.3 Detailed and general tags in ARBTAGS tag set

Before describe the detailed and general tags which have been used in ARBTAGS tag set, let us summarise all the abbreviation symbols which have been used in the

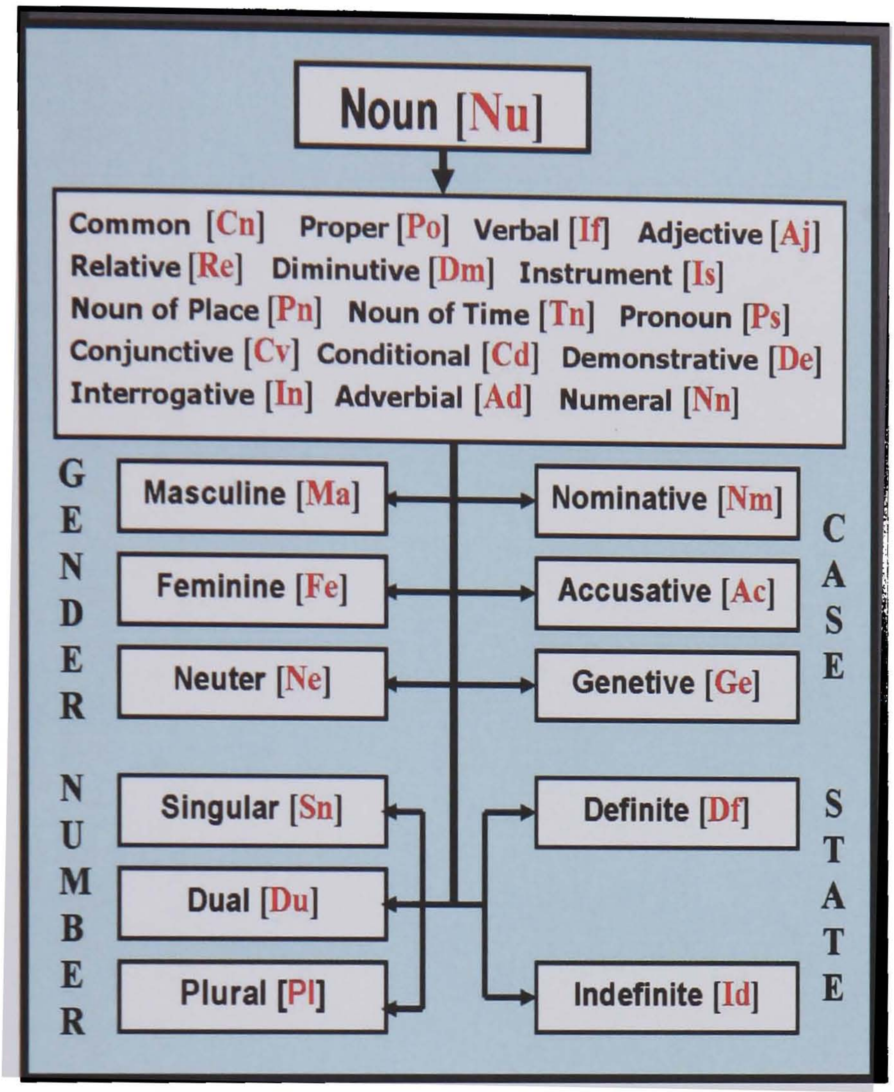

Noun [Nu]
Common [Cn] Proper [Po] Verbal [If] Adjective [Aj]
Relative [Re] Diminutive [Dm] Instrument [Is]
Noun of Place [Pn] Noun of Time [Tn] Pronoun [Ps]
Conjunctive [Cv] Conditional [Cd] Demonstrative [De]
Interrogative [In] Adverbial [Ad] Numeral [Nn]
GENDER
Masculine [Ma]
Feminine [Fe]
Neuter [Ne]
CASE
Nominative [Nm]
Accusative [Ac]
Genetive [Ge]
NUMBER
Singular [Sn]
Dual [Du]
Plural [Pl]
STATE
Definite [Df]
Indefinite [Id]

Figure 4.5: Noun sub-classes and their inflectional features

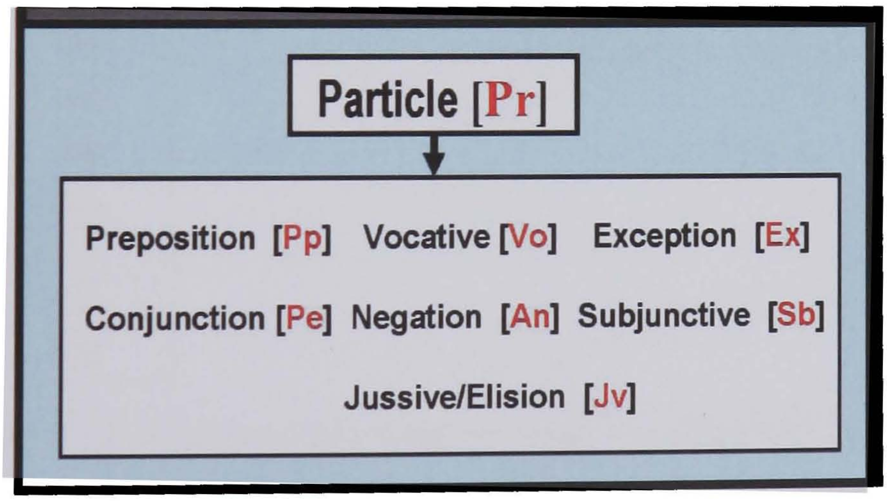

Particle [Pr]
Preposition [Pp] Vocative [Vo] Exception [Ex]
Conjunction [Pe] Negation [An] Subjunctive [Sb]
Jussive/Elision [Jv]

Figure 4.6: Particle sub-classes

developed tag set. These symbols can be seen in Table 4.12.

As mentioned above, ARBTAGS has 28 general tags and 161 detailed tags. The

Category	Abb
Verb	Ve
Noun	Nu
Particle	Pr
Perfect	Pe
Imperfect	Pi
Imperative	Pm
Adjective	Aj
Verbal	If
Noun of Place	Pn
Noun of Time	Tn
Demonstrative	De
Relative	Re
Pronoun	Ps
Diminutive	Dm
Instrument	Is
Proper	Po
Adverb	Ad
Common	Cn
Interrogative	In
Conjunctive	Cv
Conditional	Cd
Numeral	Nn
Preposition	Pp
Vocative	Vo
Exception	Ex
Negation	An
Subjunctive	Sb
Jussive/Elision	Jv
Conjunction	Co
Foreign word	Fw

Inflectional feature value	Abb
Masculine	Ma
Feminine	Fe
Neuter	Ne
Singular	Sn
Plural	Pl
Dual	Du
First	Fs
Second	Sc
Third	Th
Indicative	Dc
Subjunctive	Sj
Jussive	Js
Nominative	Nm
Accusative	Ac
Genitive	Ge
Definite	Df
Indefinite	Id

Table 4.12: Abbreviation symbols used in ARBTAGS tag set

detailed tags not only represent the name of the class that the word belong to, but also represent the inflectional features of this word.

The rational behind developing detailed tags comes from two reasons. The first reason is to enrich each word in the testing corpus with more linguistic information for the word including the inflectional feature of the word. The tagged corpus becomes more useful for linguists and NLP developers if most words are tagged with detailed tags.

The second reason is that the pattern in the pattern-based technique represents the template for the whole word. It not only includes the form of the word but also includes the prefixes and suffixes attached to the word. The suffixes provide the inflectional feature of the word. Since this pattern is generated automatically from three lexicons; prefixes with its tags, forms with its tags and suffixes with its tags, the generated tag with each pattern is a detailed tag. On the other hand, general tags are also used by applying the lexical and contextual rules as a second technique in this work.

As an example of a detailed tag, the word يشاهدونَ, *yshAhdwna*, "they watching" has the following detailed tag **VePiMaPlThDc**, which means *[Imperative verb, masculine gender, plural number, third person, subjunctive mood]*. While the general tag **NuPo** may assign to the word such as رمزي, *Rmzy* which means *[Proper noun]*.

POS tag may be very coarse (e.g **Ve** *"Verb"*) or very fine (e.g **VePiMaPlFsJs** *" Verb, Imperfect, Masculine, Plural, First Person, Subjunctive "*), depending on the task or application [114]. Since the main aim of AMT system is to produce a tagged corpus, the tags were developed with a good level of granularity, where each tag is enriched with inflectional features that meets the need of linguists and NLP developers. On the other hand, the cardinality of the POS tag set makes the tagging between a morphologically ambiguous inflective language, e.g, Arabic and a language with poor inflection such as English is different [78]. For example, the number of tags for perfect verbs between the ARBTAGS tag set presented in this work and the Penn Treebank tag set for English is shown in Table 4.13. The numbers 6 vs. 81 shown in table 4.13 illustrate the differences very clearly.

	Penn Treebank tag set (English)	Arabic tag set (ARBTAGS)
verbs	VB, VBD, VBG, VBN, VBP, VBZ	For Perfect verb only [VePe] : [MaFeNe][SnDuPl][FsScTh][DcSjJs]
	6	3 X 3 X 3 X 3 = 81

Table 4.13: ARBTAGS tag set vs. Penn Treebank tag set

ARBTAGS tag set general tags are shown in Table 4.14, while a sample of detailed tags can be seen in Table 4.15. However, the general and detailed tags with examples have been described in full in Appendix A.1 an Appendix A.2.

Tag	Dsecription
VePe	*Perfect verb*
VePi	*Imperfect verb*
VePm	*Imperative verb*
NuPo	*Proper noun*
NuCn	*Common noun*
NuAj	*Adjective noun*
NuIf	*Verbal noun*
NuRe	*Relative noun*
NuDm	*Diminutive noun*
NuIs	*Instrument noun*
NuPn	*noun of Place*
NuTn	*noun of Time*
NuPs	*Pronoun*
NuCv	*Conjunctive noun*

Tag	Description
NuCd	*Conditional noun*
NuDe	*Demonstrative noun*
NuIn	*Interrogative noun*
NuAd	*Adverbial noun*
NuNn	*Numeral noun*
Pun	*Punctuation mark*
PrPp	*Preposition*
PrVo	*Vocative Particle*
PrCo	*Conjunction Particle*
PrEx	*Exception Particle*
PrAn	*Negation Particle*
PrSb	*Subjunctive Particle*
PrJv	*Jussive Particle*
Fw	*Foreign word*

Table 4.14: ARBTAGS general tags

Tag	Description
VePeMaSnThSj	*Verb, Perfect, Masculine, Singular, Third Person, Subjunctive*
VePeMaSnFsDc	*Verb, Perfect, Masculine, Singular, First Person, Indicative*
VePeMaSnSeSj	*Verb, Perfect, Masculine, Singular, First Person, Subjunctive*
VePeFeSnSeJs	*Verb, Perfect, Feminine, Singular, Second Person, Jussive*
VePeFeSnThJs	*Verb, Perfect, Feminine, Singular, Third Person, Jussive*
VePiMaPlFsJs	*Verb, Imperfect, Masculine, Plural, First Person, Subjunctive*
VePiMaPlFsDc	*Verb, Imperfect, Masculine, Plural, First Person, Indicative*
VePmMaSnSeJs	*Verb, Imperative, Masculine, Singular, Second Person, Jussive*
VePmFeSnSeJs	*Verb, Imperative, Feminine, Singular, Second Person, Jussive*
NuDeSnAcId	*Demonstrative Noun, Singular, Accusative,Indefinite*
NuDeDuGeId	*Demonstrative Noun, Dual, Genitive, Indefinite*
NuInId	*Interrogative Noun, Indefinite*
NuCvSnId	*Conjunctive Noun, Singular, Indefinite*
NuAdId	*Adverbial Noun, Indefinite*
NuNmId	*Numeral Noun, Indefinite*
NuAjMsSnNmId	*Adjective Noun, Masculine, Singular, Nominative, Indefinite*
NuAjMsSnNmDf	*Adjective Noun, Masculine, Singular, Nominative, Definite*
NuAjMsSnAcDf	*Adjective Noun, Masculine, Singular, Accusative, Definite*
NuAjMsSnGeDf	*Adjective Noun, Masculine, Singular, Genitive, Definite*
NuIsMaDuGeId	*Instrument Noun, Masculine, Dual, Genitive, Indefinite*
NuDmSnNmId	*Diminutive Noun, Singular, Nominative, Indefinite*
NuReMaSnNmId	*Relative Noun, Masculine, Singular, Nominative, Indefinite*
NuReMaDuGeDf	*Relative Noun, Masculine, Dual, Genitive, Definite*
NuCnMaSnNmId	*Common Noun, Masculine, Singular, Nominative, Indefinite*
NuCnFeSnNmId	*Common Noun, Feminine, Singular, Nominative, Indefinite*
NuCnMaPlGeDf	*Common Noun, Masculine, Plural, Genitive, Definite*
NuPsMaSnThAc	*Personal Noun, Masculine, Singular, Third Person, Accusative*

Table 4.15: Sample of detailed tags in ARBTAGS

4.4 Chapter Summary

This chapter presented a number of criteria to take into account while developing the POS tag set. Arabic inflectional features, such as, gender, number, case, mood, person and state are described. In this chapter, we described the steps of our tag set design. An Arabic tag set called ARBTAGS contains 161 detailed tags and 28 general tags covering an Arabic main POS classes and sub-classes which have been compiled and introduced in this work. The developed tag set follows the Arabic grammatical system, based upon POS classes and inflectional morphology that Arab grammarians describe. The developed tag set differs from the tag sets which have been built for Arabic. The main difference is a tag set hierarchy be introduced and compiled in this chapter. Since the main aim of AMT system is to produce a tagged corpus, the tags were developed with a good level of granularity, where each tag is enriched with inflectional features that meets the need of linguists and NLP developers.

Chapter 5

Design and Implementation of AMT

Objectives

- To define the characteristics of AMT tagger.
- To define the proposed approach.
- To present a description of the tagger system.
- To describe the tagging process.

5.1 AMT Characteristics

The tagger system (Arabic Morphosyntactic Tagger (AMT)) presented in this work has the following characteristics :

- **Lexicon Free**

  AMT did not require a manually tagged or untagged lexicon which contains Arabic words. It requires the testing corpus only. Building a generic POS tagger system without a lexicon depends on the language and the characteristics of its grammar, both the morphological and the syntactical systems of that language.

- **Word Level Tagging**

  It is possible for the tagger system presented in this work to tag one word regardless of the context. This possibility comes from (1) the fact that the word in the testing corpus has a diacritical mark. The diacritical mark provides a semantic information and defines the inflectional features of the word, which help to resolve the lexical ambiguity may arise. (2) the main technique used in this work is based on the pattern of the word instead of the word itself. Since the Arabic word matches its correct pattern, the correct tag assigned to the word regardless of the context in most cases as described in the next section.

5.2 Rule-based - the developed approach

The approach here is a rule-based. It is based on incorporating a set of linguistic rules to assign the correct tag to each word in the testing corpus. Two different techniques were used in this work; the *pattern-based technique* and the *lexical and contextual technique*. The rules in the former technique are based on the pattern of the testing word. While the rules in the later technique are based on the character(s), affixes, the last diacritical mark, the word itself, and the surrounding words or on the tags of the surrounding words.

The basic idea of the *pattern-based technique* is to generate automatically a lexicon of patterns instead of using manually tagged or untagged Arabic words lexicon for training. Section 5.3 describes the *pattern-based technique* in more detail. The *lexical and contextual technique* is used to assist the main technique to assign the correct tag to those words not tagged by the *pattern-based technique*. Section 5.4 describes the *lexical and contextual technique* in more detail.

As mentioned in chapter 3, Arabic has a set of rules or signs described by Arab grammarians for more than 1400 years, such as, rules used to distinguish nouns from verbs and particles. It has set of facts and characteristics, such as, each original Arabic word has a pattern and many Arabic words follow only one pattern. Additionally, the diacritic is important feature (chapter 3). All of these facts and characteristics are taken into account when the above techniques are built and used in this work.

5.2.1 Justification for using the rule-based approach

The AMT system presented in this work is designed to accept any partially-vocalised Arabic text as an input and produce a tagged text. The signs that indicate the category of the word in Arabic language on the one hand, and the existence of diacritic feature on the other hand play a great role in reducing the lexical ambiguity of the words and providing a semantical information to the word leading to assigning the correct tag for each word in the testing corpus. In addition, due to the fact that semitic languages in general have a morphological system based on a root and pattern structure, using the pattern of the word instead of the word itself can achieve a good result in assigning the correct tag to each word in the testing corpus.

On the other hand, statistical approach as the second main approach in POS tagging requires a huge manually tagged lexicon to calculate the statistical information such as the probability of the particular word and tag co-occurring [73]. This approach may be useful in case we are dealing with an unvocalised Arabic text because with the missing of the diacritical mark in this type of text, the word may has multiple POS tags. But to achieve a remarkable accuracy using statistical approach, the manually tagged corpus used for training should be very huge. Unlike English, Arabic still lacks a huge manually tagged corpus from which large amounts of training data can be extracted. For example, a training corpus with about 10,000 words which is used by Khoja [87] in her tagger for Arabic, is definitely not sufficient to cover most words in Arabic language. In addition, the small training corpus used in a statistical approach presents the problem of unknown words.

Unknown words are words not appearing in the training corpus. Neither the testing corpus nor the training corpus have a lexical information and tags for these words. The statistical model in this case has no role in dealing with unknown words. So, if the training corpus is very small and most words in testing corpus may be completely different from the training corpus, the accuracy of the POS tagger in this case becomes very weak.

At the same time, many POS tagger systems have been built for English based on statistical approach and achieved very high accuracy. The reason behind achieving this remarkable accuracy is the very huge lexicon which contains hundred of millions of words that have been used in these systems. However, as mentioned above, AMT system presented in this word did not used a lexicon for training. Thus, the rule-based approach is the best approach to achieve the above goal due to the fact that the testing corpus in this work is a partially-vocalised Arabic text.

5.3 Pattern-based technique - A novel technique

Many computational work on Semitic languages assumed that a word may consist of the following elements: Prefixes, Stem and Suffixes [45,84,110,118,119]. Arabic language has a trilateral and quadrilateral verb form. The great majority of Arabic verbs are trilateral that contain three letters, the first letter is ف , *f*, the second is ع , *E*, while the third letter is ل , *l*. The Arab grammarians have used the trilateral verb form فَعَلَ, *fEla*, "do" as paradigm (called ground form) to discuss word formation.

The ground form of the trilateral and the form of the quadrilateral[1] verbs have derived a great number of other forms by inserting a long vowel, lengthening the root medial letter, and/or adding consonantal prefixes to produce a new word with new meaning that still shares the basic meaning of the root[2] [138] [81]. For example, the words لعب, *lEba*, لاعِبٌ, *lAEbun* which mean " he played", "player" respectively. The former word represents the root and belongs to the verb class which has the basic template form فَعَلَ. When adding the long vowel consonant " ا ", "*Alif*" to the medial letter of the root, a new word لاعِبٌ belonging to the noun (adjective noun) class has been produced, which has the derived form فَاعِلٌ, *fAElun* and still shares the basic meaning of the root. The ground form and other forms derived from the ground form are shown in Table 5.1. However, these derived forms express various modifications of the idea conveyed by the ground form.

As Arab grammarians described, each original Arabic word has a pattern. M.Elaffendi defined the morphological pattern as :

"***a template that shows how the word should be decomposed into its constituent morphemes (prefix + stem + suffix), and at the same time, marks the positions of the radicals comprising the root of the word***" [107].

It is important to point out here that the pattern is different from the word; it has

[1] The quadrilateral verb form is فعلل fEll, by doubling the third letter of the ground form. In Arabic, these verbs are rare.

[2] In English language, the produced words are which are termed (*stems*).

Form no	Derived form	Transliteration	Modifications of ground form
1	فعلَ	fEla	The ground form (No Modification)
2	فعّلَ	fEEla	Doubling the second letter
3	فَاعلَ	fAEla	Infixing the letter ا
4	افعلَ	AfEla	Prefixing the letter ا
5	تفعلَ	tfEla	Prefixing the letter ت
6	تفَاعلَ	tfAEla	Prefixing the letter ت and infixing the letter ا
7	انفعلَ	AnfEla	Prefixing the letters ا and ن
8	افتعلَ	AftEla	Prefixing the letters ا and infixing the letter ت
9	افعلَّ	AfElla	Prefixing the letters ا and doubling the third letter
10	استفعلَ	AstfEla	Prefixing the letters است

Table 5.1: Derived forms from the ground form (root)

no meaning itself, but its a template that indicates the positions of the root letters. The pattern represents the lexical category of the word and indicates the syntactic and semantic roles [107].

In this work the word "*pattern*" is used to represent the template of the whole word including the prefixes, form (root+infixes) and suffixes, which are attached to the word. The pattern in Arabic shares the word on the affixes may be added to the ground form (root). For example, the word ويصَافحونَ, *wySAfhHwna*, "to shake hands" has the pattern ويفَاعلونَ, "wyfAElwna" as shown in figure 5.1. The root of the word ويصَافحونَ is صفح, *SfH* which has the form فعل, *fEl*, while the whole pattern is ويفَاعلونَ, *wyfAElwna.*

The existence of the last diacritical mark in both the pattern and the word is very im-

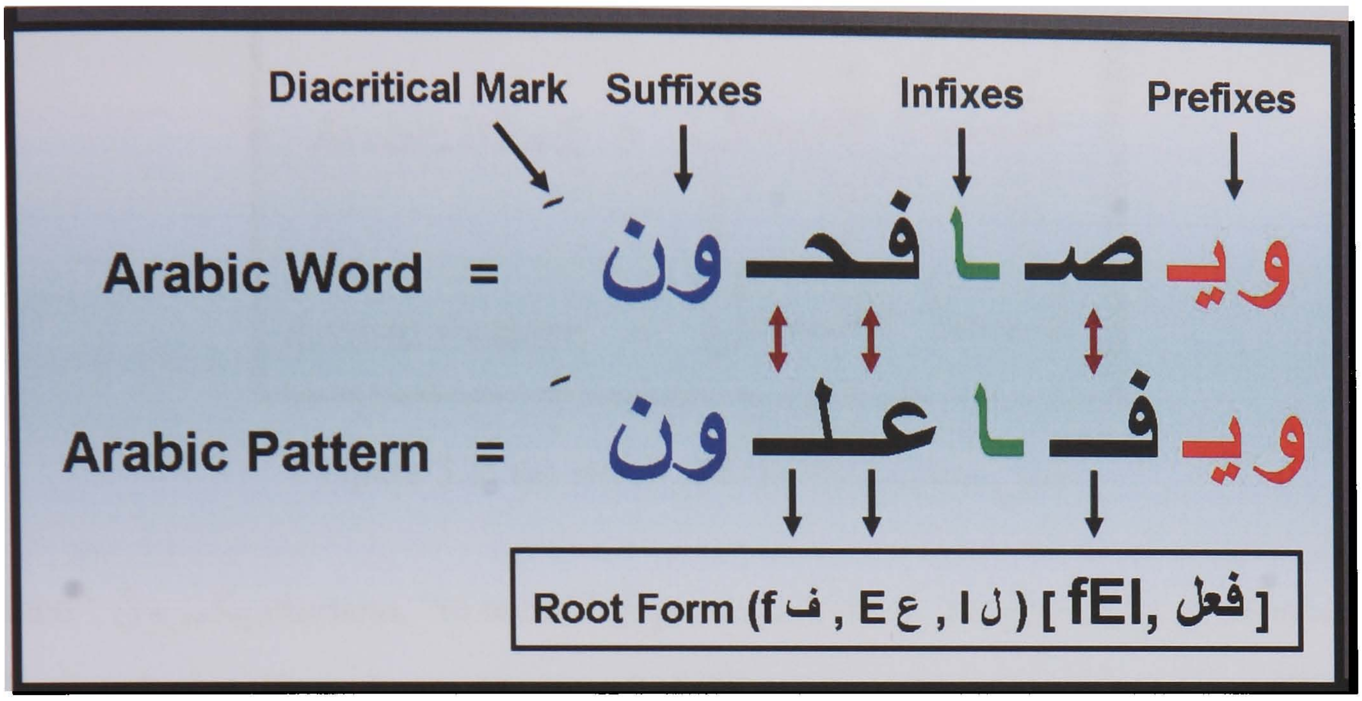


Figure 5.1: the word ويصَافحونَ and its pattern ويفَاعلونَ

portant. Without it, it becomes very difficult in most cases to determine the lexical category and to define the inflectional features of the word. For example, the word غَافل, *ghAfl* has the pattern فَاعل as shown in figure 5.2, but the word still has an ambiguity regarding its lexical category and semantic meaning due to missing the last diacritical mark in both the pattern and the word. It may be غَافلَ, *ghAfla* if the last diacritical mark is fatha mark, in this case, it means "take advantage of someone's inattention" and it belongs to the verb class or غَافلٌ, *ghAflun* if the last diacritical mark is nunation mark (tanween damm), which means "inattentive" and belongs to noun class. Thus, while the last diacritical mark is missing in the pattern as well as the word, the lexical ambiguity remaining apparent.

In the Arabic language, there is no word has more than one pattern to follow. At the same time, you may find hundreds of Arabic words may follow one pattern. For example, the words, "يشربونَ", *yshrbwna*, "to drink", يسمعونَ, *ysmEwna*, "to hear", يضربونَ, *yDrbwna*, "to beat", يكتبونَ *yktbwna*, "to write", يفهمونَ *yfhmwna*, "to under-

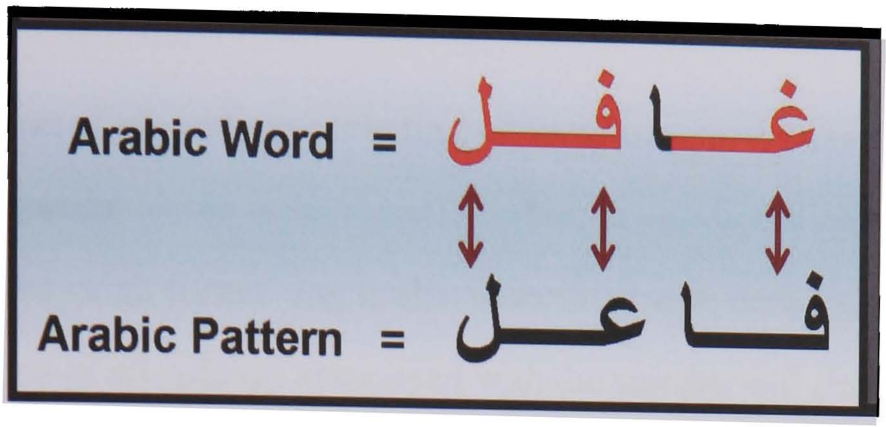


Figure 5.2: the word غَافل and its pattern فَاعل

stand", يكسرونَ *yksrwna*, "to break", يمسحونَ *ymsHwna* "to wipe", يحملونَ *yHmlwna*, "to carry". يقفزونَ *yqfzwna*, "to jump", follow the same pattern "يفعلونَ", *yfElwna*. More than 500 words other these words follow the above pattern. All the above words belong to the imperfect verb class. Another example, all of the Arabic words with three consonants, end with fatha mark and follow the pattern فعلَ, fEla, "do", are perfect verb words.

The case is also valid for the words belonging to noun class. For example, all the Arabic words following the pattern فَاعلٌ, fAElun, such as, قَاتلٌ, *qAtlun*, "killer", سَاحرٌ, *sAhrun*, "magician" and كَاتبٌ, *kAtbun*, "writer", can be categorised as Adjective nouns. The above examples show that the last diacritical mark plays a great role in determining the correct tag and adding a semantical information to the word. In addition, using the pattern of the word means that building a pattern lexicon with 100 entries may cover 15,000 words, which constitute the main advantage of the *pattern-based technique.*

5.3.1 Pattern-based Rules

Since a lexicon of Arabic words or training corpus in this system is not required, instead, we generated a lexicon of patterns which are associated with the last diacritical mark and generated automatically by combining:

1- A single lexicon of all prefixes including all valid concatenations. Tag is also associated with each prefix.

2- A single lexicon of all forms. Tag is also associated with each Form.

3- A single lexicon of all suffixes associated with the suitable last diacritical mark. Tag is also associated with each suffix.

Table 5.2 shows a simple part of the prefix, form and suffix lexicons for some imperfect verb words. The combined pattern lexicon is shown in Table 5.3.

Prefixes	Tag	Forms	Tag	Suffixes	Tag
ي y		فعل fEl	`VePi`	ونَ wna	`MaPlThSj`
وي wy	`PrCo+`	فاعِل fAEl	`VePi`	damma mark	`MaSnThDc`
				نَ na	`FePlThDc`

Table 5.2: Sample of prefixes, forms, suffixes for some imperfect verb words

There are two important things to point out here. The first is that the tags attached to forms and suffixes in Table 5.2 are valid tags only if these suffixes attached to these forms. In other word, the tag of the form may change depending on the suffixes attached to this form. For example, the tag [`VePi`] associated with the form فاعل (second line in Table 5.2) is valid only in case the form فاعِل is combined with the suffixes presented in Table 5.2. If the suffixes changed, the tag of the forms should also need to be changed. For example, Table 5.4 shows that the tag of the form فاعِل, *fAEl* is changed due to the changes happening in suffixes. The combined pattern lexicon can be seen in Table 5.5.

PNo	pattern	Transliteration	Tag
1	يفعلونَ	yfElwna	VePiMaPlThSj
2	يفعلُ	yfElu	VePiMaSnThDc
3	يفعلنَ	yfElna	VePiFePlThDc
4	يفَاعلونَ	yfAElwna	VePiMaPlThSj
5	يفَاعلُ	yfAElu	VePiMaSnThDc
6	يفَاعلنَ	yfAElna	VePiFePlThSj
7	ويفعلونَ	wyfElwna	PrCo+VePiMaPlThSj
8	ويفعلُ	wyfElu	PrCo+VePiMaSnThDc
9	ويفعلنَ	wyfElna	PrCo+VePiFePlThSj
10	ويفَاعلونَ	wyfAElwna	PrCo+VePiMaPlThSj
11	ويفَاعلُ	wyfAElu	PrCo+VePiMaSnThDc
12	ويفَاعلنَ	wyfAElna	PrCo+VePiFePlThDc

Table 5.3: Sample of pattern lexicon shows the patten for some imperfect verb words

Prefixes	Tag	Forms	Tag	Suffixes	Tag
		فَاعل fAEl	VePe	fatha mark	MaSnThSj
				تَ ta	MaSnScSj
				تمْ tm	MaPlScJs
				نَ na	FePlThSj

Table 5.4: Sample of prefixes, forms, suffixes for some perfect verb words

PNo	pattern	Transliteration	Tag
1	فَاعلَ	fAEla	VePeMaSnThSj
2	فَاعلتَ	fAElta	VePeMaSnScSj
3	فَاعلتمْ	fAEltm	VePeMaPlScJs
4	فَاعلنَ	fAElna	VePeFePlThSj

Table 5.5: Sample of pattern lexicon shows the patten for some perfect verb words

Usually, the prefixes have no tags[3] unless the prefixes represent a particle, such as, a conjunction particle, in this case a separate tag is to be associated with this particle to show that this word has a combined tag. For example, the word ويشرحُ, *wyshrhu*, "and to explain" has the following tag [`PrCo+VePiMaSnThDc`]. [`PrCo`] is the tag of conjunction particle و, *w*, "and" which appears in the word as well as the pattern. [`VePiMaSnThDc`] is the tag of the word يشرحُ.

The second thing is the tag of each suffix represents the inflectional feature of the word. Each form should have at least one suffix, that is, the last diacritical mark. The length of the suffixes ranges between 1 to 4 or 5 letters. The length of the prefixes on the other hand ranges between 0 to 4 or 5 letters. So, it becomes clear that the tags generated with patterns are detailed tags.

The rules in the pattern-based technique can be represent using the following general rule :

***Assign the tag* (T) *to the testing word* (W) *if the testing word matching the pattern* (P)**

where **T** is a variable over a set of tags in pattern lexicon, **W** is a variable over a set of testing words, and **P** is a variable over a set of patterns in pattern lexicon. For example, suppose the testing word W = يكتبُ, *yktbu*, "do writing". W is looked up in patterns lexicon to check for its correct pattern. The correct pattern here is P = يفعلُ, *yfElu* (the second pattern in Table 5.3). The tag [`VePiMaSnThDc`] which associated with the pattern يفعلُ is then extracted from pattern lexicon and assigned to the word يكتبُ.

An important question must be asked here. **How the testing word matched its correct pattern** ?

[3] In other POS tagger system built for Arabic, a separate tag such as [`Def`] used to represent the definite article ال, "Al". In the current system, this tag included with the inflectional feature of the word with the symbol [`Df`]

To answer this question, a novel algorithm has been developed and described in next section. The purpose of this algorithm is to show how the testing word is matched its correct pattern in the pattern lexicon.

5.3.2 Pattern-matching algorithm

Since the lexicon in AMT is a pattern lexicon not an Arabic words lexicon, an algorithm to match the Arabic word in the testing corpus with its correct pattern in patterns lexicon is required. A novel algorithm has been introduced in this work to achieve the above goal. The pseudo code of the pattern-matching algorithm is described in Algorithm 1. The steps of the algorithms with examples are described below in more detail.

Step - 1 :

The first step in the algorithm is responsible to return from the pattern lexicon all the patterns that have the same length of the testing word. For example, the word فكتبتنَ, *fktbtna*, "and they wrote" has the length[4] = 7 (see figure 5.3). The returned patterns that have the same length of the word فكتبتنَ are shown in Table 5.6. The next step **(Step - 2)** of the algorithm shows how to calculate the identical letters between the testing word (فكتبتنَ) and the fourth pattern (ففعلتنَ) as an example.

PNo	Paterrn	Word	Identical letters	Num
1	ستفعلكَ	فكتبتنَ	the last mark (Fatha)	1
2	فأفعلكَ	فكتبتنَ	the letters ف and last mark	2
3	فتفعلنَ	فكتبتنَ	the letters ف, ن, and last mark	3
4	ففعلتنَ	فكتبتنَ	the letters ف, ت, ن, and last mark	4

Table 5.6: Number of identical letters between the word فكتبتنَ and its patterns

[4] the last diacritical mark is counted as a letter of the word

```
Let W = Inflected word , P(i) = pattern of W , T(i) = Tag of W ,
L = Length of W or P(i) , R = Total number of patterns in lexicon ,
D = Total number of patterns have the same length of word ,
IL(j) = The number of identical letters between each pattern and
word ,
M = Total number of patterns have the maximum number of
identical letters with word
begin
    Get the word W ;
    D, M = 0 ;
    for i ← 1 to R do
        while L(P(i)) = L(W) do
            Return P(i), T(i);
            D = D + 1;
        end
    end
    for j ← 1 to D do
        Count the number of identical letters between P(j) and W;
        Store result in IL(j);
        Next j ;
    end
    for j ← 1 to D do
        Return P(j), T(j) which have the maximum number of IL(j);
        M = M + 1;
        Next j ;
    end
    for k ← 1 to M do
        Create a new pattern NP from W that is L(NP) = L(W) by
        changing W letters which correspond (mirror) only to f, E,
        1 letters in P(k);
        if NP = P (k) then
            Return P(k) and T(k);
            Exit the loop;
        else
            Next i;
        end
    end
end
```

Algorithm 1: Pattern-matching algorithm

Step - 2 :

The second step of the algorithm is responsible for calculating the number of identical letters between the testing word and the patterns which are returned from performing step-1. The aim of this step is to reduce the number of returned patterns. For example, the identical letters between the word فكتبتنَ and the pattern ففعلتنَ are shown in figure 5.3. The number of identical letters between the word فكتبتنَ and each returned pattern can be seen in Table 5.6.

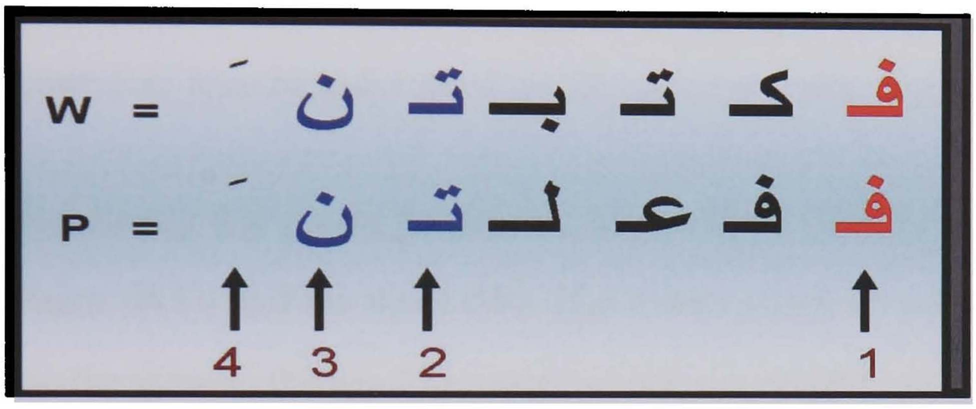


Figure 5.3: The identical letters between the word فكتبتنَ and the pattern فتفعلنَ

Step - 3 :

Choose the pattern(s) which have the maximum number of identical letters. Since the fourth pattern in Table 5.6 has the maximum number of identical letters with the testing word, the algorithm will chooses this pattern for the word فكتبتنَ.

So, in this case W = فكتبتنَ, P(1) = ففعلتنَ.

Step - 4 :

Replace the letters of W which correspond (Mirror) to the letters ف, *f*, ع, *E* and ل, *l* (the letters ف, ع and ل represent the root letters) in the pattern(s) (P) which have the maximum identical number with the word (W). Add the remaining letters in W without change and store the new pattern in NP. Figure 5.4 describes how to perform this step.

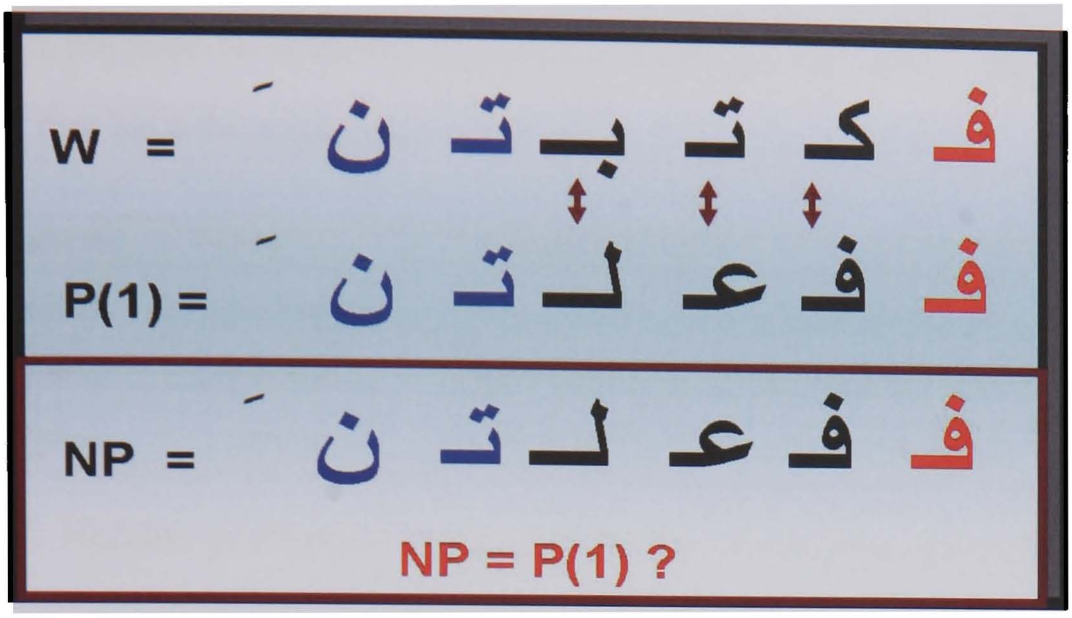


Figure 5.4: Matching the word فكتبتَن with the pattern ففعلتَن

Figure 5.4 shows clearly that a new pattern has been created with the same length of the original pattern (P(1)) and the word (W). The letters which do not correspond to the root form are the same in the word, the original pattern, and the new pattern. These letters represent the affixes which are added to the ground form (root). Since NP = P(1), this means that ففعلتَن is the correct pattern for the word فكتبتَن.

In most cases the algorithm is returned one pattern has the maximum number of identical letters with the testing word as in the above example. But, sometimes, more than one pattern has been returned, each pattern has the same number of identical letters with the testing word.

This step (**Step - 4**) is not used only to check that the only pattern which has the maximum number of identical letters with the testing word is the correct pattern, but also to choose the correct pattern in case the algorithm is returned more than one pattern, each has the same identical letters with the testing word.

For example, suppose W = the word يسمعونَ, *ysmEwna*, "to hear". Table 5.7 shows the patterns that have the same identical letters with the word يسمعونَ.

PNo	Paterrn	Word	Identical letters	Num
1	يفاعلَن	يسمعونَ	the letters ي, ع, ن, and last mark	4
2	يفتعلَن	يسمعونَ	the letters ي, ع, ن, and last mark	4
3	يفعلونَ	يسمعونَ	the letters ي, و, ن, and last mark	4

Table 5.7: Number of identical letters between the word يسمعونَ and its patterns

During this step, the algorithm is responsible to determine which one of the above patterns is a correct pattern for the word يسمعونَ. The first pattern يفاعلَن has been checked if it is the correct pattern for the word يسمعونَ or not. Figure 6.3 shows the result.

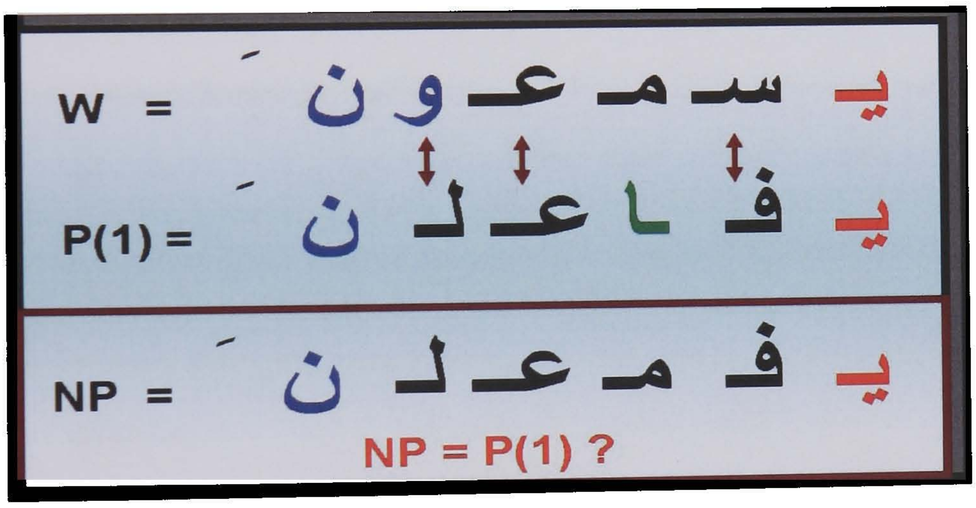


Figure 5.5: Matching the word يسمعونَ with the pattern يفاعلَن

It is clear from figure 5.5 that NP does not equal the pattern P(1), because the letter ـمـ, *miim* in the word (W) is differs from its corresponding letter in P(1) (ا, (*Alif*)). So, the pattern يفاعلَن in this case, is not the correct pattern of the word يسمعونَ.

Similarly, the pattern يفتعلَن is not the correct pattern of the word يسمعونَ as shown in figure 5.6, because the letter ـمـ, *miim* in the word (W) is differs from its corresponding letter in P(2) (ـتـ, (*taa*)).

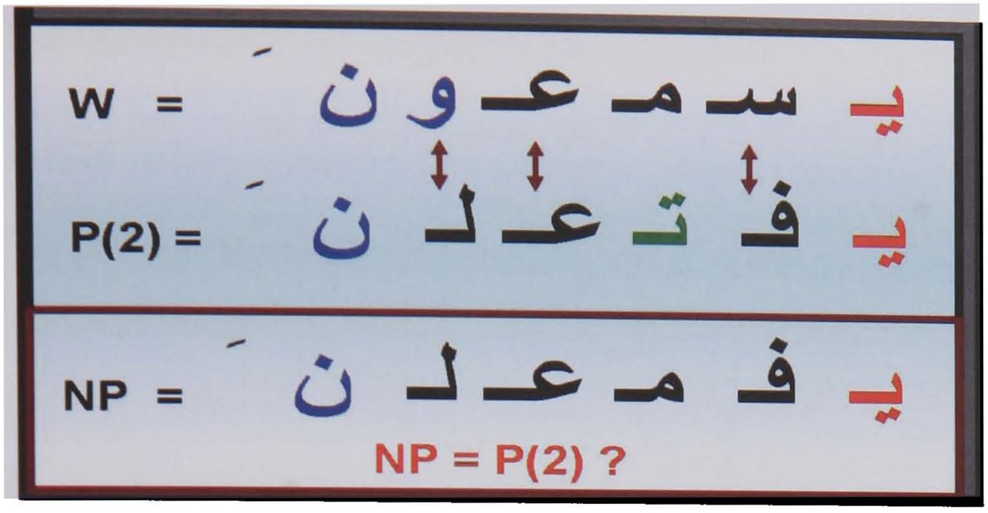


Figure 5.6: Matching the word يسمعونَ with the pattern يفتعلنَ

The last pattern يفعلونَ has been checked by the algorithm as shown in figure 5.7. Since NP = P(3), then the pattern يفعلونَ is the correct pattern of the word يسمعونَ. The algorithm in this case will choose the pattern يفعلونَ as a correct pattern for the word يسمعونَ.

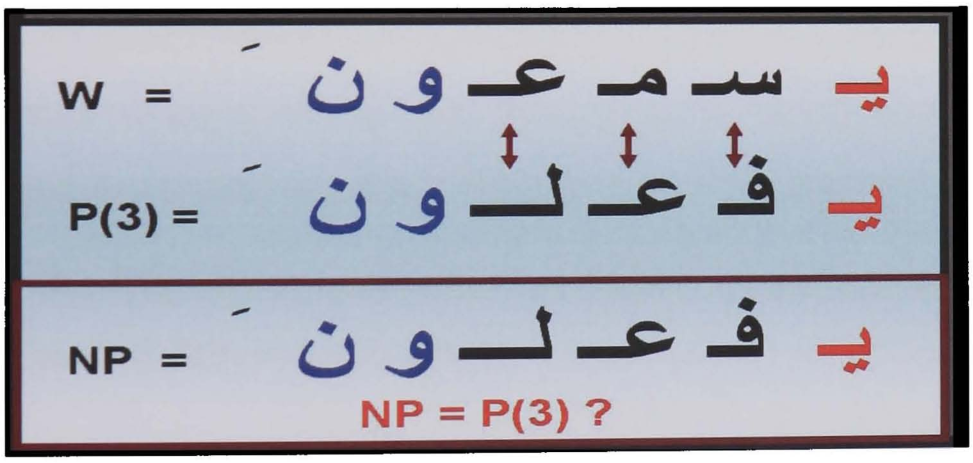


Figure 5.7: Matching the word يسمعونَ with the pattern يفعلونَ

Step - 5 :

The last step in the above algorithm is responsible to extract the tag associated with the correct pattern from pattern lexicon, and assigned this tag to the testing word. For example, the tag `VePiMaPlThSj` (see Table 5.3) is extracted and assigned to the word يسمعونَ.

5.4 Lexical and Contextual technique

The *pattern-based technique* described in section 5.3.2 which depend on the pattern of each word in the testing corpus constitute the main technique in this work. In fact, it is not easy for one person to generate all the patterns which cover all the words in the Arabic language. Since most words in Arabic belong to the noun class, difficulties may appear especially in collecting all the patterns of the words belonging to this class. In terms of the words belonging to verb class, the case is different. It is easy to collect the verb forms and all affixes associated with these forms, as the pattern lexicon is generated automatically.

The pattern lexicon in this work contains 8718 patterns. Most of these patterns are patterns for the words which belong to verb class. The tag set hierarchy (see 2.3) covers most types of sub-classes belong to noun class. Some of these sub-classes have certain patterns, for example, the patterns of adjective nouns, instrument nouns, verbal nouns and diminutive nouns, which are generated automatically and added to the patterns lexicon. The difficulties may appear in collecting and generating the patterns for other sub-classes especially common nouns.

As mentioned earlier, the patterns lexicon contains 8718 patterns, these patterns definitely not sufficient to cover all the Arabic words, especially, those words belonging to the noun and verb classes. For this reason, the *lexical and contextual technique* is used in this work to assist the *pattern-based technique* to tag those words not have patterns in lexicon, especially those words which belonging to common noun sub-class.

On the other hand, All the tags in the *pattern-based technique* are detailed tags, because these tags have been generated automatically with patterns. These tags not only represent the name of class (e.g. perfect verb, imperfect verb), but also included the inflectional feature of the word, such as, number, gender, person, and mood using the prefixes and suffixes attached to the verb form of the word. In contrast, the tags of those words belong to the noun or particle class and tagged by the *lexical and contextual technique* are vary from general to detailed.

As mentioned in section 3.4.2, Arabic language has a set of rules or signs, which have been described by Arab grammarians and used to distinguish nouns from verbs and particles. For example, they described these rules as follows :

1. **An Arabic word ends with nunation (tanween)**
2. **An Arabic word has the genitive case (end with kasra mark)**
3. **An Arabic word begins with definite artilcle ال Al**
4. **An Arabic word follows the particle يا yA**

In the Arabic language, neither the words belong to verb class nor the particle class can share the above rules. These rules have been taken into account when applying lexical and contextual rules.

5.4.1 Lexical Rules

Lexical rules are used to analyse words and take advantage of the internal structure of words. The triggers in the lexical rules depend on the character(s), affixes, and the last diacritical mark of the word. The name of the rules in *lexical and contextual technique* are written in the same way that Brill [37] has represented his rules and templates.

The names of the lexical rules (in parenthesis) and the description of each rule are given below:

Assign tag T to the current word if :

1- The last mark of the current word is X.**(CWDLM)**

2- The first character of the current word is C.**(F1CHCWD)**

3- The first two characters of the current word are C.**(F2CHCWD)**

4- The last two characters of the current word are C.**(L2CHCWD)**

5- The first three characters of the current word are C.**(F3CHCWD)**

Where X is a variable over the set of diacritic marks, C is a variable over the set of characters of the current word.

An example of a lexical rule is shown below. The list of lexical rules with examples can be seen in Appendix C.

- *Tanween Damm* **CWDLM** **NuCnNmId**

  This rule means: Assign **NuCnNmId** tag to the current word if the last diacritical mark of the current word is *Tanween Damm*.

  For Instance, the word رجلٌ, *rjlun*,"Man".

5.4.2 Contextual Rules

Contextual rules are used to assign the correct tag of the particular word based on the surrounding words or on the tags of the surrounding words. The triggers in the contextual module depend on the current word itself, and the tags or words on the context of the current word.

The names of the contextual rules (in parenthesis) and the description of each rule are given below:

Assign tag T to the current word if :

1- The preceding word is Z. **(PWD)**

2- The preceding tag is Y. **(PWDTAG)**

Where Z is a variable over all words in the testing corpus, Y is a variable over the set of tags.

An example of contextual rule is shown below. The list of contextual rules with examples can be seen in Appendix C.

- **NuCnGeId PWDTAG PrPp**

  This rule means: Assign **NuCnGeId** tag to the current word if the the tag of the preceding word is **PrPp**.

  For Instance, ”مِنَ   البيتِ”, *mna   Albyti*, ”from the house”.

On the most obvious problem in tagging the Arabic text is recognising proper nouns. A proper noun in Arabic may be represent the name of a specific person, place, organization, thing, an idea, an event, date, time, or other entity. Unlike English language, Arabic does not distinguish between lower and upper case letters; this makes it not nearly as easy to locate the proper nouns as in English text. Furthermore, these words may be solid or derived or words borrowed from another language (Arabised words), which add another level of complexity to recognising these words [15] [14].

Abuleil and Evens [15] presented a technique for tagging proper nouns in Arabic text, which depends on the keywords stored in a lexicon. Table 5.8 shows how they have classified these keywords.

In this work, their classification (keywords) have been applied, but by using the *lexical and contextual technique* instead of using a lexicon.

For example, **NuPo PWD** مدينةَ or مدينةُ or مدينةِ

No	Classification	Example
1	Personal names (title)	**السيد** رمزي, **Mr**.Ramzi
2	Personal names (job title)	**الرئيس** صالح, **President**.Saleh
3	Organization names	**جامعة** ديمونتفورت, DeMontfort **University**
4	Locations (political names)	**جمهورية** فرنسا, French **Republic**
5	Locations (natural) names)	**مدينة** عمان, Amman **City**
6	Times	**شهر** أيلول, **Month** of September
7	Product	**كاميرا** نيكون, Nikon **Camera**
8	Events	**معرض** سيارات, Cars **Exhibition**

Table 5.8: Classification of Proper noun

This rule means: Assign **NuPo** tag to the current word if the preceding word is (مدينةَ or مدينةُ or مدينةِ). For Instance, مدينةُ لندن, "London City".

Furthermore, the particle lexicon contains those words belonging to particle class has been built in this work. The motivation behind building the particle lexicon comes from the fact that, during the initial experiments which have been done to test the tagger, some words have been tagged incorrectly. Since the pattern-based module has been designed for those words belonging to verb class or noun class, some words belonging to particle class have been matched the incorrect patterns when applying the pattern-matching algorithm to those words. For example, the word ومنها, *wmnhA*, "and-from it" match the pattern فعلها,*fElhA* as shown in figure 5.8 and takes an incorrect tag, because this word belongs to particle class while all the words follow the pattern فعلها belonging to verb class, such as the word كتبها, *ktbhA*, "he wrote it" as shown in figure 5.9. Thus, to reduce the errors in tagging such words and enhance the performance of the tagger system, the decision has been taken to generate a separate particle lexicon.

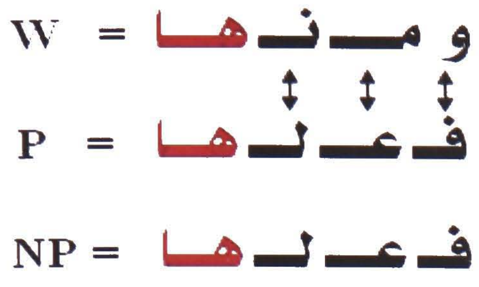


Figure 5.8: Matching the word ومنهَا with the pattern فعلهَا

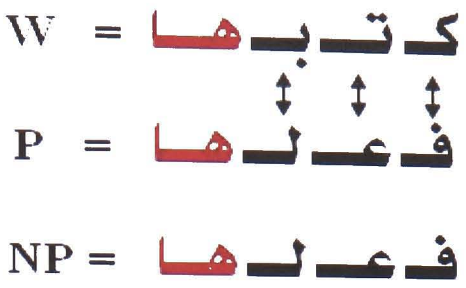


Figure 5.9: Matching the word كتبهَا with the pattern فعلهَا

The particles lexicon is generated automatically by combining: a single lexicon of all prefixes including all valid concatenations, a single lexicon of all Arabic words belonging to particle class and a single lexicon of all suffixes.

Table 5.10 shows a sample of particles lexicon which generated from Table 5.9 elements.

Prefixes	Tag	particle word	Tag	Suffixes	Tag
و, *w*, ”and”	PrCo+	في, fy, ”in”	PrPp	هَا, *hA*	FeSn
		من, mn, ”from”	PrPp	كُم, *km*	MaPl
		إن, In, ”if”			

Table 5.9: Sample of prefixes, particle word, suffixes for some particles words

particle	Tag
في	PrPp
من	PrPp
إن	PrAn
فيهَا	PrPpFeSn
فيكُم	PrPpMaPl
منهَا	PrPpFeSn
منكُم	PrPpMaPl
إنهَا	PrAnFe
إنكُم	PrAnMaP

particle	Tag
وفي	PrCo+PrPp
ومن	PrCo+PrPp
وإن	PrCo+PrAn
وفيهَا	PrCo+PrPpFeSn
وفيكُم	PrCo+PrPpMaPl
ومنهَا	PrCo+PrPpFeSn
ومنكُم	PrCo+PrPpMaPl
وإنهَا	PrCo+PrAnFe
وإنكُم	PrCo+PrAnMaPl

Table 5.10: Sample of particles lexicon

5.5 A description of the tagger system

5.5.1 Tagger Modules

The main function of AMT is to take an untagged partially-vocalised (the diacritical mark assigned only to the last letter of each word in testing corpus) Arabic text as input, and to produce a POS tagged partially-vocalised Arabic corpus. AMT as shown in figure 5.10 is composed of three main modules : *Tokeniser Module*, *Pattern-based Module*, and *Lexical and Contextual Module*.

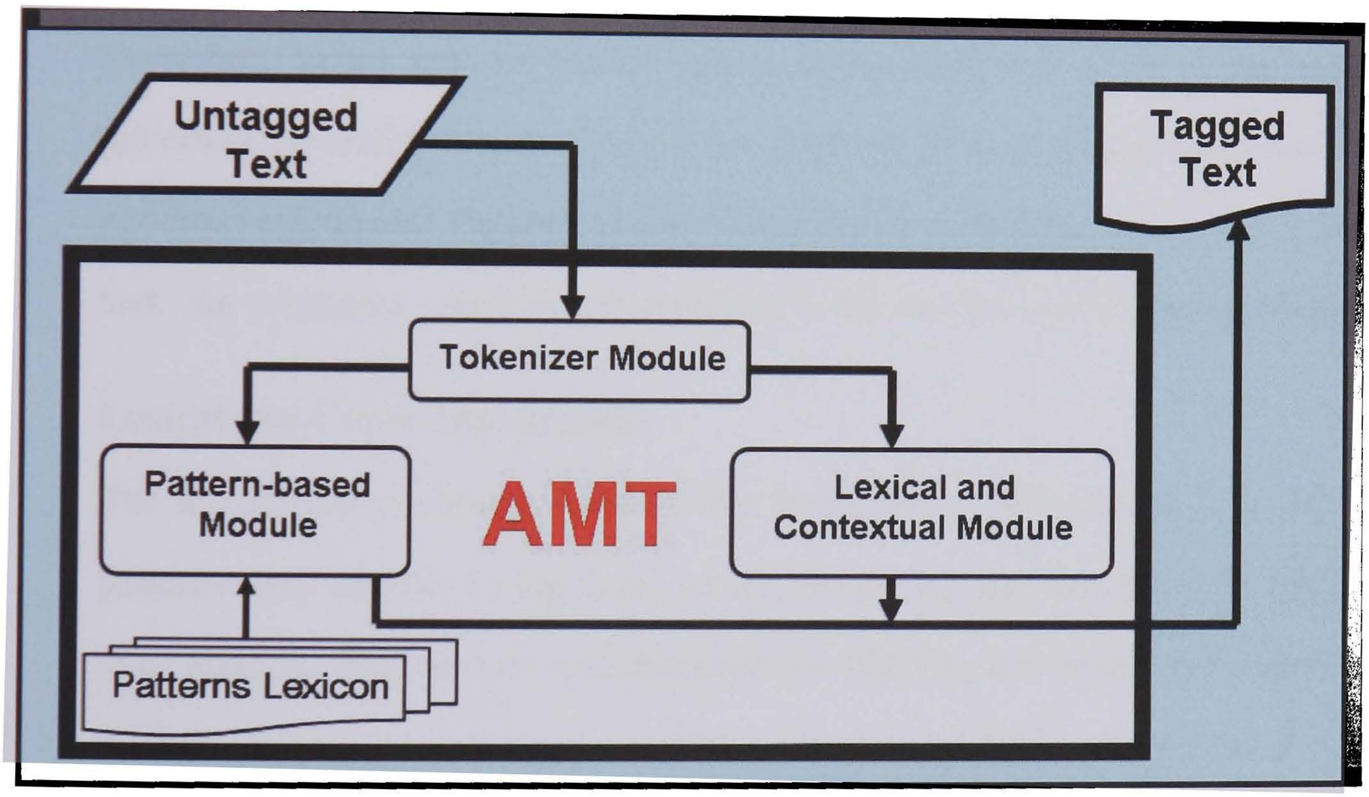


Figure 5.10: An overview of AMT

The list below describes the function of each module in more detail.

- **Tokeniser Module**

  A *token* is not just a word. It is defined as a sequence of characters having a collective meaning [17]. A token represents any special character, number and word. The main function of a tokeniser module is to convert the untagged input text into a form that is more manageable by the machine. This conversion is called *tokenisation*. The tokenisation process is responsible for locating an untagged input text and identifying words, punctuation marks, numbers and other marks using the space as a delimiter. The tokeniser simply separates the input text into tokens including the splitting of punctuation marks (such as full stops and commas) from their previous words.

- **Pattern-based Module**

  The main function of this module is to look up each testing word in the patterns lexicon. It performs the pattern-matching algorithm steps to match each word in the testing corpus with its correct pattern in the patterns lexicon. If the correct pattern of the testing word is found in the patterns lexicon, the tag extracted from patterns lexicon and assigned to the testing word. After this module finished its task, the remaining words are then passed to the *lexical and contextual Module*.

- **Lexical and Contextual Module**

  The lexical and contextual module has been built in this system to assist the pattern-based module to tag those words not having patterns stored in the patterns lexicon. This module is responsible for applying the lexical and contextual rules to assign the correct tag to each word not tagged by the pattern-based module.

5.5.2 Tagging Process

AMT performs many steps during the tagging process as shown in figure 5.11. During the tagging process, the token is first looked up in the particle lexicon. If it is found, then the tag extracted and associated to the token. The token is then passed to the pattern-based module, where the pattern-matching algorithm is applied to the token to

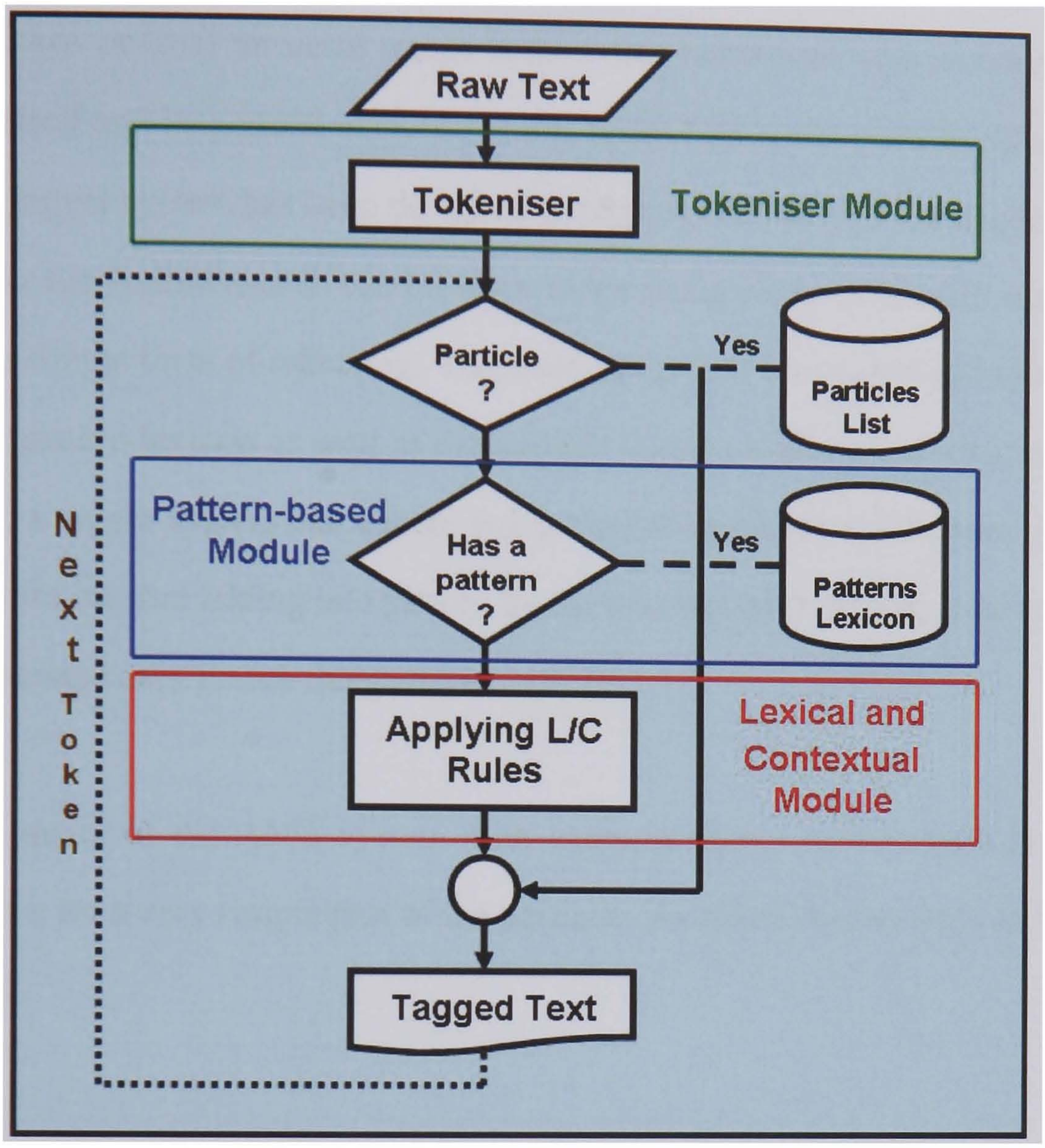


Figure 5.11: How AMT performs tagging

check if the token has a pattern in the pattern lexicon or not. If the token matches its pattern, then the tag is extracted from the pattern lexicon and assigned to the token. If the pattern of the token is not found in the pattern lexicon, then it is passed to the lexical and contextual module.

At this stage, the lexical and contextual module has been applied to assign the correct tag to each token which has not been tagged by the pattern-based module. Finally, for those very few tokens still untagged by the above modules, a user intervention menu has been designed in the main menu (see figure 5.12) of the system to allow the user to add a new pattern and its general tag or at least the simple form of tag (e.g **Ve**)

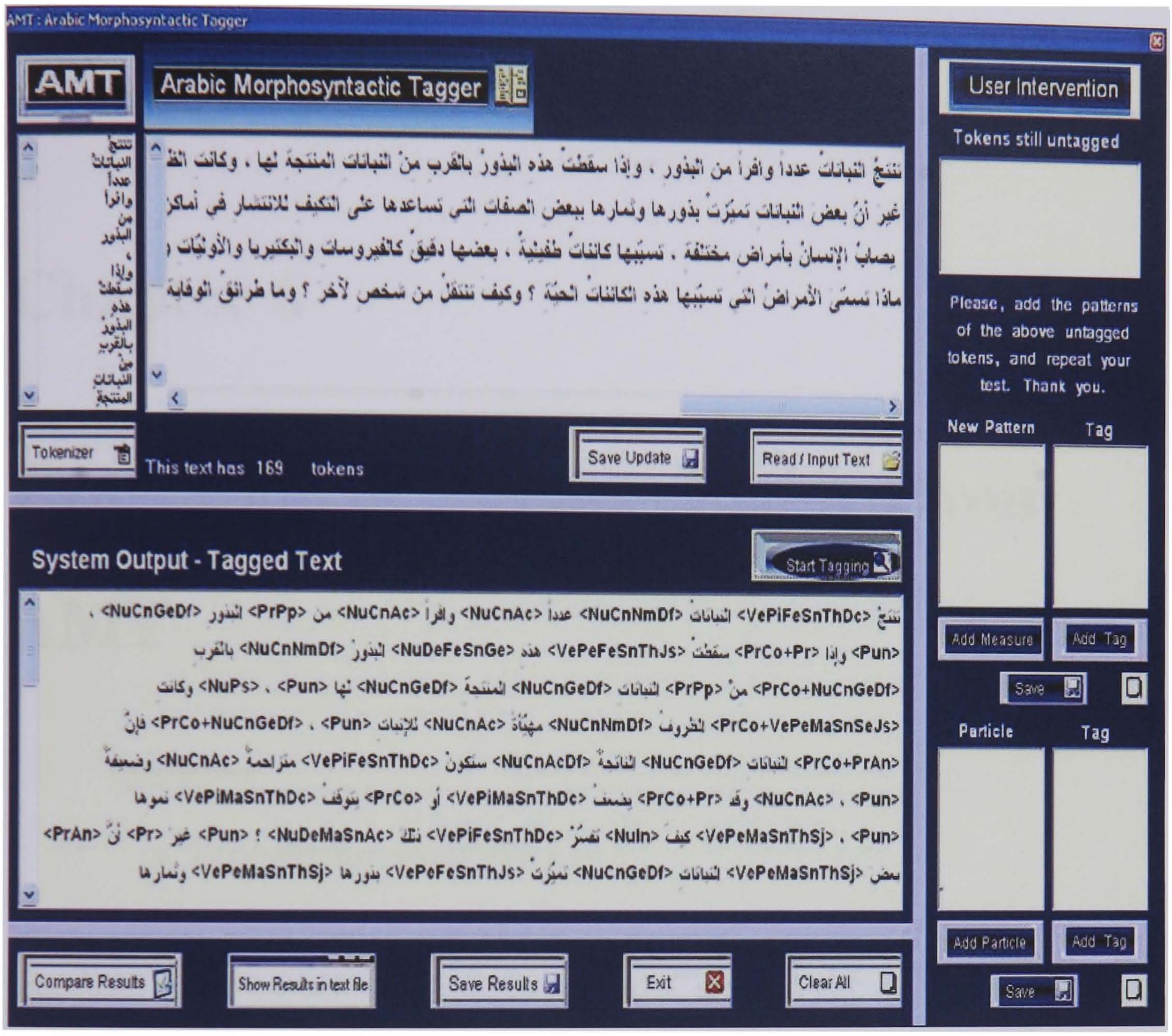
for verb words or (Nu) for noun words if the token belongs to verb or noun class, or the token itself and its general tag (Pr) if this token belongs to particle class.
Since this tagger system has been designed to tagging Arabic text, it is expected that it is easier for the Arabic user to use his knowledge to tag those words still untagged by adding the simple form of token tag. The main purpose of user intervention menu is to enrich the pattern lexicon as well as the particle lexicon with new entries, which lead to develop a tagger system can accept any partially-vocalised Arabic text. It is interested to point out that adding one pattern by the user means that many Arabic words in Arabic language may match this pattern.

The main menu of the AMT system with example shows how to perform the tagging process for a very simple part of the partially-vocalised Arabic text can be seen in figure 5.12.

5.6 Chapter Summary

This chapter presented the design and implementation steps for a new rule-based POS tagger called AMT : Arabic Morphosyntactic Tagger. We defined the characteristics of AMT tagger : free manually tagged lexicon or training corpus, word level tagging and tagging partially-vocalised Arabic text. In the current literature such a tagger does not exist. A new technique with a novel algorithm has been applied for AMT system. Since a lexicon of Arabic words or training corpus in this system is not needed, instead, we generated a lexicon of patterns which are associated with the last diacritical mark and generated automatically.

In this work the word "pattern" is used to represent the template of the whole word in-



تنتج
النباتات
عددا
وافرا
من
البذور
،
وإذا
سقطت
هذه
البذور
بالقرب
من
النباتات
المنتجة

تنتجُ النباتاتُ عددا وافرا من البذور ، وإذا سقطتْ هذه البذورُ بالقرب منْ النباتات المنتجة لها ، وكانت الظ
غير أنّ بعض النباتات تميّزتْ بذورها وثمارها ببعض الصفات التي تساعدها على التكيف للانتشار في أماكن
يصابُ الإنسانُ بأمراض مختلفة . تسبّبها كائناتٌ طفيلية . بعضها دقيقٌ كالفيروسات والبكتيريا والأوليّات (
ماذا تسمّى الأمراضُ التي تسبّبها هذه الكائنات الحيّة ؟ وكيف تنتقلُ من شخص لآخر ؟ وما طرائقُ الوقاية

This text has 169 tokens

تنتجُ <VePiFeSnThDc> النباتات <NuCnNmDf> عددا <NuCnAc> وافرا <NuCnAc> من <PrPp> البذور <NuCnGeDf> ،
<Pun> وإذا <PrCo+Pr> سقطتْ <VePeFeSnThJs> هذه <NuDeFeSnGe> البذورُ <NuCnNmDf> بالقرب
<PrCo+NuCnGeDf> منْ <PrPp> النباتات <NuCnGeDf> المنتجةَ <NuCnGeDf> لها <Pun> ، <NuPs> وكانت
<PrCo+VePeMaSnSeJs> الظروفُ <NuCnNmDf> مهيّأةً <NuCnAc> للإنبات <Pun> ، <PrCo+NuCnGeDf> فإنّ
<PrCo+PrAn> النباتات <NuCnGeDf> الناتجةَ <NuCnAcDf> ستكونُ <VePiFeSnThDc> متزاحمةً <NuCnAc> وضعيفةً
<Pun> ، <NuCnAc> وقد <PrCo+Pr> يضعفُ <VePiMaSnThDc> أو <PrCo> يتوقفُ <VePiMaSnThDc> نموها
<Pun> ، <VePeMaSnThSj> كيف <NuIn> تفسّرُ <VePiFeSnThDc> ذلك <NuDeMaSnAc> ؟ <Pun> غير <Pr> أنّ <PrAn>
بعض <VePeMaSnThSj> النباتات <NuCnGeDf> تميّزتْ <VePeFeSnThJs> بذورها <VePeMaSnThSj> وثمارها

Please, add the patterns of the above untagged tokens, and repeat your test. Thank you.

Figure 5.12: Tagging process for simple part of text

cluding the prefixes, form (root+infixes) and suffixes, which are attached to the word. The main technique based on the pattern of the word instead of the word itself.

In addition, the lexical and contextual rules have been used in this system to assist the pattern-based technique to tag those words not having a pattern stored in pattern lexicon. The AMT system presented in this work deals with partially-vocalised Arabic text. It is the first POS tagger uses purely rule-based approach. A full description of AMT tagger system modules and the function of each module also has been addressed in this chapter. Finally, we described the tagging process that AMT system carried out.

Chapter 6

Evaluation of Results obtained from AMT

Objectives

- To present the testing data sets.
- To define the measure was used to calculate system performance.
- To describe the experiments been done to evaluate AMT system.
- To explain the analysis of results.
- To present AMT system shortcomings

6.1 Testing Data sets

A partially-vocalised Arabic text is needed to test the AMT system. The lack of large partially-vocalised Arabic corpus is one of the problems we faced. In order to obtain the testing corpus for the tagger, a new partially-vocalised Arabic corpus has been compiled. It contains 20,000 words. Since the text in school textbooks contains diacritics, the corpus is extracted and collected from these textbooks via the official site of ministry of education[1]-Jordan, with a permission and authorization from the department of curricula and textbooks management (see Appendix D).

The text in testing corpus had been normalised manually; that is, the diacritics other than the last diacritical mark have been removed and the last diacritical mark has been added to those words do not have it. Despite that not all words in school textbooks have diacritics, especially for the higher level classes, but the text in school textbooks is still the closest.

The aim of the normalisation process which has been done with consultation and collaboration of an Arabic linguist[2] is to ensure that each word in the corpus is attached only with the correct last diacritical mark. Also, the corpus is manually tagged for comparison with the system tagged texts.

The corpus is chosen and extracted from different books for different levels of school classes. It is not limited to a particular domain; it covers a wide range of topics such as scientific topics and literary topics.

Test data for the experiments was taken from the testing corpus. The data sets consists of raw original Arabic script words; no further annotations exist for this data set. Data spread across three sets :

1. **Set-1 :** consists of 3170 words representing several articles extracted from the book of computer science and other science topics, such as biology for different

[1] http://www.elearning.jo/eduwave/elearningme.aspx

[2] The author gratefully acknowledge the collaboration of Mr. Walid Alqrini - Ministry of Education - Jordan. Email: walidalqrini123@yahoo.com

secondary school classes.

2. **Set-2 :** consists of 7620 words representing several articles extracted from the book of Arabic language topics for classes 7, 8, and 9 of elementary level.

3. **Set-3 :** consists of 9210 words representing several articles extracted from the books of literary topics for classes 10, 11 and 12 of secondary level.

6.2 AMT Experiments and accuracy measurement

Five experiments were done to evaluate the AMT tagger system. The first three experiments were performed on set-1, set-2, and set-3 respectively. The fourth experiment was performed to calculate the ratio of pattern-based module and lexical and contextual module that have been applied in the above three experiments. The last experiment was done on a different text, that is the Quran text. A sample of the Quran text was taken from chapter Almulk and Alforqan, it contains 1016 words. The diacritics were removed except the last diacritical mark. The aim of this experiment is to get a picture of the AMT performance on a different text. The results of this experiment also described in this chapter with more detail.

There are several measurements used to indicate the performance of tagger systems. *Success rate*[3], *ambiguity*[4], *recall and precision* are the most popular measures which try to indicate the accuracy of the tagger output([73], p.82). *Success rate* measure is used in case the tagger is assigned a single tag to each token as the tagger presented in this work. It is expressed as a percentage and defined as follows :

$$Success\ rate\ = \frac{Number\ of\ correctly\ tagged\ tokens}{Total\ number\ of\ tokens}$$

[3] also called correctness or score

[4] also called average number of tags per token

Ambiguity measure is used when the tagger is assigned multiple tags per token. *Ambiguity* is calculated by dividing the total number of tags by the total number of tokens. *Recall and precision*, which find their original in information retrieval are also an alternative pair of measures used in tagging. *Recall* is calculated by dividing the number of correct token-tag pairs that is produced, by the number of correct token-tag pairs that is possible. *Precision* is the number of correct token-tag pairs that is produced, divided by the total number of token-tag pairs that is produced. Like the success rate measure, ambiguity and recall and precision are expressed as a percentage ([73], p.83).

Since the AMT tagger presented in this work produces a single tag to each word in testing corpus . *success rate* measure was used in to indicate the performance of the AMT system. The success rate for each experiment and the ratio of tag types which have been used in each experiment were calculated. In addition, the distribution of POS classes for the first three experiments has also been addressed. The details of the results for each experiment are described in the following sections (6.2.1 - 6.3.1). Section 6.3 describe the analysis of all experiment results.

6.2.1 Experiment-1

The first experiment was performed on the first set. AMT correctly tagged 89% of set-1 words as shown in figure 6.1.

Out of the correctly tagged words, 66% of the tags which were assigned to tokens in the first experiment are detailed tags which included inflectional feature for each word (see figure 6.2). This ratio indicates that the majority of the correctly tagged tokens were not at the general level. In addition, the distribution of POS classes for the text in experiment-1 can be seen in figure 6.3. It is expected that the ratio of tokens in the text

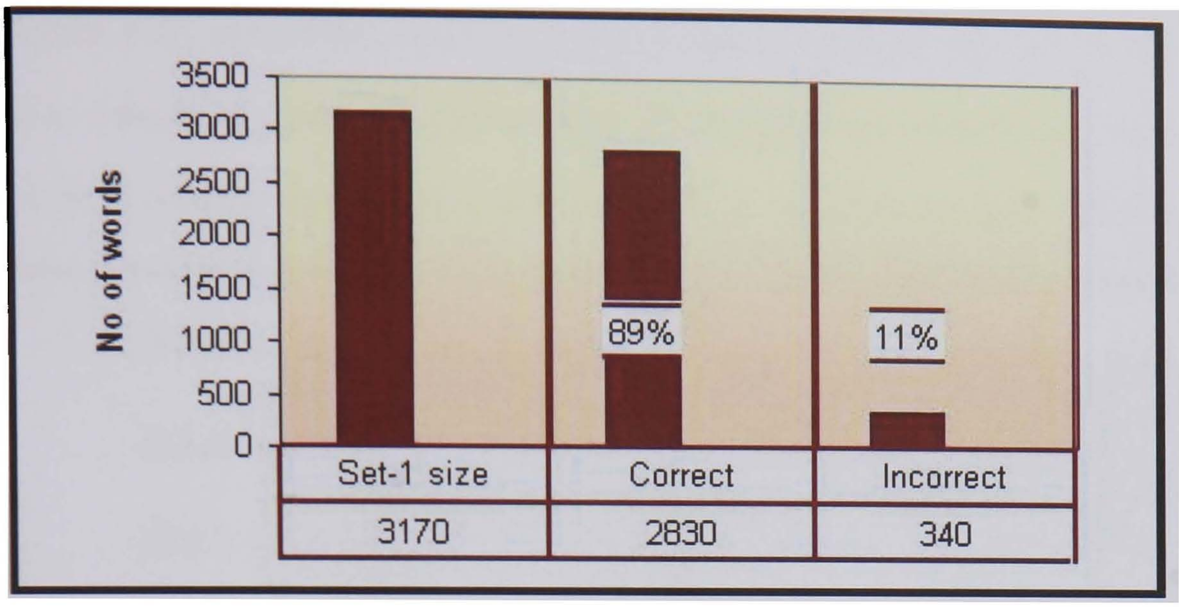


Figure 6.1: Success rate of experiment-1

which belong to noun class is a higher ratio since most of the Arabic words belong to noun class rather than any other POS classes. Usually but not always (depend on the testing text) particles in Arabic have the second higher ratio after the ratio of nouns.

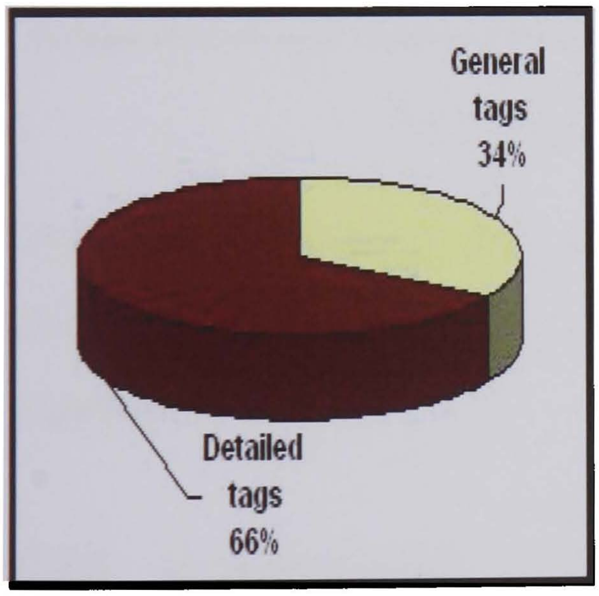


Figure 6.2: Detailed and general tags ratio in experiment-1

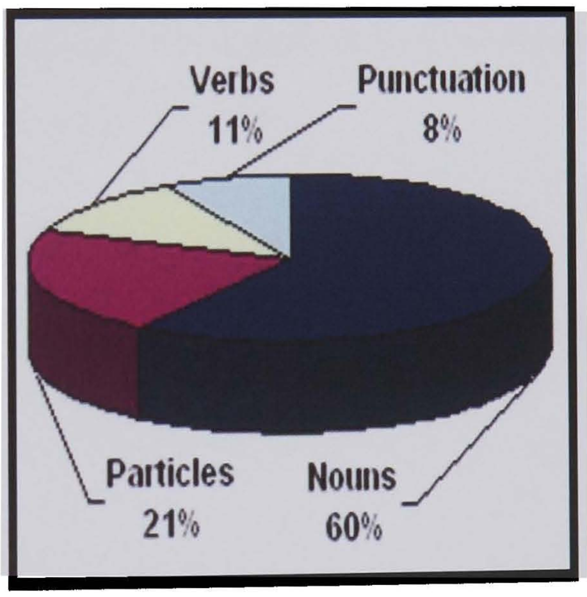


Figure 6.3: Distribution of POS classes in experiment-1

6.2.2 Experiment-2

The second experiment was performed on the second set which contains 7620 words. AMT correctly tagged 94% of set-1 words as shown in figure 6.4. During this experiment (figure 6.5), out of the correctly tagged words, 78% of tags which have been assigned to tokens in the secod experiment are detailed tags, while 22% are general tags. The ratio is varies according to the type of text. In addition, figure 6.6 shows that 63% of text tokens in this experiment belong to noun class, while 13% belong to verb class. Tokens belonging to particle class and puctuation class consititute 16% and 8% respectivally.

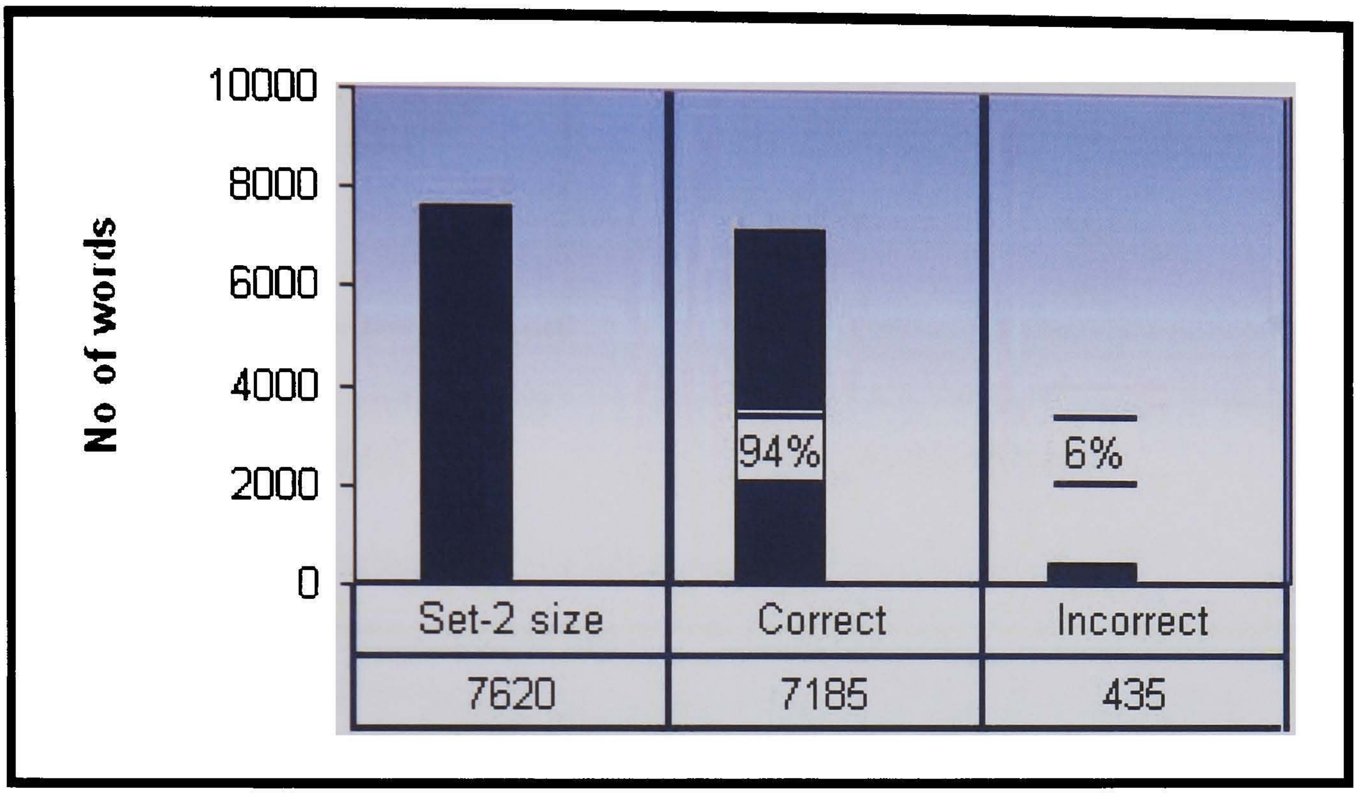


Figure 6.4: Success rate of experiment-2

6.2.3 Experiment-3

The third experiment was performed on set-3 which contains 9210 words. Out of set-3 size, AMT correctly tagged 91% of set-3 words as shown in figure 6.7. Out of the correctly tagged words in this experiment, the ratio of detailed tags which have been assigned to tokens is 59% while 41% are general tags (see figure 6.8). On the other hand, 67% of text tokens in this experiment belong to noun class, while 10% belong to verb class. Also, figure 6.9 shows that 17% of text tokens in this experiment belong to

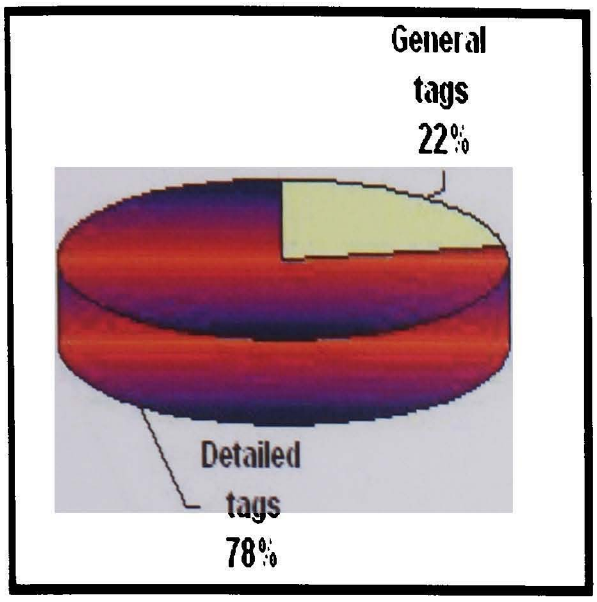


Figure 6.5: Detailed and general tags ratio in experiment-2

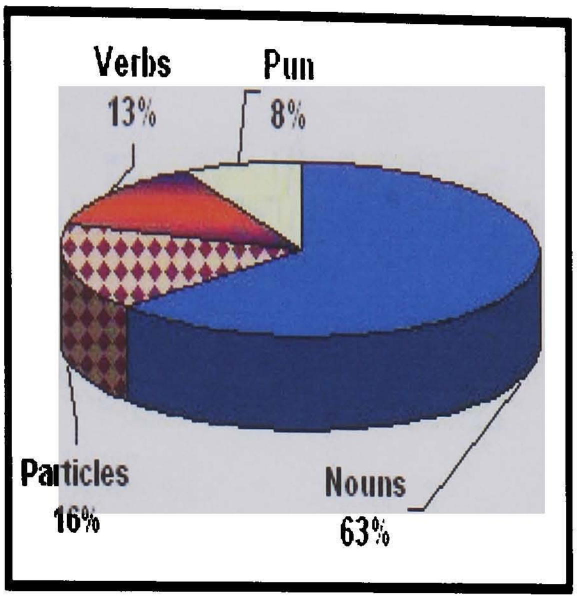


Figure 6.6: Distribution of POS classes in experiment-2

particle class and the ratio of tokens belonging to puctuation class is 9%.

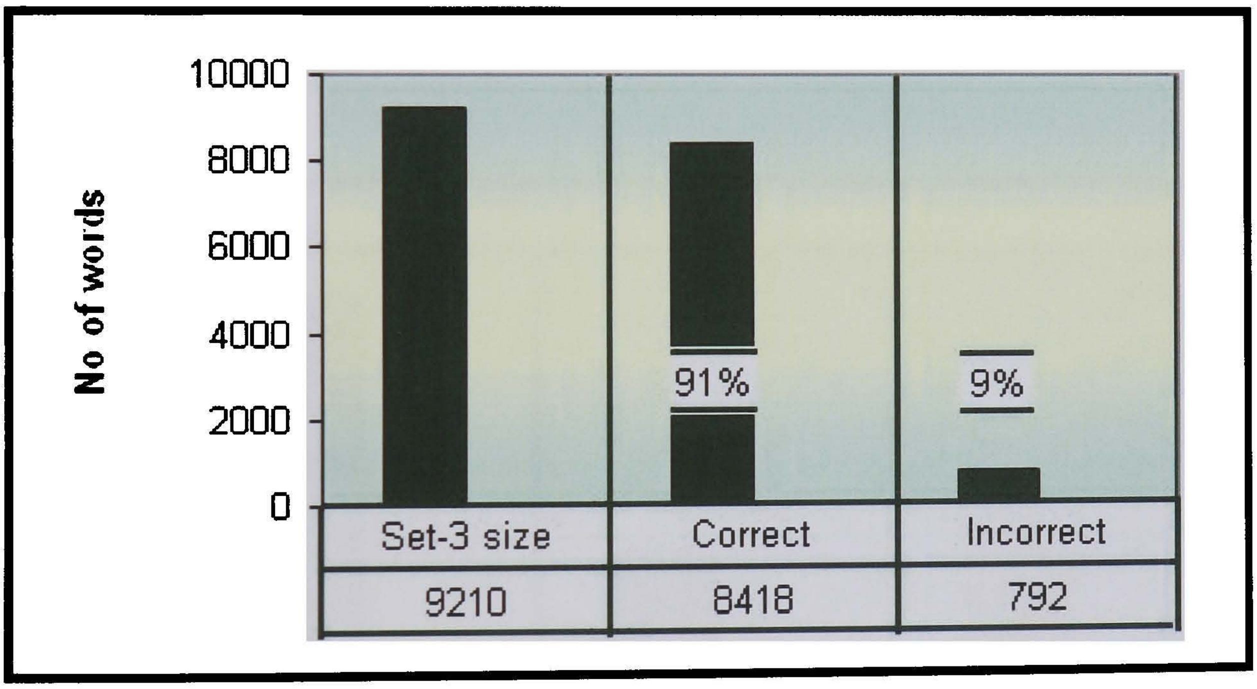


Figure 6.7: Success rate of experiment-3

6.2.4 Experiment-4

This experiment was performed to get a picture of the ratio of pattern-based module and lexical and contextual module that have been applied in the above three experiments. Figure 6.10 shows that 91% of testing coprus tokens are tagged correctly. Out

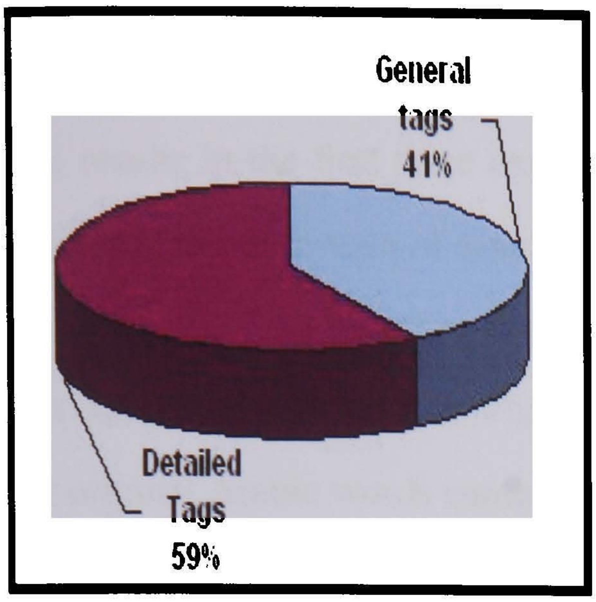


Figure 6.8: Detailed and general tags ratio in experiment-3

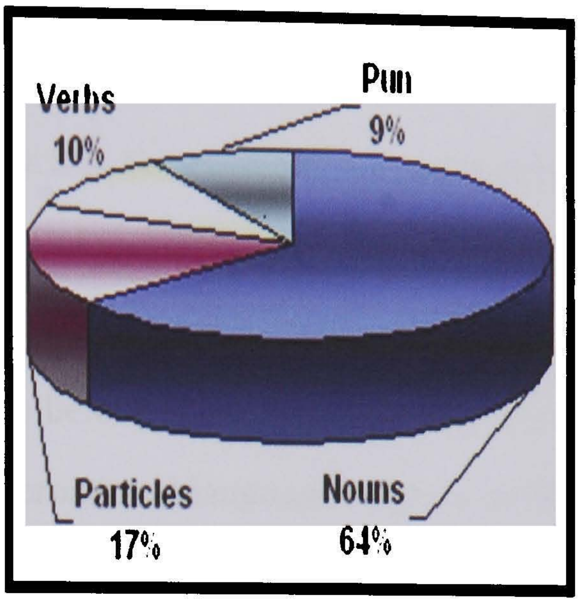


Figure 6.9: Distribution of POS classes in experiment-3

of the tokens tagged correctly, 48% of correctly tagged tokens are achieved by applying pattern-based module while 52% are achieved by applying lexical and contextual module.

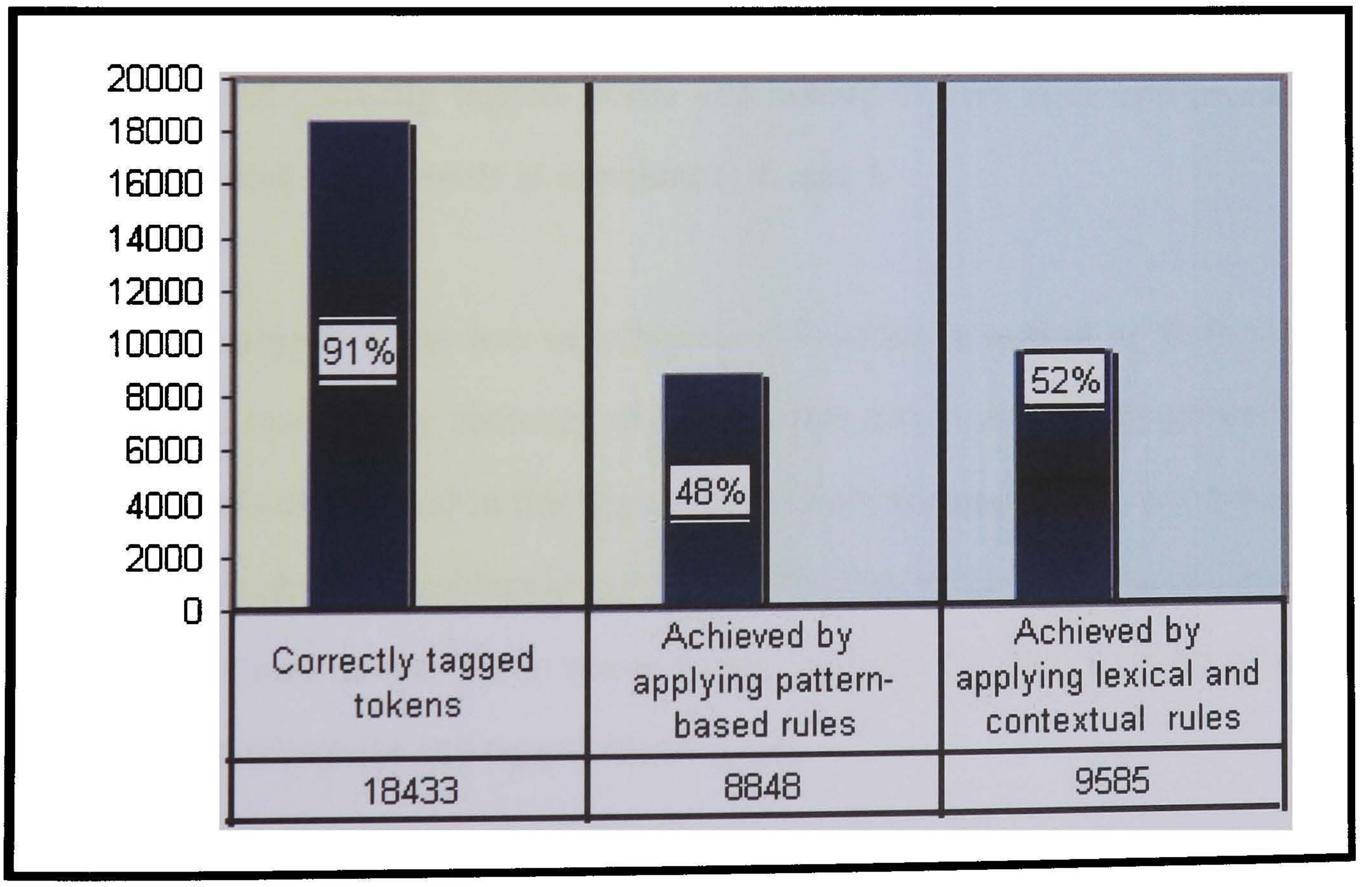


Figure 6.10: Percentage of rules applicability based on type

6.3 Experimental results Analysis

The results in the first three experiments show that the correctly tagged words vary according to the domain of each text. The style and text content is one of the main reasons that affect the accuracy of the tagger system. The text in experiment-1 is related to a computer science topic where some words belong to Arabised words; which are not original Arabic words came from other international languages and do not have a root or pattern. For example, the word كمبيوتر, *kmbywtr*, "computer". Most of these words are tagged incorrectly.

The percentage of correctly tagged words in experiment-2 is higher than experiment-1 and experiment-3. As the text of experiment-2 is related to Arabic language topic and specified for school level where most of words in the text of this experiment are original Arabic words which have a root and a pattern, it is an expected result. In addition, the percentage of correctly tagged words that belong to verb class and punctuation class is higher than those words in experiment 1, and 3.

The different subject of the text in experiment-3 which is related to literary topics is probably the reason why accuracy of tagging this text is low. Many proper nouns and Arabised words are used in this type of text. Since recognising proper nouns constitutes the most obvious problem in tagging Arabic text, most of the errors came from proper and Arabised nouns. These words belonging to proper and Arabised nouns are very difficult to recognise and tagged incorrectly.

In addition, Arabic has irregular verb words such as, the word ضلّ, *Dl=a*, "to go astray". Also some words in Arabic language are considered as primitive verbs such as, بئسَ, *b'sa*, "what a bad ... !". These words are also tagged incorrectly. For example, the word بئسَ matches the pattern فعلَ and the wrong tag assigned to this word.

The above three experiments show that most words in the Arabic language belong to noun class followed by particles, verbs and punctuation marks. An overall accuracy of the tagger system has been calculated by comparing the tagger system output with the goal corpus that is manually tagged. The tagger achieves 91%. Since, there is no training corpus in this system, this accuarcy is very good.

On the other hand, 48% of accuracy is achieved by applying the pattern-based module while 52% is achieved by applying the lexical and contextual module (see figure 6.10). Since the ratio of the patterns in the lexicon which belong to verb class is higher than the patterns of words which belong to noun class on the one hand, and most of the words in the testing corpus which belong to noun class rather than any other POS class on the other hand, it is natural these words have been tagged using lexical and contextual rules. For this reason, the accuracy achieved by applying the pattern-based rules is lower than achieved by applying the lexical and contextual rules.

One of the problems we are faced during experiments is the tag of the passive perfect verb. The passive perfect verb word is tagged and assigned with the same detailed tag assigned to active perfect verb since these words (passive and active perfect verb) share the same last diacritical mark. For example, the words كتبَ, *katba*, "he wrote", and كُتبَ, *kutba*, "it was written", the former represents an active perfect verb while the later represents passive perfect verb. Since both words share the same last diacritical mark and match the pattern فعلَ[5], AMT will be extracted and assigned the detailed tag

[5] ignore any diacritical mark other than the last one

VePeMaSnThSj to both words which means *Perfect Verb, Masculine gender, Singular number, Third Person, Subjunctive mood.* This detailed tag is correct for the former word (كَتَبَ) because this word means "there is a gentleman who wrote" and the inflectional features MaSnThSj describes that clearly. While it is not correct (except mood feature(Sj)) as a detailed tag for the later word (كُتِبَ) since this word which describes something other than human (book,lesson) has been written. At the same time, the general tag VePe is valid and correct for both words (كَتَبَ and كُتِبَ). This example shows that a smaller tag set (general) may contribute to increase the performance of the tagger.

AMT system presented in this work does not differentiate between the passive and active perfect verb and assign a detailed tag to both words. This problem appears only to those words represent passive perfect verb. Despite that the general tag is valid and correct for those words, but solving this problem means to adding another additional diacritic mark to the first letter of each word in the testing corpus and each pattern in the pattern lexicon which requires great effort and time compared with the very scarce number of words that can be found in testing corpus since most of perfect verb words in Arabic are active perfect verb. For example, out of our testing corpus words, 0.0005% of words are passive perfect verb. Thus, the general tag assigned to these words.

Another problem appeared during the experiments relates to nouns end with long vowel ا, *Alif.* Some nouns are wrongly matched with verb patterns. As an example, the word نموَها, *nmwhA*, "growth" matches the pattern فعلهَا as shown in figure 6.11. Since the pattern فعلهَا is a verb class pattern, an incorrect tag was extracted from patterns lexicon and assigned to the word نموَها, because this word belongs to noun class. The main reason behind the error in matching the incorrect pattern is, the pattern as

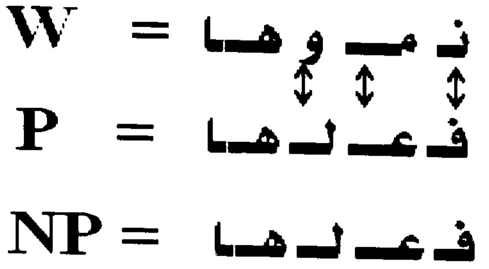


Figure 6.11: Matching the word نموهَا with the pattern فعلهَا

well as the word do not ended with a diacritical mark, instead they are ended with the long vowel letter (ا, *Alif*), this letter letter fills the place of the last diacritical mark (fatha mark).

A very few number of words are ended with the ا, *Alif* which belonging to noun class, but these words still remaining the pattern-matching algorithm is not 100% accurate. The best soluation to solve this problem is to compile a lexicon contains all the Arabic root words. One more step may add to the pattern-matching algorithm. The aim of this new step is to extract from the testing word the three letters which corresponding the root letters (ف *f*, ع *E*, ل *l*) in the pattern. The new word then look up in the root lexicon to check if this word constitute a valid Arabic root or not. If the word found in root lexicon, then the original testing word belongs to verb class, otherwise it belongs to noun class. For example, the root of the word نموهَا is نمو (see figure 6.11). The word نمو is not a valid Arabic root. So, the original testing word نموهَا belongs to noun class.

Compiling a lexicon contains all the Arabic valid roots is possible, but it needs a time to compile, due to the fact that this problem did not have a noticeable impact on the effectiveness of our tagger performance because the number of words that can be found in testing corpus (i.e 0.0043%) is scarce. In addition, the emergence of this problem came in the final stages of our experiements and out of the scope in this research.

Therefore, it has been left as future work in this research.

The size of the tag set for an annotation system has a direct influence on the accuracy of the tagging system. A smaller tag set may contribute to increase the performance of the tagger. But, using a smaller tag set means providing a less linguistic information making the whole tagging system less useful for linguistic and NLP developers (i.e. to build an educational system), especially if the aim of the tagger is to produce a tagged corpus.

Out of the overall tagged corpus tokens, 68% of the tags were detailed tags while 32% are general tags as shown in figure 6.12. Since all the tags in pattern lexicon are deatiled tags, each token is tagged by applying the pattern-based rules definitely assigned a detailed tag. In addition, most of the tags designed with lexical and contextual rules are also detailed tags. Most of the 32% of general tags included one or sometimes two inflectional features. In other word, most of the tags designed with lexical and contextual rules, are attached with inflectional features such as, mood, state or case. For example, **NuCnId** *(Common noun, Indefinite)*. Such a tag has been calculated with 32% as general tag. However, we tend to enrich each word in the testing corpus with a detailed tag.

All POS tagging systems were built for Arabic (described in chapter 2) share the following characteristics (1) they deal with unvocalised Arabic text (2) they need a manually tagged corpus. The current tagger system deals with partially-vocalised Arabic text without using a manually tagged corpus. In addition, the current tagger assigs the tag to the testing word based on the pattern of that word instead of the word itself. Despite that the current tagger is uses a different technique and a different type of text, we still would like to compare the results obtained from the current tagger (AMT) with the results of Khoja tagger (APT). Unfortunately, the source code of Khoja tagger is not available on her site[6]. In addition, we had no luck in contacting her to acquire the source code for her tagger.

A sample of 1500 words have been taken randomly from the above three test sets, the last diacritical mark removed from the words, in other word, the text become un-vocalised. An experiment was performed to tag this sample, and the result is shown in figure 6.13.

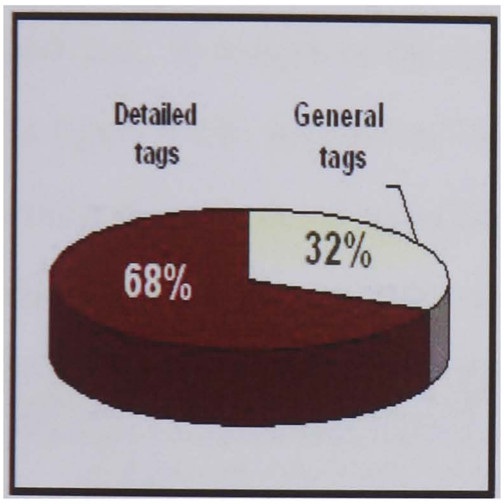


Figure 6.12: Detailed and general tag ratio overall in the correctly tagged corpus

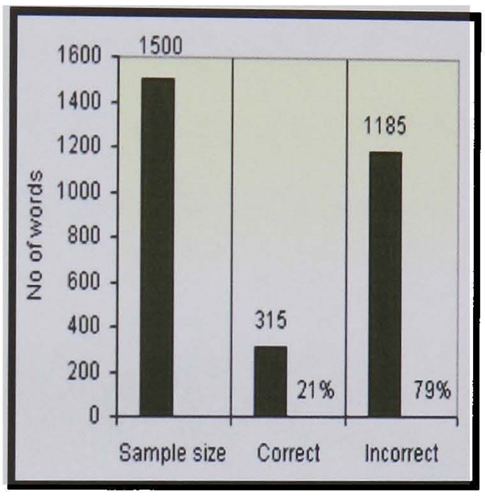


Figure 6.13: Success rate for un-vocalised sample text which contains 1500 words

The AMT correctly tagged 21% of the unvocalised sample text. Most of the correctly tagged tokens in this sample belong to particle and punctuation marks and some proper nouns. It is not a surprise result since the patterns as well as the lexical and contextual rules examined the last diacritical mark during the tagging process.

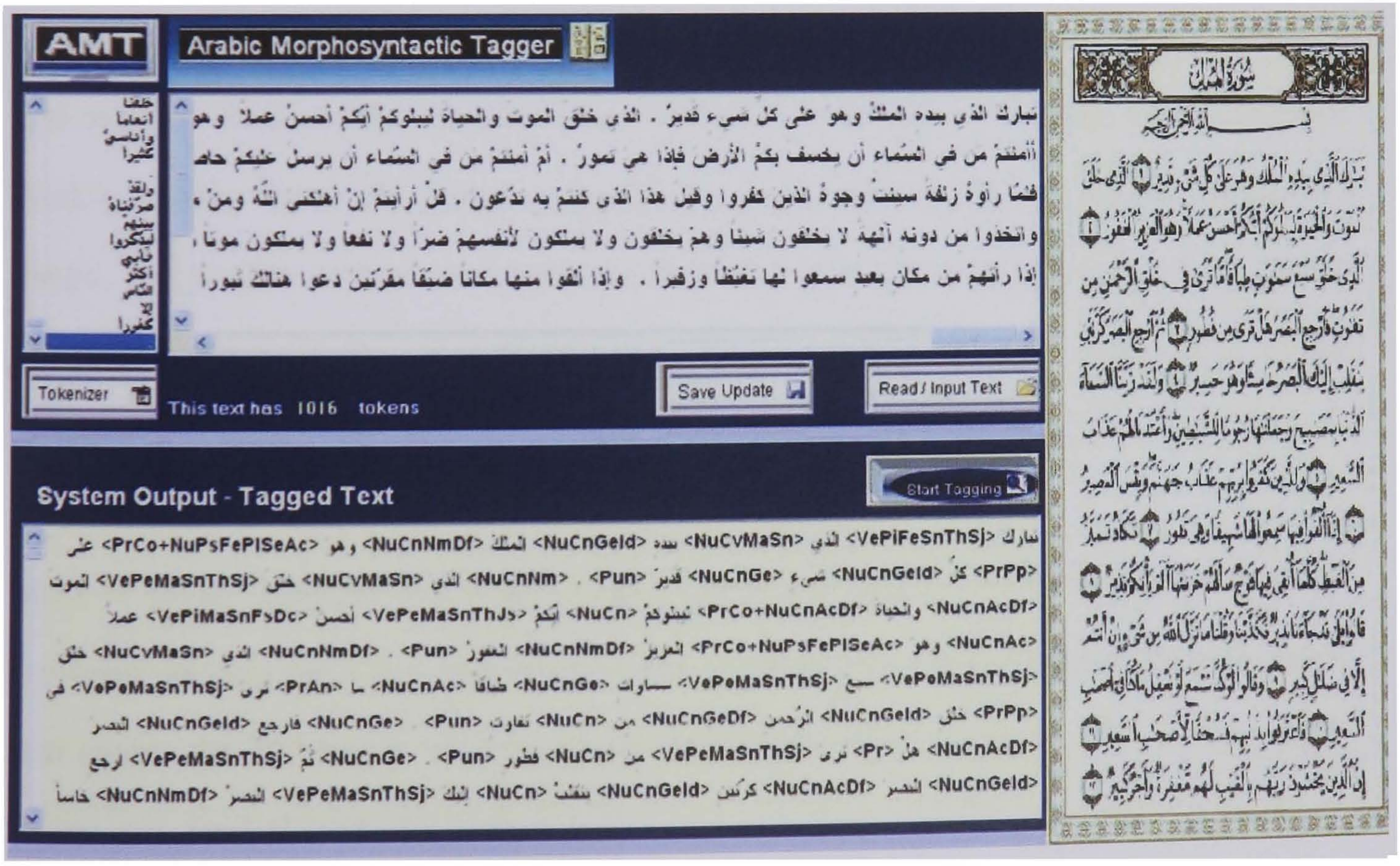
In addition, the result of experiment-4 shows the importance of the last diacritical mark in reducing the lexical ambiguity and providing the semantic information to the

[6] http://zeus.cs.pacificu.edu/shereen/

word which helping the POS tagger to determine the correct tag of each word in the testing text. AMT correctly tagged 91% of testing words. Since the majority of Arabic words are noun words, a default tag, that is, **NuCn** (Common noun) is assigned to the remaining words (9%). These words are stored in a special list and were reviewed. The deafult tag is correct for most of the remaining words, and is reduced the ratio of these words to 3% which are manullay tagged.

6.3.1 The Quran text experiment

Another experiment was performed to get a picture of the tagger accuracy score in different text. A sample of the Quran text was taken from chapter Almulk and Alforqan (see figure 6.14), it contains 1016 words. The diacritics were removed except the last diacritical mark. A set contains 1016 words was taken from the Quran. The diacritics were removed except the last diacritical mark.



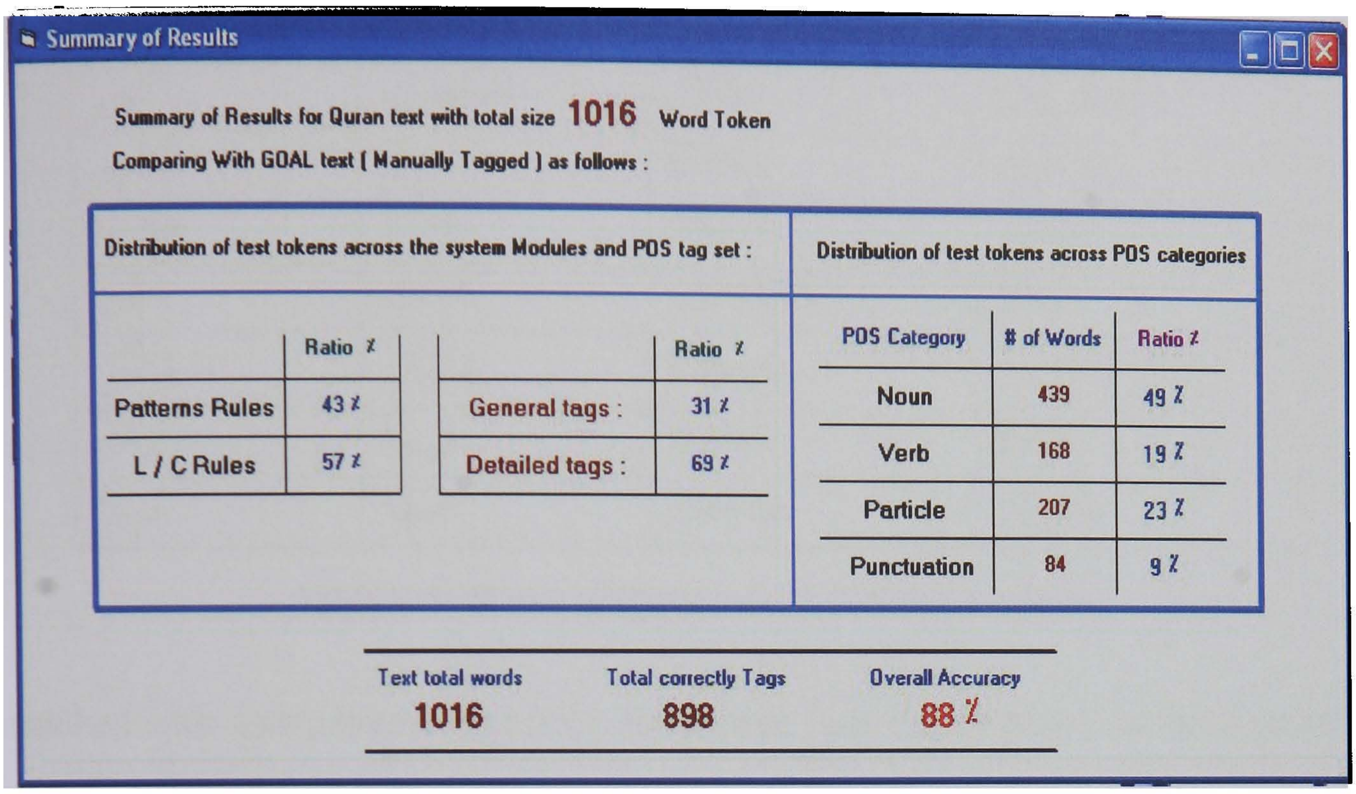
خلقه
أنعاما
وأناسي
كثيرا
ولقد
صرفناه
بينهم
ليذكروا
فأبى
أكثر
الناس
إلا
كفورا

تبارك الذي بيده الملكُ وهو على كل شيء قديرٌ . الذي خلق الموت والحياة ليبلوكمْ أيكمْ أحسنُ عملا وهو
أأمنتمْ من في السَّماء أن يخسف بكمْ الأرض فإذا هي تمورُ . أمْ أمنتمْ من في السَّماء أن يرسل عليكمْ حاصِ
فلمّا رأوهُ زلفة سيئتْ وجوهُ الذين كفروا وقيل هذا الذي كنتمْ به تدّعون . قلْ أرأيتمْ إنْ أهلكني اللَّهُ ومنْ ،
واتخذوا من دونه آلهة لا يخلقون شيئا وهمْ يخلقون ولا يملكون لأنفسهمْ ضرّاً ولا نفعا ولا يملكون موتا ،
إذا رأتهمْ من مكان بعيد سمعوا لها تغيّظا وزفيرا . وإذا ألقوا منها مكانا ضيّقا مقرّنين دعوا هنالك ثبورا

This text has 1016 tokens

```
تبارك <VePiFeSnThSj> الذي <NuCvMaSn> بيده <NuCnGeld> الملكُ <NuCnNmDf> وهو <PrCo+NuPsFePlSeAc> على
<PrPp> كلّ <NuCnGeld> شيء <NuCnGe> قديرٌ <Pun> . <NuCnNm> الذي <NuCvMaSn> خلق <VePeMaSnThSj> الموت
<NuCnAcDf> والحياة <PrCo+NuCnAcDf> ليبلوكمْ <NuCn> أيكمْ <VePeMaSnThJs> أحسنُ <VePiMaSnFsDc> عملا
<NuCnAc> وهو <PrCo+NuPsFePlSeAc> العزيزُ <NuCnNmDf> الغفورُ <Pun> . <NuCnNmDf> الذي <NuCvMaSn> خلق
<VePeMaSnThSj> سبع <VePeMaSnThSj> سماوات <NuCnGe> طباقا <NuCnAc> ما <PrAn> ترى <VePeMaSnThSj> في
<PrPp> خلق <NuCnGeld> الرّحمن <NuCnGeDf> من <NuCn> تفاوت <Pun> <NuCnGe> فارجع <NuCnGeld> البصر
<NuCnAcDf> هلْ <Pr> ترى <VePeMaSnThSj> من <NuCn> فطور <Pun> . <NuCnGe> ثمّ <VePeMaSnThSj> ارجع
<NuCnGeld> البصر <NuCnAcDf> كرّتين <NuCnGeld> ينقلبْ <NuCn> إليك <VePeMaSnThSj> البصرُ <NuCnNmDf> خاسئا
```

Figure 6.14: A sample of Quran text



Summary of Results

Summary of Results for Quran text with total size 1016 Word Token
Comparing With GOAL text (Manually Tagged) as follows :

Distribution of test tokens across the system Modules and POS tag set :

	Ratio %
Patterns Rules	43 %
L / C Rules	57 %

	Ratio %
General tags :	31 %
Detailed tags :	69 %

Distribution of test tokens across POS categories

POS Category	# of Words	Ratio %
Noun	439	49 %
Verb	168	19 %
Particle	207	23 %
Punctuation	84	9 %

Text total words	Total correctly Tags	Overall Accuracy
1016	898	88 %

Figure 6.15: The result of the Quran text

The AMT system correctly tagged 88% of the Quran sample as shown in figure 6.15. Out of the tokens tagged correctly, 43% of correctly tagged tokens are achieved by applying pattern-based module while 57% are achieved by applying lexical and contextual module. Some of the sample words in this experiment (experiment-5) are classical Arabic words. Since the Modern Standard Arabic (MSA) text is used in the current usage, the Arabic writers are used the meaning of these words instead of using the classical Arabic words. A sample of classical words which have been used in the Quran text and their meaning in MSA text can be seen in Table 6.1.

Most of these classical words are tagged incorrectly due to the fact that the patterns are valid patterns for MSA text rather than Classical text. On the other hand, the Quarn text shares the MSA text in some errors described above. For example, some proper nouns are used also in the Quran text, such as موسَى, ابرَاهيم, نوح and عيسَى. Each of these proper nouns do not have a pattern to follow. Also, some nouns are wrongly

Quran word	MSA word	transliteration	translation
تمور	تضطرب	tDTrb	disturbed
فطور	شقوق	shqwq	creases
حَاصبَا	حجَارة	HjArP	stones
لجوَا	تمَادوَا	tmAdwA	gone
ذرأكم	خلقكم	khlqkm	created you
زلفة	قريبَا	qrybA	soon
غورَا	عميقَا	Emyqa	deeply

Table 6.1: Some of Quran words VS MSA words

matched with verb patterns especially the pattern فَعل. Furthermore, the same problem was appeared during the Quran text experiment relates to nouns end with long vowel ا.

Despite the errors described above, the AMT has achieved very good accuracy in the Quran text. Figure 6.15 shows that most of the Quran text are similar to MSA text in regarding to POS classes they belonged to. For example, most of the Quran words are belonged to nouns and particles (72%) rather than verb words. In addition, 43% of the Quran sample words have been tagged using pattern rules. This is a nature ratio since the Quran words are words derived from the root and most of the Quran words have patterns to follow. At the same time, 69% of the sample words are tagged by deatiled tags (see figure 6.15).

6.4 Summary of results obtained from the AMT system

The summary of all the results obtained from the AMT system for all the experiments described above show that the correctly tagged words vary according to the domain of each text. The AMT system achieved very good accuracy due to the fact that it does not used a lexicon for training (91%).

Since there is no a huge tagged corpus available to the tagger system presented in this work, this accuracy enables us to point out that it is possible to build a tagger system for Arabic that did not require a huge tagged corpus. Such this tagger helps to solve the problem of the lack of a huge tagged corpus for Arabic in the current literature. In addition, the diacritical mark especially at the last letter of the word plays a great role in reducing the lexical ambiguity and determining the correct POS tag to each word in testing corpus. Despite that the tagger system presented in this work has many strength points (described in next chapter, section 7.4), the problems that have been faced during all experiments can be summarised as follow :

- **The system does not accurate in tagging proper nouns and Arabised words.**
- **The system does not differentiate between the passive and active perfect verb and assign a detailed tag to both words.**
- **Some nouns are wrongly matched with verb patterns.**

As mentioned earlier in section 6.3, the above shortcomings did not have a noticeable impact on the effectiveness of the tagger performance because the number of words that can be found in testing corpus is scarce. For example, 0.0005% of testing words are passive perfect verb and 0.0043% relates to nouns end with long vowel ا, *Alif*. However, solving these shortcomings to enhance the performance of AMT tagger are taken into account and described in next chapter (section 7.3).

6.5 Chapter Summary

This chapter presented several experiments have been done to evaluate the AMT system using a new partially-vocalised Arabic testing Corpus. The description of the data sets were used during the experiments is shown. The result of the experiments and the analysis of these results are also explained. The results show that AMT is achieved an average accuracy 91% of the testing corpus which contains 20,000 words. The shortcomings that AMT system has also mentioned. The main conclusion yielded during the course of this research, the strenght points that the tagger system has, and future work are described in next chapter.

Chapter 7

Conclusion

Several Part-of-Speech tagging systems with high tagging accuracy have been developed, especially for English based on text statistics or on grammar rules. Unlike English, the Part-of-Speech tagging systems for Arabic as a research field in Arabic NLP is relatively reviewed. A few systems have been developed in Part-of-Speech tagging for Arabic. These systems were built to tag unvocalised Arabic text using a lexicon or dictionary that was tagged manually and used as a training corpus containing all possible tags (lexical information) for each word.

The Arabic language has a valuable and an important feature, called *diacritics*, which are marks placed over and below the characters of the word. An Arabic text may be written with diacritics or without. An Arabic text that appears without a short vowel and diacritics is called unvocalised text while written Arabic text with full representation of short vowels and other diacritics marks is called fully-vocalised text. An Arabic text is a partially-vocalised text when the the diacritical marks assigned to one or maximum two letters in the word.

This thesis represents a substantial starting point for developing a rule-based part-of-speech tagging system deals with partially-vocalised Arabic text. It is the first tagger (1) uses only linguistic rules, (2) investigate the role of the last diacritical mark in help to determine the correct POS tag to each word in testing corpus. The main function of the tagger system is to produce a POS tagged corpus.

A novel technique: pattern-based, has been explored using a novel algorithm (pattern-matching algorithm). In this technique, the Arabic word was tagged based on its pattern. A lexicon of patterns which are associated with the last diacritical mark was generated automatically and used instead of a huge Arabic word lexicon. The advantages of this technique are twofold: First, it does not need a lexicon or training corpus. Second, it reduces the space since hundreds of Arabic words may follow one pattern. Additionally, a set of linguistic rules (lexical and contextual technique) based on the character(s), affixes, the last diacritical mark, the word itself, and the surrounding words or on the tags of the surrounding words were used to tag those words not tagged by pattern-based technique.

The system developed to answer hypothesis and research questions mentioned in chapter 1. Since the accuracy of the AMT system that can be achieved is 91%. This enables us to make the following assertions :

1. **it is possible to build a tagger system for Arabic with out needs a huge lexicon for training.**
2. **the diacritical mark especially at the last letter of the word plays a great role in reducing the lexical ambiguity and determining the correct POS tag to each word in testing corpus.**

3. **the accuracy is comparable to that of statistics-based tagging systems were built for Arabic. But these systems deal with unvocalised text and need a huge manually tagged lexicon which still not available in the current literature and most of the current taggers were used a small training corpus.**

Section 7.1 summarise the importance of diacritic feature. Section 7.2 describe the contributions of this research while section 7.3 point out a direction for future works.

7.1 Importance of diacritic feature

The lack of diacritics in Arabic texts is presented as a major challenge to most Arabic NLP tasks. The use of diacritics in Arabic texts are extremely important. The list below summarises the importance of using diacritics in Arabic language :

1. They add a semantic information to words which helps with resolving ambiguity in the meaning of words.
2. They help determining the correct POS tag to the words in the sentence.
3. They ascribe grammatical functions to the words, differentiating the word from other words, and determining the syntactic position of the word in the sentence.
4. Indicating the correct pronunciation of words, correct syntactical analysis which leads to reducing problems for NLP applications such as text-to-speech or speech-to-text, and removing the semantical confusion of Arabic readers.

In addition, the last diacritical mark helps not only in determining the correct part-of-speech of the words in the sentence, but also in providing full information regarding the inflectional features for the sentence words.

7.2 Contributions

The contributions of this research to the field of NLP can be summarise as follow :

1. **AMT: Arabic Morphosyntactic Tagger**

   This research has developed a POS tagger system called AMT (short for Arabic Morphosyntactic Tagger). AMT deals for the first time with partially-vocalised Arabic text . The main aim of AMT is to annotate the testing corpus by adding POS tag or label to each word in the testing corpus and toproduce a POS tagged partially-vocalised Arabic text. It can also used as a prerequisite tool for many NLP tasks, such as, parsing and informational retrieval systems.

2. **A new tag set for Arabic**

   A new morphosyntactic tag set that is derived from the ancient Arabic grammar has been developed, which is based on the Arabic system of inflectional morphology. The tag set does not follow the traditional Indo-European tag set that is based on Latin but instead it's based on the semitic tradition of analysing language. These tags contain a large amount of information and add more linguistic attributes to the word. The Arabic tag set contains 161 detailed tags and 28 general tags covering an Arabic major POS classes and sub-classes which have been compiled and introduced in Chapter 4 in this work.

3. **Partially-vocalised Arabic corpus**

   A new partially-vocalised Arabic corpus that contains 20,000 Arabic words chosen and extracted from different books for different levels of school classes has been compiled in this work and introduced in chapter 6. The corpus was tagged using AMT system presented in this research. It will be available (both raw and tagged) freely for public.

4. **Pattern-based technique**

   AMT is rule-based system. It has two rule components. The first component is the pattern-based rule. The trigger in pattern-based technique depends on the pattern of the testing word. These patterns are associated with the last diacritical mark and generated automatically. A novel algorithm (Pattern-Matching Algorithm) has been designed and built in this work and introduced in chapter 5. The aim of this algorithm is to match the inflected word in the testing corpus with its pattern in the pattern lexicon. The second component is the lexical and contextual rule. The trigger in the contextual rules depends on the current word itself, the tags or words on the context of the current word, while the trigger in the lexical rules depends on the character(s), affixes, and diacritics of a word.

7.3 Future Works

During the course of this research, there are many areas that deserve more study, these areas can be summarised as follows:

- Further research in the expansion of pattern lexicon to contains all Arabic patterns.
- Improving the tagger by defining and encoding an additional set of Arabic tagging rules.
- Encode the testing corpus in SGML marks.
- Produce the output in a standard format (e.g XML).
- Evaluate the tagger and compare its result with other tagger(s) deal with partially-vocalised Arabic text.

- Building a lexicon contains all Arabic roots to enhance the performance of the tagger system and pattern-match algorithm as well.

7.4 Summary

In conclusion, all the POS tagging systems for Arabic described in this work (see chapter 2) were built to tag unvocalised Arabic text. AMT system presented in this work is different from the described systems in the following aspects :

- **It is the first tagger deals with partially-vocalised Arabic text.**
- **It is the first tagger uses purely rule-based approach, applied a novel technique, that is,** ***pattern-based technique.*** **The tag assigned to the word based on the pattern of that word instead of the word itself.**
- **It does not need a lexicon (manually tagged or untagged) for training.**
- **It is the first tagger investigated the role of diacritic feature in the Arabic language.**

An overall ambiguity in vocalised Arabic text seems to be lower than in an unvocalised text. The last diacritical mark plays a great role to remove a great deal of lexical ambiguity when the text at least is partially-vocalised.

Bibliography

[1] Al-hayat corpus can be found at :. http://www.elda.org/catalogue/en/text/W0030.html.

[2] An-nahar newspaper text corpus can be found at :. http://www.elda.org/catalogue/en/text/W0027.html.

[3] Arabic newswire corpus can be found at :. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2001T55.

[4] Buckwalter arabic corpus can be found at :. http://www.qamus.org.

[5] Buckwalter arabic morphological analyzer (online). http://students.cs.byu.edu/~jonsafar/buckwalter.html.

[6] Nijmegen corpus can be found at :. http://www.let.kun.nl/wba/Content2/1.4.5_Nijmegen_Corpus.html.

[7] Penn arabic treebank corpus can be found at:. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2005T20.

[8] Useful annotated list of arabic corpora can be found at:. www.comp.leeds.ac.uk/eric/latifa/arabic_corpora.html.

[9] A useful resource for corpora can be found in :. http://www.essex.ac.uk/linguistics/clmt/w3c/corpus_ling/content/introduction3.html.

[10] Useful resources for corpus-based computational linguistics can be found at :. http://www.essex.ac.uk/linguistics/clmt/w3c/corpus_ling/content/corpora/list/index2.html\#languages and http://www.athel.com/corpora.html.

[11] Buckwalter arabic morphological analyzer. version 1 (2002) : http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002L49. vesrion 2 (2004) :http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004L02., 2002.

[12] Flor Aarts. Relative who and whom: Prescriptive rules and linguistic reality. *Journal Information for American Speech*, 69(1):71–79, 1994.

[13] Steven Abney. *Part-of-Speech Tagging and Partial Parsing*. S. Young and G. Bloothooft (eds.) Corpus-Based Methods in Language and Speech Processing. An ELSNET book. Kluwer Academic Publisher, Dordrecht, 1997. http://www.sfs.nphil.uni-tuebingen.de/.

[14] Abuleil, Alsamara, and Evens. Acquisition system for arabic noun morphology. In *Proceedings of the Computational Approaches to Semitic Languages Workshop, University of Pennsylvania.*, 2002.

[15] S. Abuleil and M Evens. Discovering lexical information by tagging arabic newspaper text. In *Proceedings of the workshop on Semitic Language Processing. COLING-ACL.98, University of Montreal, Montreal, PQ, Canada.*, pages 1–7, 1998.

[16] S AbuRabia and J Awwad. Morphological structures in visual word recognition: the case of arabic. *Journal of Research in Reading*, 27(ISSN 0141-0423):321336, 2004.

[17] Alfred V. Aho and Jeffrey D. Ullman. *The theory of parsing, translation, and compiling*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1972.

[18] Latifa Al-Sulaiti and Eric Atwell. The design of a corpus of contemporary arabic. *International Journal of Corpus Linguistics*, 11:135–171, 2006.

[19] Ahmed Al-Tarouti. *Temporality in Arabic grammar and discourse*. PhD thesis, University of California, 1991.

[20] Ali Alhamad. In the arabic vocalbulary. *Arabization Journal* *http://www.acatap.htmlplanet.com/jounal.htm*, (20), Dec 2000.

[21] Reima Aljarf. Egnlish and arabic infletions for translation students. http://docs.ksu.edu.sa/PDF/Articles29/Article290045.pdf. Technical report, King Saud University, Saudi Arabia., 2007.

[22] Mohammed Aljlayl and Ophir Frieder. On arabic search: improving the retrieval effectiveness via a light stemming approach. In *CIKM*, pages 340–347. ACM, 2002.

[23] James Allen. *Natural Language Understanding*. Benjamin-Cummings. Menlo Park, California., 2nd edition, 1995.

[24] Shihadeh Alqrainy and Aladdin Ayesh. Developing a tagset for automated pos tagging in arabic. *WSEAS TRANSACTIONS on COMPUTERS*, 5(11):2787–2792, 2006.

[25] Gisle Andersen and Anna-Brita Stenstrom. Colt: a progress report. *ICAME Journal.*, 20:133–136, 1996.

[26] Andras Kocsor Andras Kuba, Laszlo Felfoldi. Pos tagger combinations on hungarian text. In *The Second International Joint Conference on Natural Language Processing (IJCNLP-05), Korea.*, 2005.

[27] Chinatsu Aone and Kevin Hausman. Unsupervised learning of a rule-based spanish part of speech tagger. In *COLING*, pages 53–58, 1996.

[28] Eric. Atwell. Lob corpus tagging project: Post-edit handbook. Department of Linguistics and Modern English Language, University of Lancaster. http://www.comp.leeds.ac.uk/amalgam/tagsets/lob.html, 1982.

[29] Eric. Atwell. Grammatical analysis of scribe: Spoken corpus recordings in british english. SERC Advanced Research Fellowship proposal, Science and Engineering Research Council., 1989.

[30] Eric Atwell. Development of tag sets for part-of-speech tagging. In Anke Ludeling & Merja Kyto, editor, *Corpus Linguistics: An International Handbook.* Mouton de Gruyte, 2007.

[31] Eric Atwell, John Hughes, and Clive Souter. Amalgam: Automatic mapping among lexicogrammatical annotation models. In J Klavans, editor, *The Balancing Act: Combining Symbolic and Statistical Approaches to Language - Proceedings of the ACL Workshop, Association for Computational Linguistics.*, pages pp. 21–28, 1994.

[32] L. R. Bahl and R. L. Mercer. Part-of-speech assignment by a statistical decision algorithm. In *IEEE International Symposium on Information Theory-Ronneby-Sweden.*, pages 88–89, 1976.

[33] Baker, Franz, and Jordan. Coping with ambiguity in knowledge-based natural language analysis. In *the 8th International FLAIRS Conference, USA.*, 1994.

[34] Raffaella Bernardi, Andrea Bolognesi, Corrado Seidenari, and Fabio Tamburini. Pos tagset design for italian. In *In Proc. 5th International Conference on Language Resources and Evaluation - LREC 2006, Genova.*, pages 1396–1401, 2006.

[35] Thorsten Brants. Tnt a statistical part-ofspeech tagger. In *Proceedings of the 6th Applied NLP Conference, ANLP-2000, April.*, 2000.

[36] Eric Brill. A simple rule-based part of speech tagger. In *ANLP*, pages 152–155, 1992.

[37] Eric Brill. Unsupervised learning of disambiguation rules for part of speech tagging. In David Yarovsky and Kenneth Church, editors, *Proceedings of the Third Workshop on Very Large Corpora*, pages 1–13, Somerset, New Jersey, 1995. Association for Computational Linguistics, Association for Computational Linguistics.

[38] Benny Brodda. Problems with tagging and a solution. *Nordic Journal of Linguistics*, pages 93–116, 1982.

[39] Tim Buckwalter. Buckwalter arabic morphological analyser. Linguistic Data Consortium, Philadelphia. `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002L49`, 2002.

[40] Ceccato, Kiyavitskaya, Zeni, Mich, and Berry. Ambiguity identification and measurement in natural language texts. Technical Report DIT-04-111, Informatica e Telecomunicazioni, University of Trento., 2004.

[41] Jean-Pierre Chanod and Pasi Tapanainen. Creating a tagset, lexicon and guesser for a french tagger. In ACL SIGDAT Workshop on Prom Texts to Tags: Issues in Multilingual Language Analysis , University college - Dublin - Ireland., 1995.

[42] Jean-Pierre Chanod and Pasi Tapanainen. Tagging french - comparing a statistical and a constraint-based method. In *EACL*, pages 149–156, 1995.

[43] Gerald Chao. *A probabilistic, Intergative Approach for Improved Natural Language Disambiguiation.* PhD thesis, Departement of Computer Science, University of California, Los Angeles., 2003.

[44] Kenneth W. Church. Current practice in part of speech tagging and suggestions for the future. In Simmons (ed), Sbornik Praci : In Honor of Henry Kucara, Michigan Salvic Studies. 13-48. Michigan., 1992.

[45] Kenneth Ward Church. A stochastic parts program and noun phrase parser for unrestricted text. In *ANLP*, pages 136–143, 1988.

[46] Jan Cloeren. Towards a cross-linguistic tagset. In *In Proceedings of the Workshop on Very Large. Corpora (WVLC), Columbus, Ohio.*, pages 30–39, 1993.

[47] Cutting, Kupiec, Pederson, and Sibun. A practical part-of-speech tagger,. In *Proceedings of the Third Conference on Applied Natural Language Processing, Trento,Italy.*, 1992.

[48] Walter Daelemans, Jakub Zavrel, Peter Berck, and Steven Gillis. Mbt: Memory-based part of speech tagger-generator. *CoRR*, cmp-lg/9607012, 1996.

[49] Aniket Dalal, Kumar Nagaraj, Uma Sawant, Sandeep Shelke, and Pushpak Bhattacharyya. Building feature rich pos tagger for morphologically rich languages: Experiences in hindi. In *ICON-2007: 5th INTERNATIONAL CONFERENCE ON NATURAL LANGUAGE PROCESSING, Hyderabad, India.*, 2007.

[50] James H.Martin. Daniel Jurafsky. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition.* prentice-hall, USA., 2000.

[51] Mona Diab, Kadri Hacioglu, and Daniel Jurafsky. Automatic tagging of arabic text: From raw text to base phrase chunks. In *Proceedings of HLT-NAACL.*, 2004.

[52] Kevin Duh and Katrin Kirchhoff. Pos tagging of dialectal arabic: A minimally supervised approach. In *Proceeding of ACL-05. Computational Approaches. to Semitic Languages. Workshop Proceedings.University of Michigan. Ann Arbor, Michigan, USA*, 2005.

[53] Dzeroski, Erjavec, and Zavrel. Morphosyntactic tagging of slovene: Evaluating taggers and tagsets. `http://citeseer.ist.psu.edu/437793.html;http://nl.ijs.si/et/Bib/LREC00/lrec-tag.ps`, 2000.

[54] El-Kareh and Al-Ansary. An arabic interactive multi-feature pos tagger. In *Proceeding of the international conference on Artificial and Computational intelliegence for Decision Control and Automation in engineering and Industrial Application (ACIDCA) conference, Tunisia.*, pages 83–88, 2000.

[55] Hashish M El-Sadany T. An arabic morphological system. *IBM SYSTEMS JOURNAL*, 28:600–612, 1999.

[56] M Elaraby. Alarge scale computational processor of the arabic morphology and application. Master's thesis, Cairo University, Egypt, 2000.

[57] Antonie Eldahdah. *A dictionary of Arabic grammar in charts and tables*. Number 01D110410. Librairie du liban publishers, 9th edition, 2002.

[58] Ayman Elnaggar. A phrase structure grammar of the arabic language. *COLING*, pages 342–344, 1990.

[59] David Elworthy. Tagset design and inflected languages. *CoRR*, cmp-lg/9504002, 1995.

[60] N. W. Francis and H. Kucera. Brown corpus manual of information: to accompany a standard corpus of present-day edited american english, for use with digital computers. Providence, R.I.: Department of Linguistics, Brown University. `http://www.comp.leeds.ac.uk/amalgam/tagsets/brown.htmll`, 1979.

[61] Anis Frayha. *Essentials of Arabic : A Manual for teaching classical and colloquial Arabic*. American university of Beirut, 1953.

[62] A Freeman. Brills pos tagger and a morphology parser for arabic. In *Proceedings of the Arabic Language Processing: Status and Prospects Workshop at the 39th Annual Meeting of the Association of Computational Linguists, Toulouse, France*, page 148154, 2001.

[63] R. Garaside and N. Smith. *A Hybrid Grammatical Tagger : CLAWS4, in Garside, Leech, and McEnery.*, chapter 7, pages 102–122. Longman, London., 1997.

[64] R Garside. The claws word-tagging system. In: R. Garside, G. Leech and G. Sampson (eds), The Computational Analysis of English: A Corpus-based Approach. London: Longman., 1987.

[65] Roger. Garside. The robust tagging of unrestricted text: the bnc experience. In Jenny Thomas and Mick Short (eds) Using corpora for language research: studies in the honour of Geoffrey Leech, p167-180. London: Longman., 1996.

[66] G Gazadr and C Mellish. *Natural Language Processing in LISP*. Addison-Wesley, Reading, Massachuestts., 1989.

[67] Sidney Greenbaum. The tagset for the international corpus of english. In Clive Souter and Eric Atwell (eds) Corpus-based Computational Linguistics. pp11-24. Amsterdam: Rodopi., 1993.

[68] B. B. Greene and G. M. Rubin. Automatic grammatical tagging of english. Technical Report, Department of Linguistics, Brown University, 1971.

[69] Ihab Joseph Griess. *Syntactical Comparsion Between Classical Hebrew and Classical Arabic Based On The Translation Of Mohammad 'Id's Arabic Grammar*. PhD thesis, The Southern Baptist Theological Seminary, louisville,Kentucky, USA., 2006.

[70] Marshall G.S.Hodgson. *The Venture of Islam*. Number ISBN : 0226476936. University of chicago press, 1974.

[71] Nizar Habash and Owen Rambow. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *ACL*. The Association for Computer Linguistics, 2005.

[72] Saeed Raheel Haidar M. Harmanani, Walid T. Keirouz. A rule-based eextensible stemmer for information retrieval with application to arabic. In *Proceedings of the Eighth IASTED International conference , Marbella , Spain .*, 2004.

[73] Hans Van Halteren. *Syntactic Wordclass Tagging*, volume 9. Kluwer Academic Publishers, Netherlands., 1999.

[74] Andrew Hardie. Developing a tagset for automated part-of-speech tagging in urdu. In *Proceedings of the Corpus Linguistics 2003 conference, Lancaster University, UK*, 2003.

[75] Harmain M. Harmain. Arabic part-of-speech tagging. In *The Fifth Annual U.A.E. University Research Conference, Al-Ain, U.A.E.*, 2006.

[76] Haywood and Nahmad. *A new arabic grammar: of the written language.* LUND HUMPHRIES , USA, 2005.

[77] Donald Hindle. Acquiring disambiguation rules from text. In *ACL*, pages 118–125, 1989.

[78] Barbora Hladka and Kiril Ribarov. Part-of-speech tags for automatic tagging and syntactic structures. *Issues of Valency and Meaning. Studies in Honour of Jarmila Panevov, Karolinum, Charles University Press, Prague, Czech Republic.*, pages 226–240, 1998.

[79] Barbora Hladka Jan Hajic. Czech language processing - pos tagging. In *In Proceedings of the First International Conference on Language Resources and Evaluation, Granada, Spain.*, pages 931–936, 1998.

[80] A.P. Hendrikse Jens Allwood, Leif Grnqvist. Developing a tagset and tagger for the african languages of south africa with special reference to xhosa. *Southern African Linguistics and Applied Language Studies*, 21(4):223–237, 2003.

[81] Mark R. Titchener Jim Yaghi. T-code compression for arabic computational morphology. In *Proceedings of the Australasian Language Technology Workshop. Melbourne.*, 2003.

[82] Johansson, Atwell, Garside, and Leech. The tagged lob corpus users manual. Bergen: Norwegian Computing Centre for the Humanities., 1986.

[83] Pavel SMRZ Karel PALA. Building czech wordnet. *ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY*, 7(1-2):79–88, 2004.

[84] Vangelis Karkaletsis, Constantine D. Spyropoulos, and George Petasis. Named entity recognition from greek texts: the GIE project. 1998.

[85] Fred Karlsson. Constraint grammar as a framework for parsing running text. In *Proceedings of the 13th International Conference on Computational Linguistics, Helsinki.*, volume 3, pages 168–173, 1990.

[86] Yasuhiro Kawata. *Tagsets for Morphosyntactic Corpus Annotation: the idea of a reference tagset for Japanese.* PhD thesis, Department of Language and Linguistics - University of Essex., 2005.

[87] Shereen Khoja. Apt: Arabic part-of-speech tagger. In *Proceedings of the Student Workshop at the Second Meeting of (NAACL2001), Carnegie Mellon University, Pittsburgh, Pennsylvania.*, 2001.

[88] Shereen Khoja. *APT: an Automatic Arabic Part-of-speech Tagger.* PhD thesis, Ph.D. thesis, Lancaster University., 2003.

[89] Khojah, Graside, and Knowels. A tagset for the morphosyntactic tagging of arabic. In *presented at Corpus Linguistics 2001, Lancaster University, UK.*, 2001.

[90] Sheldon Klein and Robert F. Simmons. A computational approach to grammatical coding of English words. *Journal of the ACM*, 10(3):334–347, July 1963.

[91] J. Kupiec. Robust part-of-speech tagging using a hidden markov model. *Computer Speech and Language*, 6, 1992.

[92] G. Leech. 100 million words of english. *English Today*, 9:9–15, 1993.

[93] G. Leech, R. Garside, and E. Atwell. The automatic grammatical tagging of the lob corpus. *ICAME News*, 7:13–33, 1983.

[94] Wolfgang Lezius, Reinhard Rapp, and Manfred Wettler. A morphology-system and part-of-speech tagger for german. *CoRR*, cmp-lg/9610006, 1996.

[95] Ann Bies Maamouri and Seth Kulick. Diacritization: A challenge to arabic treebank annotation and parsing. In *In Proceedings of the British Computer Society Arabic NLP/MT Conference, London, UK*, 2006.

[96] J Mace. *Arabic verbs and essential grammar.* Hodder/Stoughton. London, 1999.

[97] H. E. Mahgoub, M. A. Hashish, and Ahmed Taher Hassanein. A matrix representation of the inflectional forms of arabic words: A study of co-occurrence patterns. In *Proceeding of COLING*, pages 419–421, 1990.

[98] Yong Mao. Natural language processing module - pos tagging and sentence parsing - laboratory manual. http://www.csic.cornell.edu/201/natural_language/, 1997.

[99] Marcus, Santorini, and Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics: Special Issue on Using Large Corpora*, 19(2):313–330, 1993.

[100] Marques and Pereira. A neural network approach to part-of-speech tagging. In *Proceedings of the second workshop on spoken and written Portuguese, Curitiba, Brazil*, pages 1–9, 1996.

[101] L. Marquez and H. Rodriguez. Part-of-speech tagging using decision trees. In *In Proceedings of the 10th European Conference on Machine Learning, ECML'98. Chemnitz, Germany*, 1998.

[102] E Marsi, A van den Bosch, and A Soudi. Memory-based morphological analysis generation and part-of-speech tagging of arabic. In *ACL-05. Computational Approaches. to Semitic Languages. Workshop Proceedings.University of Michigan. Ann Arbor, Michigan, USA*, 2005.

[103] A. M. McEnery. *Computaional Linguistics - a bandbook and toolbox for natural language procesiing.* SIGMA PRESS-Wilmslow, United Kingdom., 1992.

[104] Karine Megerdoomian. Developing a persian part-of-speech tagger. In *Proceedings of First Workshop on Persian Language and Computers. Invited talk. Tehran University, Iran.*, 2004.

[105] B Megyesi. Brill's rule-based part of speech tagger for hungarian. Master's thesis, Computational Linguistics, Stockholm University, Sweden., 1998.

[106] B Megyesi. Brill's pos tagger with extended lexical templates for hungarian. ACAI'99, 1999.

[107] M.Elaffendi. An lvq connectionist solution to the non-determinacy problem in arabic morphological analysis : a learning hybrid algorithm. *Natural Language Engineering*, 8(1):3–23, 2001.

[108] Bernard Merialdo. Tagging english text with a probabilistic model. *Computational Linguistics*, 20(2):155–171, 1994.

[109] George A. Miller. The lexical component of natural language processing. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, College Park, Maryland*, pages 21–21. Association for Computational Linguistics Morristown, NJ, USA, 1999.

[110] Shinsuke Mori. A stochastic parser based on an SLM with arboreal context trees. In *COLING*, 2002.

[111] Joakim Nivre. Logic programming tools for probabilistic part-of-speech tagging. Technical report, MSI Report - Vaxjo University - sweede, 2000.

[112] Kemal Oflazer and Ilker Kuru. Tagging and morphological disambiguation of turkish text. In *Proceeding of ANLP*, pages 144–149, 1994.

[113] H. Paulussen and W. Martin. Dilemma-2: A lemmatizer-tagger for medical abstracts. In *Third Conference on Applied Language Processing, Trento, Italy.*, pages 141–146, 1992.

[114] Juan Prez-Ortiz and Mikel Forcada. Part-of-speech tagging with recurrent neural networks. In *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2001.*, pages p. 1588–1592., 2001.

[115] K Prtz. Part-of-speech tagging for swedish. In *Proceeding in Parallel Corpora, Parallel Worlds , University, Sweden.*, pages 201–206, 2002.

[116] Adwait Ratnaparkhi. A maximum entropy model for part-of-speech tagging. In *The empirical methods in Natural Language Processing Conference.*, pages 133–142, 1996.

[117] Steven J. Rose. Grammatical category disambiguation by statistical optimization. *Computational Linguistics*, 14(1):31–39, 1988.

[118] Yoad Winter Roy Bar-Haim, Khalil Sima'an. Part-of-speech tagging of modern hebrew text. *Natural Language Engineering, Cambridge University Press, UK.*, 1, 2006.

[119] Thomas Russi. A syntactic and morphological analyzer for a text-to-speech system. In *COLING*, pages 443–445, 1990.

[120] Karin Ryding. *A Reference Grammar of Modern Standard Arabic.* Cambridge University Press, 2005.

[121] G. Sampson. *English for the Computer: the SUSANNE corpus and analytic scheme.* Oxford: Clarendon Press., 1995.

[122] C. Samuelsson. Morphological tagging based entirely on bayesian inference. In *Proceedings of the 9th Nordic Conference of Computational Linguistics.*, Stockholm, Sweden, 1993.

[123] C. Samuelsson. A novel framework for reductionistic statistical parsing. In *Proceedings of the 4th International Workshop on Parsing Technologies*, pages 208–215, Prague/Karlovy Vary, Czech Republic, 1995.

[124] Beatrice. Santorini. Part-of-speech tagging guidelines for the penn treebank project. Technical Report MS-CIS-90-47, University of Pennsylvania: Department of Computer and Information Science., 1990.

[125] Helmut Schmid. Part-of-speech tagging with neural networks. In *Proceeding of COLING*, pages 172–176, 1994.

[126] Gerold Schneider and Martin Volk. Adding manual constraints and lexical look-up to a brill-tagger for german., 1998.

[127] Fatma Al Shamsi and Ahmed Guessoum. A hidden markov model based pos tagger for arabic. In *JADT 2006 - 8th International Conference on the Statistical Analysis of Textual Data, Fance.*, 2006.

[128] Clive. Souter. A short handbook to the polytechnic of wales corpus. Bergen University, Norway: ICAME, The Norwegian Computing Centre for the Humanities., 1989.

[129] Stolz, Tannenbaum, and Carstensen. A stochastic approach to the grammatical coding of english. *Communications of the ACM*, 8(6):399–405, 1965.

[130] Richard Sutcliffe, Heinz-Detlev Koch, and Annette McElligott (eds.). Industrial parsing of software manuals. Amsterdam: Rodopi., 1996.

[131] J. Svartvik. The london corpus of spoken english: Description and research. Lund: Lund University Press. Lund Studies in English 82., 1990.

[132] Pasi Tapanainen and Atro Voutilainen. Tagging accurately – don't guess if you know. ANLP 94, 1994.

[133] Lolita Taylor and Gerry Knowles. Manual of information to accompany the sec corpus: The machine readable corpus of spoken english. University of Lancaster: Unit for Computer Research on the English Language., 1988.

[134] Kees Versteegh. *The Arabic Language*. Number ISBN-10: 0748614362. Edinburgh University Press, 2001.

[135] Atro Voutilainen. A syntax-based part-of-speech analyser. In *EACL*, pages 157–164, 1995.

[136] Tams Vradi and Csaba Oravecz. Morpho-syntactic ambiguity and tagset design for hungarian, 1999.

[137] R Weishedel, R Scewartz, J Ralmucci, M Meteer, and L Rawshaw. Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19:359–382, 1993.

[138] Wright. *A Grammar of the Arabic Language.* Cambridge University Press, 1988.

[139] Imed Zitouni, Jeffrey S. Sorensen, and Ruhi Sarikaya. Maximum entropy based restoration of arabic diacritics. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, Sydney*, pages 577– 584, 2006.

[140] Muhammad Zughoul. Developing computer based corpora of arabic: A preliminary proposal. In *at the Conference on Situated Languages, Technology and Communication, Institute of studies and Research on Arabicization, Rabat, Morocco.*, 1997.

Appendix A

Tagset Appendices

A.1 General Tags

Tag	Dsecription
VePe	*Perfect verb*
VePi	*Imperfect verb*
VePm	*Imperative verb*
NuPo	*Proper noun*
NuCn	*Common noun*
NuAj	*Adjective noun*
NuIf	*Infinitive noun*
NuRe	*Relative noun*
NuDm	*Diminutive noun*
NuIs	*Instrument noun*
NuPn	*noun of Place*
NuTn	*noun of Time*
NuPs	*Pronoun*
NuCv	*Conjunctive noun*

Tag	Description
NuCd	*Conditional noun*
NuDe	*Demonstrative noun*
NuIn	*Interrogative noun*
NuAd	*Adverb*
NuNn	*Numeral noun*
Fw	*Foreign noun*
Pun	*Punctuation mark*
PrPp	*Preposition*
PrVo	*Vocative Particle*
PrCo	*Conjunction Particle*
PrEx	*Exception Particle*
PrAn	*Annulment Particle*
PrSb	*Subjunctive Particle*
PrJs	*Jussive Particle*

A.2 Detailed Tags

Tag	Dscription	Arabic Example	Transliteration	Translation
VePeMaSnThSj	Verb, Perfect, Masculine, Singular, Third Person, Subjunctive	كتبَ	ktba	He Wrote
VePeMaSnFsDc	Verb, Perfect, Masculine, Singular, First Person, Indicative	كتبتُ	ktbtu	I Wrote
VePeMaSnSeSj	Verb, Perfect, Masculine, Singular, First Person, Subjunctive	كتبتَ	ktbta	You(Sn,Ma) Wrote
VePeFeSnSeJs	Verb, Perfect, Feminine, Singular, Second Person, Jussive	كتبتِ	ktbti	You(Sn,Fe) Wrote
VePeFeSnThJs	Verb, Perfect, Feminine, Singular, Third Person, Jussive	كتبتْ	Ktbtx	She Wrote
VePeNeDuSeSj	Verb, Perfect, Neuter, Dual, Second Person, Subjunctive	كتبتما	ktbtmA	You(Du) Wrote
VePeMaDuThSj	Verb, Perfect, Masculine, Dual, Third Person, Subjunctive	كتبا	ktbA	They(Du,Ma) Wrote
VePeFeDuThSj	Verb, Perfect, Feminine, Dual, Third Person, Subjunctive	كتبتا	ktbtA	They(Du,Fe) Wrote
VePeMaPlFsSj	Verb, Perfect, Masculine, Plural, First Person, Subjunctive	كتبنا	ktbnA	We Wrote
VePeMaPlSeJs	Verb, Perfect, Masculine, Plural, Second Person, Jussive	كتبتُمْ	Ktbtmx	You(Pl,Ma) Wrote
VePeFePlSeJs	Verb, Perfect, Feminine, Plural, Second Person, Subjunctive	كتبتنَّ	Ktbtna	You(Pl,Fe) Wrote
VePeFePlThJs	Verb, Perfect, Feminine, Plural, Third Person, Subjunctive	كتبنَ	Ktbna	They(Pl,Fe) Wrote
VePeMaPlThDc	Verb, Perfect, Masculine, Plural, Third Person, Indicative	كتبوا	ktbwA	They(Pl,Ma) Wrote

VePeMaSnThDc	Verb, Perfect, Masculine, Singular, Third Person, Indicative	كتبهُ	Ktbhu	He Wrote It
VePeNeSnFsJs	Verb, Perfect, Neuter, Singular, First Person, Jussive	علّمتهمْ	El mthmx	I teach them
VePeMaPlThSj	Verb, Perfect, Masculine, Plural, Third Person, Subjunctive	علّمونَا	El mwnA	They teach us
VePeMaPlFsJs	Verb, Perfect, Masculine, Plural, First Person, Jussive	علّمنَاهمْ	El mnAhmx	We teach them
VePeMaSnThJs	Verb, Perfect, Masculine, Plural, Third Person, Jussive	علّمنيْ	El mnyx	He teach me
VePeMaPlThJs	Verb, Perfect, Masculine, Plural, Third Person, Jussive	علّمونيْ	El mwnyx	They teach me
VePeFePlThSj	Verb, Perfect, Feminine, Plural, Third Person, Subjunctive	علّمكنَ	El mkna	They teach you
VePeNePlSeJs	Verb, Perfect, Neuter, Plural, Second Person, Jussive	علّمتوهمْ	El mtwhmx	You teach them
VePeMaSnFsSj	Verb, Perfect, Masculine, Singular, First Person, Subjunctive	علّمتهَا	El mthA	You (Sn) teach her
VePeMaPlSeSj	Verb, Perfect, Masculine, Plural, Second Person, Subjunctive	علّمتوهَا	El mtwhA	You (Pl) teach her
VePeFePlThDc	Verb, Perfect, Feminine, Plural, Third Person, Indicative	علّمنهُ	El mnhu	They (Fe) teach him
VePeFePlSeSj	Verb, Perfect, Feminine, Plural, Second Person, Subjunctive	علّمتمَاهَا	El mtmAhA	You teach her (Du)
VePeMaPlSeDc	Verb, Perfect, Masculine, Plural, Second Person, Indicative	علّمتموهُ	El mtmwhu	You teach him
VePeNePlThSj	Verb, Perfect, Neuter, Plural, Third Person, Subjunctive	علّمتموهَا	El mtmwhA	You teach her
VePeMaPlThJs	Verb, Perfect, Masculine, Plural, Third Person, Subjunctive	علّموكمْ	El mwkmx	They teach you

VePiMaSnFsSj	Verb, Imperfect, Masculine, Singular, First Person, Subjunctive	أعلّمهَا	OElmhA	I teach her
VePiMaSnFsJs	Verb, Imperfect, Masculine, Singular, First Person, Subjunctive	أعلّمهمْ	OElmhmx	I teach them
VePiMaSnFsDc	Verb, Imperfect, Masculine, Singular, First Person, Indicative	أعلّمُ	OElmu	I teach
VePiMaPlFsSj	Verb, Imperfect, Masculine, Plural, First Person, Subjunctive	نعلّمهَا	nElmhA	We teach her
VePiMaPlFsJs	Verb, Imperfect, Masculine, Plural, First Person, Subjunctive	نعلّمهمْ	nElmhmx	We teach them
VePIMaPlFsDc	Verb, Imperfect, Masculine, Plural, First Person, Indicative	نعلّمُ	nElmu	We teach
VePiMaDuThJs	Verb, Imperfect, Masculine, Dual, Third Person, Subjunctive	يعلّمَانِ	yElmAni	They(Ma,Du) teach
VePiMaPlThSj	Verb, Imperfect, Masculine, Plural, Third Person, Subjunctive	يعلّمونَ	yElmwna	They(Ma,Pl) teach
VePiFePlThSj	Verb, Imperfect, Feminine, Plural, Third Person, Subjunctive	يعلّمنَ	yElmna	They (Fe,Pl) teach
VePiMaSnThSj	Verb, Imperfect, Masculine, Singular, Third Person, Subjunctive	يعلّمهَا	yElmhA	He teach her
VePiMaSnThJs	Verb, Imperfect, Masculine, Singular, Third Person, Subjunctive	يعلّمهمْ	yElmhmx	He teach them
VePiMaSnThDc	Verb, Imperfect, Masculine, Singular, Third Person, Indicative	يعلّمُ	yElmu	He teach
VePiFeDuSeJs	Verb, Imperfect, Feminine, Dual, Second Person, Subjunctive	تعلّمَانِ	tElmAni	You(Du) teach
VePiFePlSeSj	Verb, Imperfect, Feminine, Plural, Second Person, Subjunctive	تعلّمنَ	tElmna	You(Pl) teach
VePiFeSnThDc	Verb, Imperfect, Feminine, Singular, Third Person, Indicative	تعلّمهُ	tElmhu	You(Sn) teach him

VePiFeSnThSj	Verb, Imperfect, Feminine, Singular, Third Person, Subjunctive	تعلّمهَا	tElmhA	You(Sn) teach her
VePiFeSnThJs	Verb, Imperfect, Feminine, Singular, Third Person, Subjunctive	تعلّمهمْ	tElmhmx	You(Sn) teach them
VePmMaSnSeJs	Verb, Imperative, Masculine, Singular, Second Person, Jussive	أُكتبْ	Ouktbx	You(Sn,Ma) Write
VePmFeSnSeJs	Verb, Imperative, Feminine, Singular, Second Person, Jussive	أُكتبيْ	Ouktbyx	You(Sn,Fe) Write
VePmNeDuSeSj	Verb, Imperative, Neuter, Dual, Second Person, Subjunctive	أُكتبَا	OuktbA	You(Du) Write
VePmFePlSeSj	Verb, Imperative, Feminine, Plural, Second Person, Subjunctive	أُكتبنَ	Ouktbna	You(Pl,Fe) Write
VePmMaPlSeSj	Verb, Imperative, Feminine, Plural, Second Person, Subjunctive	أُكتبوَا	OuktbwA	You(Pl,Ma) Write
PrPp	Preposition Particle	في	fy	In
PrVo	Vocative Particle	يَا	yA	Announcement
PrCo	Conjunction Particle	و	w	and
PrEx	Exception Particle	إلا	IlA	Except
PrAn	Annulment Particle	لَا	lA	Negation
PrSb	Subjunctive Particle	لن	ln	Never
PrJs	Jussive/Elision Particle	لم	lm	Never
Pr	Particle	إذَا	IdhA	If
NuPsMaSnThAcId	Personal Noun, Masculine, Singular, Third Person, Accusative,Indefinite	هوَ	hwa	He
NuPsNeDuThAcId	Personal Noun, Neuter, Dual, Third Person, Accusative, Indefinite	همَا	hmA	They(Dual)

NuPsMaPlThNmId	Personal Noun, Masculine, Plural, Third Person, Nominative, Indefinite	هُمْ	hmx	They(Pl,Ma)
NuPsFeSnThAcId	Personal Noun, Feminine, Singular, Third Person, Accusative,Indefinite	هِيَ	hya	She
NuPsFePlThAcId	Personal Noun, Feminine, Plural, Third Person, Accusative, Indefinite	هُنَّ	hn a	They(Pl,Fe)
NuPsMaSnSeAcId	Personal Noun, Masculine, Singular, Third Person, Accusative, Indefinite	أَنْتَ	Onta	You(Sn,Ma)
NuPsNeDuSeAcId	Personal Noun, Neuter, Dual, Third Person, Accusative,Indefinite	أَنْتُمَا	OntmA	You(Dual)
NuPsMaPlSeNmId	Personal Noun, Masculine, Plural, Third Person, Nominative,Indefinite	أَنْتُمْ	Ontmx	You(Pl,Ma)
NuPsFeSnSeGeId	Personal Noun, Feminine, Singular, Third Person, Genitive,Indefinite	أَنْتِ	Onti	You(Sn,Fe)
NuPsFePlSeAcId	Personal Noun, Feminine, Plural, Third Person, Accusative,Indefinite	أَنْتُنَّ	Ontn a	You(Pl,Fe)
NuPsNeSnFsAcId	Personal Noun, Neuter, Singular, First Person, Accusative,Indefinite	أَنَا	OnA	Me
NuPsNePlFsNmId	Personal Noun, Neuter, Plural, First Person, Nominative,Indefinite	نَحْنُ	nHnu	We
NuDeSnAcId	Demonstrative Noun, Singular, Accusative,Indefinite	هذَا	h*A	This
NuDeDuGeId	Demonstrative Noun, Dual, Genitive, Indefinite	هذَانِ	h*Ani	These(Dual)
NuDeSnGeId	Demonstrative Noun, Singular, Genitive, Indefinite	هذه	h*h	This(Sn)

NuDePlGeId	Demonstrative Noun, Plural, Genitive, Indefinite	هؤلَاء	hWlA’	These(Pl)
NuDe	Demonstrative Noun, Indefinite	هنَاكَ	hnAka	There
NuInId	Interrogative Noun, Indefinite	كيفَ	kyfa	How
NuCvSnId	Conjunctive Noun,Singular, Indefinite	الذي	Aldhy	Which/Who(Sn)
NuCvDuId	Conjunctive Noun,Dual, Indefinite	اللذَانِ	All*Ani	Which/Who(Du)
NuAdId	Adverbal Noun,Indefinite	فوقَ	fwqa	Over
NuVnId	Verbal Noun,Indefinite	هيَّا	hy A	Come On
NuCdId	Conditional Noun,Indefinite	متَى	mtY	When
NuNmId	Numeral Noun,Indefinite	وَاحد	wAHd	One
NuAjMsSnNmId	Adjective Noun, Masculine, Singular, Nominative, Indefinite	معلّمٌ	mEl mN	Instructor
NuAjMsSnAcId	Adjective Noun, Masculine, Singular, Accusative, Indefinite	معلّمَاً	mEl mAF	Instructor
NuAjMsSnGeId	Adjective Noun, Masculine, Singular, Genitive, Indefinite	معلّمٍ	mEl mK	Instructor
NuAjMsSnNmDf	Adjective Noun, Masculine, Singular, Nominative, Definite	المعلّمُ	AlmEl mu	Instructor(Ma,Sn)
NuAjMsSnAcDf	Adjective Noun, Masculine, Singular, Accusative, Definite	المعلّمَ	AlmEl ma	Instructor(Ma,Sn)
NuAjMsSnGeDf	Adjective Noun, Masculine, Singular, Genetive, Definite	المعلّمِ	AlmEl mi	Instructor(Ma,Sn)
NuAjMsDuGeId	Adjective Noun, Masculine, Dual, Genetive, Indefinite	معلّمَانِ	mEl mAni	Instructor(Ma,Du)
NuAjMsDuGeDf	Adjective Noun, Masculine, Dual, Genetive, Definite	المعلّمَانِ	AlmEl mAni	Instructor(Ma,Du)
NuAjFeSnNmId	Adjective Noun, Feminine, Singular, Nominative, Indefinite	معلّمةٌ	mEl mPN	Instructor(Fe,Sn)

NuAjFeSnAcId	Adjective Noun, Feminine, Singular, Accusative, Indefinite	معلّمةً	mEl mpF	Instructor(Fe,Sn)
NuAjFeSnGeId	Adjective Noun, Feminine, Singular, Genetive, Indefinite	معلّمةٍ	mEl mpK	Instructor(Fe,Sn)
NuAjFeSnNmDf	Adjective Noun, Feminine, Singular, Nominative, Definite	المعلّمةُ	mEl mpu	Instructor(Fe,Sn)
NuAjFeSnAcDf	Adjective Noun, Feminine, Singular, Accusative, Definite	المعلّمةَ	mEl mpa	Instructor(Fe,Sn)
NuAjFeSnGeDf	Adjective Noun, Feminine, Singular, Genetive, Definite	المعلّمةِ	AlmEl mpi	Instructor(Fe,Sn)
NuAjFeDuGeId	Adjective Noun, Feminine, Dual, Genetive, Indefinite	معلّمتَانِ	mEl mtAni	Instructor(Fe,Du)
NuAjFeDuGeDf	Adjective Noun, Masculine, Dual, Genetive, Definite	المعلّمتَانِ	AlmEl mtAni	Instructor(Fe,Du)
NuAjFeplNmId	Adjective Noun, Feminine, Plural, Nominative, Indefinite	معلّمَاتٌ	mEl mAtN	Instructor(Fe,Pl)
NuAjFePlGeId	Adjective Noun, Feminine, Plural, Genetive, Indefinite	معلّمَاتٍ	mEl mAtK	Instructor(Fe,Pl)
NuAjFePlNmDf	Adjective Noun, Feminine, Plural, Nominative, Definite	المعلّمَاتُ	AlmEl mAtu	Instructor(Fe,Pl)
NuAjFePlAcDf	Adjective Noun, Feminine, Plural, Accusative, Definite	المعلّمَاتَ	AlmEl mAta	Instructor(Fe,Pl)
NuAjFePlGeDf	Adjective Noun, Feminine, Plural, Genetive, Definite	المعلّمَاتِ	AlmEl mAti	Instructor(Fe,Pl)
NuAjFePlAcId	Adjective Noun, Masculine, Plural, Accusative, Indefinite	معلّمونَ	mEl mwna	Instructor(Ma,Pl)
NuAjFePlAcDf	Adjective Noun, Masculine, Plural, Accusative, Definite	المعلّمونَ	AlmEl mwna	Instructor(Ma,Pl)
NuIsMaSnNmId	Instrument Noun, Masculine, Singular, Nominative, Indefinite	مفتَاحٌ	mftAHN	Key

NuIsMaDuGeId	Instrument Noun, Masculine, Dual, Genetive, Indefinite	مفتاحانِ	mftAHAni	(Two) Keys
NuIsMaPlNmId	Instrument Noun, Masculine, Plural, Nominative, Indefinite	مفاتيحٌ	mfAtyHN	Keys
NuIsMsSnNmDf	Instrument Noun, Masculine, Singular, Nominative, Definite	المفتاحُ	AlmftAHu	The Key
NuIsMsSnAcDf	Instrument Noun, Masculine, Singular, Accusative, Definite	المفتاحَ	AlmftAHa	The Key
NuIsMsSnGeDf	Instrument Noun, Masculine, Singular, Genetive, Definite	المفتاحِ	AlmftAHi	The Key
NuIsMaDuGeId	Instrument Noun, Masculine, Dual, Genetive, Indefinite	المفتاحانِ	AlmftAHAni	(Two) Keys
NuIsMaPlNmDf	Instrument Noun, Masculine, Plural, Nominative, Definite	المفاتيحُ	AlmfAtyHu	Keys
NuIsMaPlAcDf	Instrument Noun, Masculine, Plural, Accusative, Definite	المفاتيحَ	AlmfAtyHa	Keys
NuIsMaPlGeDf	Instrument Noun, Masculine, Plural, Genetive, Definite	المفاتيحِ	AlmfAtyHi	Keys
NuDmSnNmId	Diminutive Noun, Singular, Nominative, Indefinite	مطعمٌ	mTyEmN	Restaurant
NuReMaSnNmId	Relative Noun, Masculine, Singular, Nominative, Indefinite	اردنيٌ	ArdnyN	Jordanian (Ma,Sn)
NuReFeSnNmId	Relative Noun, Feminine, Singular, Nominative, Indefinite	اردنيةٌ	ArdnypN	Jordanian (Fe,Sn)
NuReMaDuGeId	Relative Noun, Masculine, Dual, Genitive, Indefinite	اردنيانِ	ArdnyAni	Jordanian (Ma,Du)
NuReFeDuGeId	Relative Noun, Feminine, Dual, Genitive, Indefinite	اردنيتانِ	ArdnytAni	Jordanian (Fe,Du)
NuReMaPlAcId	Relative Noun, Masculine, Plural, Accusative, Indefinite	اردنيونَ	Ardnywna	Jordanian (Ma,Pl)

NuReFePlNmId	Relative Noun, Feminine, Plural, Nominative, Indefinite	اردنيَاتٌ	ArdnyAtN	Jordanian (Fe,Pl)
NuReMaSnNmDf	Relative Noun, Masculine, Singular, Nominative, Definite	الَاردنيُ	AlArdnyu	Jordanian (Ma,Sn)
NuReFeSnNmDf	Relative Noun, Feminine, Singular, Nominative, Definite	الَاردنيةُ	AlArdnypu	Jordanian (Fe,Sn)
NuReMaDuGeDf	Relative Noun, Masculine, Dual, Genitive, Definite	الَاردنيَانِ	AlArdnyAni	Jordanian (Ma,Du)
NuReFeDuGeDf	Relative Noun, Feminine, Dual, Genitive, Definite	الَاردنيتَانِ	AlArdnytAni	Jordanian (Fe,Du)
NuReMaPlAcDf	Relative Noun, Masculine, Plural, Accusative, Definite	الَاردنيونَ	AlArdnywna	Jordanian (Ma,Pl)
NuReFePlNmDf	Relative Noun, Feminine, Plural, Nominative, Definite	الَاردنيَاتُ	AlArdnyAtu	Jordanian (Fe,Pl)
NuCnMaSnNmId	Common Noun, Masculine, Singular, Nominative, Indefinite	كتَابٌ	ktAbN	Book (Sn)
NuCnFeSnNmId	Common Noun, Feminine, Singular, Nominative, Indefinite	مدرسةٌ	mdrspN	School (Sn)
NuCnMaSnAcId	Common Noun, Masculine, Singular, Accusative, Indefinite	كتَابً	ktAbF	Book (Sn)
NuCnFeSnNmId	Common Noun, Feminine, Singular, Accusative, Indefinite	مدرسةً	mdrspF	School (Sn)
NuCnMaSnGeId	Common Noun, Masculine, Singular, Genitive, Indefinite	كتَابٍ	ktAbK	Book (Sn)
NuCnFeSnGeId	Common Noun, Feminine, Singular, Genitive, Indefinite	مدرسةٍ	mdrspK	School (Sn)
NuCnMaDuGeId	Common Noun, Masculine, Dual, Genitive, Indefinite	كتَابَانِ	ktAbAni	Books (Du)
NuCnFeDuGeId	Common Noun, Feminine, Dual, Genitive, Indefinite	مدرستَانِ	mdrstAni	Schools (Du)

NuCnFePlGeId	Common Noun, Feminine, Plural, Genitive, Indefinite	مدَارِسِ	mdArsi	Schools (Pl)
NuCnFePlAcId	Common Noun, Feminine, Plural, Accusative, Indefinite	مدَارِسَ	mdArsa	Schools (Pl)
NuCnFePlNmId	Common Noun, Feminine, Plural, Genitive, Indefinite	مدَارِسُ	mdArsu	Schools (Pl)
NuCnMaPlNmId	Common Noun, Masculine, Plural, Nominative, Indefinite	كتبُ	ktbu	Books (Pl)
NuCnMaPlAcId	Common Noun, Masculine, Plural, Accusative, Indefinite	كتبَ	ktba	Books (Pl)
NuCnMaPlGeId	Common Noun, Masculine, Plural, Genitive, Indefinite	كتبِ	ktbi	Books (Pl)
NuCnMaSnNmDf	Common Noun, Masculine, Singular, Nominative, Definite	الكتَابُ	AlktAbu	Book (Sn)
NuCnMaSnAcDf	Common Noun, Masculine, Singular, Accusative, Definite	الكتَابَ	AlktAba	Book (Sn)
NuCnMaSnGeDf	Common Noun, Masculine, Singular, Genitive, Definite	الكتَابِ	AlktAbi	Book (Sn)
NuCnFeSnNmDf	Common Noun, Feminine, Singular, Nominative, Definite	المدرسةُ	Almdrspu	School (Sn)
NuCnFeSnAcDf	Common Noun, Feminine, Singular, Accusative, Definite	المدرسةَ	Almdrspa	School (Sn)
NuCnFeSnGeDf	Common Noun, Feminine, Singular, Genitive, Definite	المدرسةِ	Almdrspi	School (Sn)
NuCnMaDuGeDf	Common Noun, Masculine, Dual, Genitive, Definite	الكتَابَانِ	AlktAbAni	Books (Du)
NuCnFeDuGeDf	Common Noun, Feminine, Dual, Genitive, Definite	المدرستَانِ	AlmdrstAni	Schools (Du)
NuCnFePlGeDf	Common Noun, Feminine, Plural, Genitive, Definite	المدَارِس	AlmdArsi	Schools (Pl)

NuCnFePlAcDf	Common Noun, Feminine, Plural, Accusative, Definite	المدَارسَ	AlmdArsa	Schools (Pl)
NuCnFePlNmDf	Common Noun, Feminine, Plural, Genitive, Definite	المدَارسُ	AlmdArsu	Schools (Pl)
NuCnMaPlNmDf	Common Noun, Masculine, Plural, Nominative, Definite	الكتبُ	Alktbu	Books (Pl)
NuCnMaPlAcDf	Common Noun, Masculine, Plural, Accusative, Definite	الكتبَ	Alktba	Books (Pl)
NuCnMaPlGeDf	Common Noun, Masculine, Plural, Genitive, Definite	الكتبِ	Alktbi	Books (Pl)
NuIf	Infinitive Noun	صومٌ	swmN	Fasting
NuPo	Proper Noun	رمزي	ramzy	Proper Noun
NuPn	Noun of Place	مطبخٌ	mtbxN	Kitchen
NuTn	Noun of Time	موعدٌ	mwEdN	Engagement

Appendix B

The Arabic Language Orthography

B.1 Arabic words and the Roman alphabet

The issue of transliteration and transcription codes used to describe Arabic language using the Roman alphabet to give a reader unfamiliar with the language sufficient information for accurate pronunciation still presented. Marshall Hodgson ([70],p4) define transliteration as : "is the rendering of the spelling of a word from the script of one language into another language"', and transcription as : " is the rendering of the sound of a word so that a reader can pronounce". For example, the transliteration of the Arabic word كتب may *'ktb'*, while one of the transcription is *"kataba"*, other may be "kutub".[1] Many different approaches and a variety of ways for transliteration (romanizing) Arabic language have been developed. Some of these transliteration systems are listed below[2]:

- Deutsche Morgenlandische Gesellschaft (1936): adopted by the International Convention of Orientalist Scholars in Rome.
- Romanization Tables adopted by the US Library of Congress and the American Library Association for cataloguing books (ALA-LC).
- ISO 233 published by the International Standards Organisation, BS 4280:1968 produced by British Standards Institute.

[1] Since the word كتب is unvocalized, the problem of pronouncing the word may arises.

[2] For more information : http://www.al-bab.com/arab/language/roman1.htm
See also : http://en.wikipedia.org/wiki/

- (UNGEGN) United Nations Romanization System for Geographical Names
- Romanization, Transcription and Transliteration by Kenneth R. Beesley (Xerox company).
- The Buckwalter Transliteration System.
- Al-kitaab Transliteration System published by by Kristen Brustad, Mahmoud Al-Batal, and Abbas Al-Tonsi.
- The Standard Arabic Technical Transliteration System (SATTS).
- DIN 31635 developed by the Deutsches Institut fr Normung (German Institute for Standardization).
- SAS: Spanish Arabists School (Jos Antonio Conde and others).
- BGN/PCGN 1956: Romanization System For Arabic.

Unfortunately, none of the systems described above is an universal standard for transliteration and transcription Arabic language. All the systems described above suffer from many difficulties, such as : they use special characters or add special marks to normal characters which make these systems difficult to memories as well as most of these systems cannot be used easily with a standard computer keyboard. On other hand, a few Arabic letters have a clear equivalent in the Roman alphabet(B,F,K,L,M,N,R, and Z)[3].

Due to some difficulties described above, each person uses their own standard. Throughout this thesis we use a transliteration system compound from Buckwalter and Al-kitaab transliteration systems with a little bit of my own update to transliterate diacritical marks described in table B.4.

B.2 Arabic alphabet and other diacritical marks

The alphabet in Arabic language consist of 28 letters. Unlike European languages, no separate printed form of the letter. Arabic script is a cursive and written from right to left [76].

Table B.1 shows the various forms of Arabic letters and transliteration of each letter which has been used to transliterate Arabic words throughout this thesis.

In addition, Arabic lanaguge has hamza ء (glottal stop) consonant which can also occur on alif or waaw or yaay consonants and Ta Marboota ة, these letters shown in table B.2).

No.	Name	Consonant	Transliteration	Pronunciation
1	Alif	ا	A	m*a*n
2	baa	ب	b	*b*ack
3	taa	ت	t	*t*ablet
4	thaa	ث	th	*th*row
5	jiim	ج	j	*j*ohn
6	Haa	ح	H	*H*at
7	khaa	خ	kh	
8	daal	د	d	*d*ad
9	dhaal	ذ	dh	
10	raa	ر	r	*r*ush
11	zaay	ز	z	car*z*y
12	siin	س	s	*s*un
13	shiin	ش	sh	*sh*adow
14	Saad	ص	S	*S*uffix
15	Daad	ض	D	
16	Taa	ط	T	
17	DHaa	ظ	DH	
18	ayn	ع	E	
19	ghayn	غ	gh	
20	faa	ف	f	*f*at
21	qaaf	ق	q	*q*uick
22	kaaf	ك	k	
23	laam	ل	l	*l*aptop
24	miim	م	m	*m*ark
25	nuun	ن	n	*n*ovel
26	haa	ه	h	*h*assel
27	waaw	و	w	*w*elcome
28	yaay	ي	y	*y*oung

Table B.1: Arabic Alphabet

Name	Consonant	Transliteration
Hamza	ء	'
hamza above Alif	أ	O
hamza below Alif	إ	I
hamza above waaw	ؤ	W
hamza above yaay	ئ	}
Ta Marboota	ة	p
Alif Maqsoura	ى	Y

Table B.2: Hamza (glottal stop) with Alif, waaw, and yaay consonants

Furthermore, table B.3 shows the transliteration system for short vowels diacritical marks, while the transliteration of other diacritical marks (Nunation,Sukun,gemination) described in table B.4.

Name	Mark in consonant	Transliteration	Pronunciation
Fatha sign	كَ	a	/a/
damma sign	كُ	u	/u/
kasra sign	كِ	i	/i/

Table B.3: Arabic short vowels

Name	Mark in consonant	Transliteration	Pronunciation
Tanween fath	دً	an	/an/
Tanween damm	دٌ	un	/un/
Tanween kasr	دٍ	in	/in/
Sukun	دْ	x	
Shadda	بّ	=	

Table B.4: Other diacritical marks (Nunation,Sukun,gemination) in Arabic

[3] For more information : http://www.al-bab.com/arab/language/roman1.htm

Appendix C

Lexical and Contextual Rules

C.1 Names and description of lexical rules

Rule Name	Description
CWD	The current word
CWDLM	The last diacritical mark of the current word
F1CHCWD	The first character of the current word
F2CHCWD	The first two characters of the current word
L2CHCWD	The last two characters of the current word
F3CHCWD	The first three characters of the current word
L3CHCWD	The last three characters of the current word

C.2 Lexical Rule Examples

Tanween Damm **CWDLM** `NuCnNmId`

Tanween Fath **CWDLM** `NuCnAcId`

Tanween Kasr **CWDLM** `NuCnGeId`

يُّ **L3CHCWD** `NuRe`

وَال **F3CHCWD** and *Kasra mark* **CWDLM** `PrCo+NuCnGeDf`

C.3 Names and description of contextual rules

Rule Name	Description
PWD	The preceding word
PWDTAG	The preceding tag

C.4 Examples used contextual rules

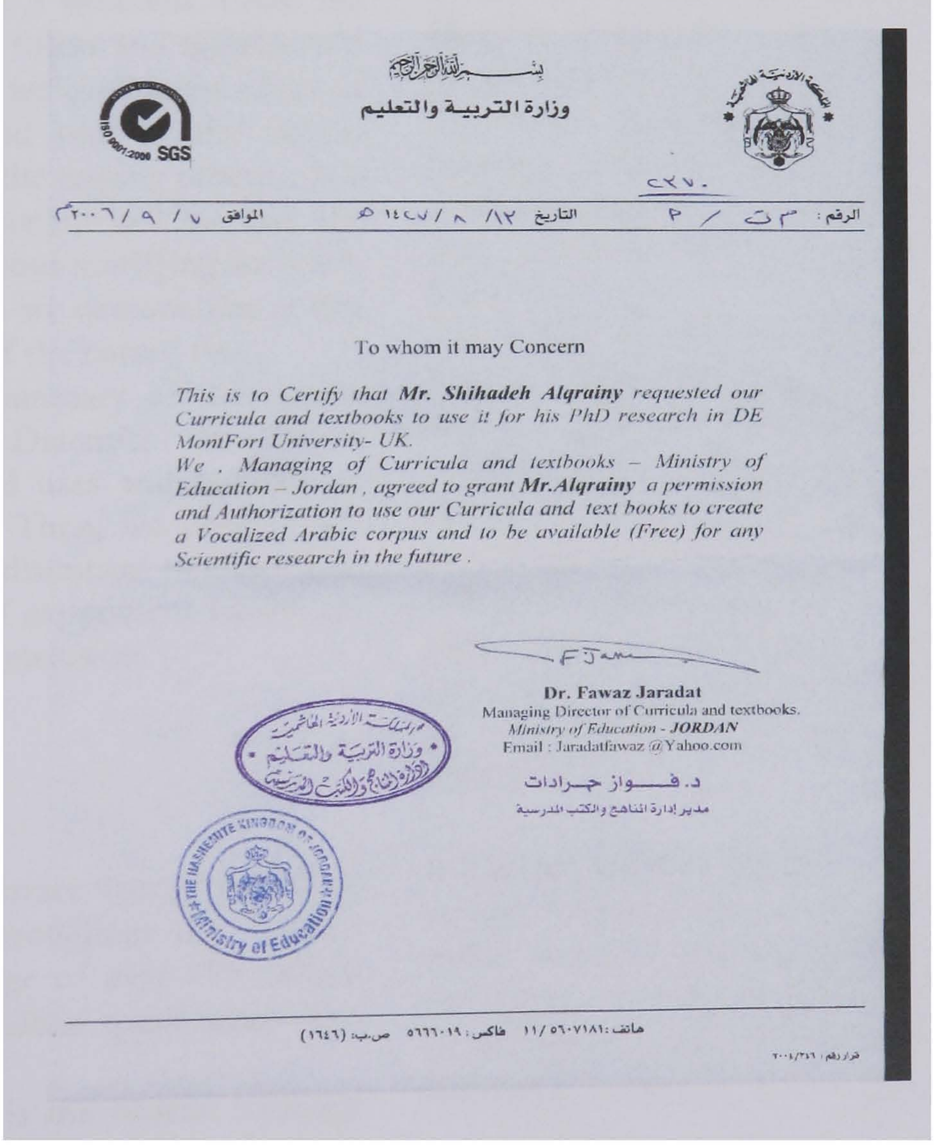
`NuCnGeId` **PWDTAG** `PrPp`

`NuCn` **PWD** يَا

`NuPo` **PWD** جَامعةَ `or` جَامعةِ `or` جَامعةُ

Appendix D

Permission for Collecting Testing Corpus



بسم الله الرحمن الرحيم

وزارة التربية والتعليم

ISO 9001:2000 SGS

٢٧٠

الرقم : م ت / ٣   التاريخ ١٢/ ٨ /١٤٢٧ هـ   الموافق ٧ / ٩ / ٢٠٠٦م

To whom it may Concern

This is to Certify that ***Mr. Shihadeh Alqrainy*** *requested our Curricula and textbooks to use it for his PhD research in DE MontFort University- UK.*

We , Managing of Curricula and textbooks – Ministry of Education – Jordan , agreed to grant ***Mr.Alqrainy*** *a permission and Authorization to use our Curricula and text books to create a Vocalized Arabic corpus and to be available (Free) for any Scientific research in the future .*

F Jaradat

Dr. Fawaz Jaradat
Managing Director of Curricula and textbooks.
Ministry of Education - ***JORDAN***
Email : Jaradatfawaz @Yahoo.com

د. فـــــواز جـــرادات
مدير إدارة المناهج والكتب المدرسية

المملكة الأردنية الهاشمية
وزارة التربية والتعليم
إدارة المناهج والكتب المدرسية

THE HASHEMITE KINGDOM OF JORDAN
Ministry of Education

هاتف: ٥٦٠٧١٨١/ ١١   فاكس: ٥٦٦٦٠١٩   ص.ب: (١٦٤٦)

قرار رقم : ٢٠٠٤/٣٤٦

Word-Class Tagger and Tagset Design for Vocalized Arabic Text

SHIHADEH ALQRAINY+ , ALADDIN AYESH+
+Centre for Computational Intelligence (CCI) - School of Computing
De Montfort University, Leicester – The Gateway, UNITED KINDOM
{alqrainy , aayesh}@dmu.ac.uk

Abstract: - Arabic language has a valuable and important feature, called diacritics, which are marks placed over and below the letters of Arabic word. This feature plays a great role in adding linguistic attributes to Arabic words and in indicating pronunciation and grammatical function of the words. This feature enriches the language syntactically while removing a great deals of morphological and semantically ambiguities. This paper present diacritics rule-based part-of-speech (POS) tagger which automatically tags a partially vocalized Arabic text. The aim is to remove ambiguity and to enable accurate fast automated tagging system. A tagset is being designed in support of this system. Tagset design is at an early stage of research related to automatic morphosyntactic annotation in Arabic language. Preliminary results of the tagset design have been reported in this paper.

Keywords: - Arabic Language, Part-Of-Speech (POS), Diacritics, Tagset, Morphological, Syntactical

1 Introduction

Arabic language is syntactically and morphologically a rich language, which means several words and meanings, can be derived from the same word leading to ambiguity. The ambiguity of Arabic lies on 3 different levels, the core word level, the derived word forms and agglutinative forms of words [1].In this paper; we exploit the effect of vocalization, which is considered one of the Arabic Language distinctive features, on the tagging process. It is envisaged that the use of vocalization will increase the speed of the tagging process without scarifying accuracy. Indeed, the use of vocalization, as we demonstrate in this paper, will reduce the ambiguity of the parsed text.

The paper starts with a brief summary of the Arabic language overview followed Diacritics in Arabic Language. The tagset design and uses and benefits of tagging systems are highlighted. Then, we present our tagging system architecture and diacritical rule-based as our approach. Finally, analyses of experiment results are presented with future work and conclusion.

2 Arabic Language

2.1 Background

The Arabic language is spoken in more than 20 countries, from Egypt to Morocco and throughout the Arabian Peninsula. It is the native language of over 195 million people. Plus, at least another 35 million speak Arabic as a second language.

Modern Standard Arabic (MSA) is the official language throughout the Arab world, and its written form is relatively consistent across national boundaries. MSA is used in official documents, in educational settings, and for communication between Arabs of different nationalities. However, the spoken forms of Arabic vary widely, and each Arab country has its own dialect. Dialects are spoken in most informal settings, such as at home, with friends, or while shopping.

The Arabic language belongs to the Semitic family of languages, and, like Hebrew, is written from right to left. Arabic has been a literary language since the 6th century A.D., and is the liturgical language of Islam in its classical form.

The Arabic writing system is quite different from the English system. The Arabic alphabet consists of 28 letters that change shape depending on their position within a word and the letters by which they are surrounded. Some Arabic letters must be connected to other letters; others may stand alone. Arabic vowels are indicated by marks (Diacritics) above and below the consonants. In many cases, these diacritics play the role of vowels in English and thus influence pronunciation. Additionally, there are no special forms, such as the use of capital letters in English [17].

2.2 Diacritics in Arabic

Arabic language has a valuable and important feature, called diacritics, which are marks placed over and below the letters of Arabic word. This feature plays a great role in adding linguistic attributes to Arabic words and in indicating pronunciation and grammatical function of the

words. It is particularly of interest for the purpose of this paper. Table 1 shows Arabic vowel diacritics.
The pronunciation of diacritized languages words cannot be fully determined by spelling their characters only; special marks are put above or below the characters to determine the correct pronunciation. They also indicate the grammar function of the word within the context of the sentence [2].

Name :	Fatha	Damma	Kasra
Symbol :	َ	ُ	ِ
	/a /	/u /	/i /
Explanation:	Written above the Consonant	Written above the Consonant	Written below the Consonant
Example :	بَ	بُ	بِ
Pronunciation:	ba	bu	bi
Name :	Tanween Fatha	Tanween Damm	Tanween Kasr
Symbol :	ً	ٌ	ٍ
	/an/	/un/	/in/
Explanation:	Written above the Consonant	Written above the Consonant	Written below the Consonant
Example :	بًا	بٌ	بٍ
Pronunciation:	ban	bun	bin
Name :	Shadda		Sukun
Symbol :	ّ		ْ
Explanation:	Written above the Consonant		Written above the Consonant
Example :	بّ		بْ
Pronunciation:	bb		b

Table 1: Arabic vowel diacritics

In Arabic, short vowels are not a part of the Arabic alphabet, instead they are written as marks over or below the consonant. They are used in both Noun and Verb in Arabic Language. They indicate the case of the noun and the mood of the verb.

Many words are in general ambiguous in their part-of-speech, for various reasons. In English, for example, a word such as "Make" can be "Verb" or "Noun". In Arabic there are ambiguities. For example, the word "ذهب" which either means "go" or "gold" can be Verb or Noun.
Diacritics are used to prevent misunderstandings, to determine the correct pronunciation, reduce the ambiguity, and indicating grammatical functions. These functions play a great role in removing ambiguity and enabling accurate fast automated tagging system.
To remove ambiguity and to determine the correct tag of the word " ذهب " in the above example, adding the short vowels (Fatha sign) to the last letter of the word to become " ذهبَ " enough to get the correct tag [Verb] without any ambiguity and without regards to the context.

3 Arabic Tagset and EAGLES guidelines

A tag is a code which represents some features or set of features and is attached to the segment in a text. Single or complex information are carried by a tag. The development of a tagset to support diacritical based tagging system is at early stage. The need for such a tagset comes from the fact that there is no standardized and comprehensive Arabic tagset.

EAGLES [16] guidelines outline a set of features for tagsets; these guidelines were designed to help standardise tagsets for what were then the official languages of the European Union. EAGLES tags are defined as sets of morphosyntactic attribute-value pairs (e.g. Gender is an attribute that can have the values Masculine, Feminine or Neuter). The tagset discussed here is not being developed in accordance with the EAGLES guidelines for morphosyntactic annotation of corpora. Arabic is very different from the languages for which EAGLES was designed, and belongs to the Semitic family rather than the Indo-European one. Following a normalised tagset and the EAGLES recommendations would not capture some of Arabic's relevant information, such as the jussive mood of the verb and the dual number that are integral to Arabic. Another important aspect of Arabic is inheritance, where all subclasses of words inherit properties from the classes from which they are derived. For example, all subclasses of the noun inherit the Nunation when in the indefinite which is one of the main properties of the noun [14].

3.1 Previous work on POS tagsets

There are small numbers of popular tagsets for English, such as: 87-tag tagset used Brown Corpus, 45-tag Penn Treebank tagset and 61-tag C5 tagset [3]. For Arabic also very small number of tagset had been built, El-Kareh S, Al-Ansary[10] described the tagset, they classifying the words into three main classes, Verbs are sub classified into 3 subclasses; Nouns into 46 subclasses and Particles into 23 subclasses. Shereen Khoja[14] described more detail tagset. Her tagset contains 177 tags, 57 Verbs, 103 Nouns, 9 Particles, 7 residual and 1 punctuation.

3.2 Proposed Arabic Tagset

We have based our Arabic tagset on inflectional morphology system. The traditional description of Arabic grammarians consider as a base to create the linguistic categories of Arabic tagset. Arabic grammarians describe Arabic as being derived from three main categories: noun, verb and particle. Figure 1 shows the tagset hierarchy.

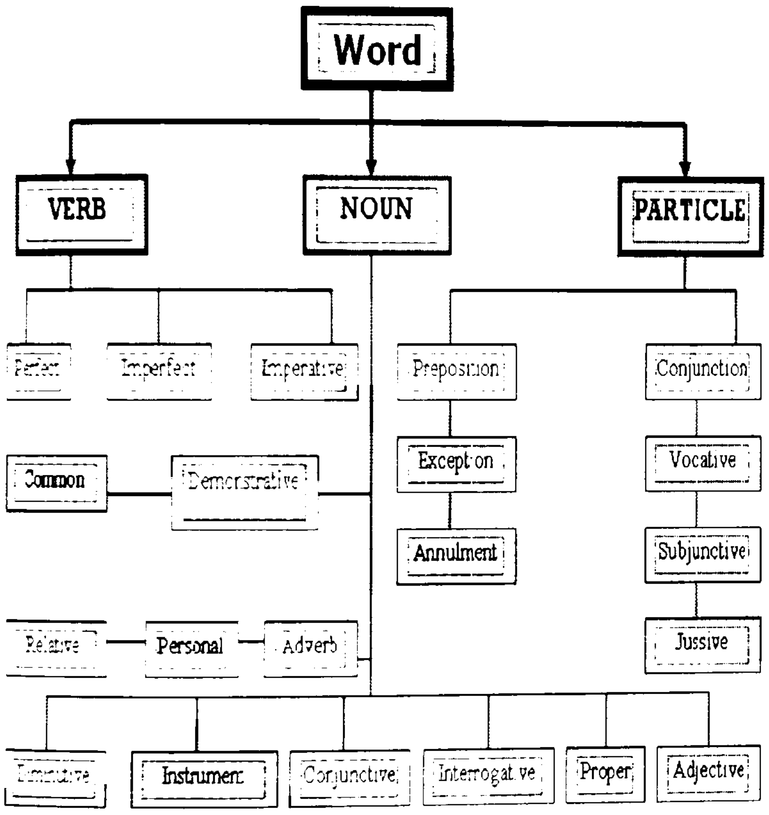


Fig. 1: Tagset Hierarchy.

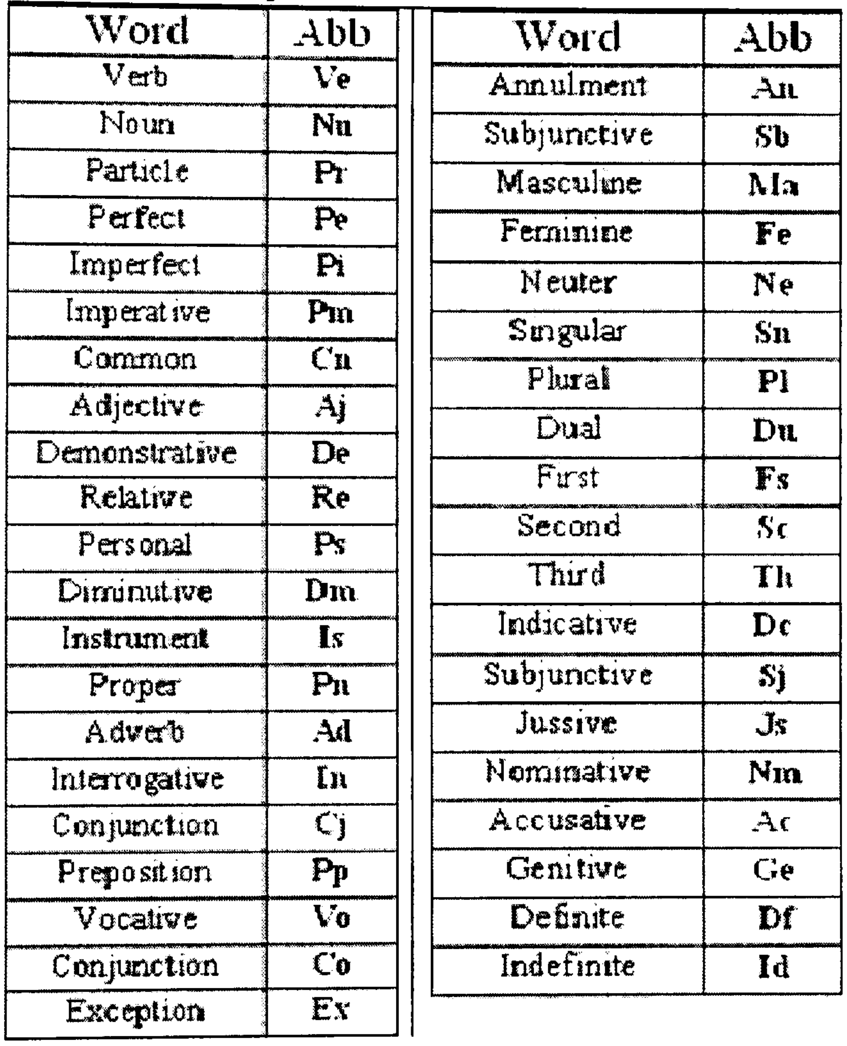
The tagset has the following main formula:

[T , S , G , N , P , M , C , F] , Where:
T (Type) = {Verb, Noun, Particle}
S = Sub-Class {Common, Demonstrative, Relative, Personal,Adverb,Diminutive,Instrument, Conjunctive, Interrogative, Proper and Adjective}
G (gender) = {Masculine, Feminine, Neuter}
N (Number) = {Singular, Plural, Dual}
P (Person) = {First, Second, Third}
M (Mood) = {Indicative, Subjunctive, Jussive}
C (Case) = {Nominative, Accusative, Genitive}
F (State) = {Definite, Indefinite}

Figure 2 shows the Abbreviations which was used to define the words in our tagset.

Let us try to explain the symbols of the tagset formula for a moment.

The symbols [**T , S , G , N , P , M**] consider as linguistic attributes for class Verb, while the symbols [**T , S , G , N , P , C , F**] consider as linguistic attributes for class Noun. For example , the word " كتب " which means "he wrote" has the following tag [VePeMaSnThSj], which means [Perfect Verb , Masculine Gender , Singular Number , Third Person , Subjunctive Mood].



Word	Abb	Word	Abb
Verb	Ve	Annulment	An
Noun	Nu	Subjunctive	Sb
Particle	Pr	Masculine	Ma
Perfect	Pe	Feminine	Fe
Imperfect	Pi	Neuter	Ne
Imperative	Pm	Singular	Sn
Common	Cn	Plural	Pl
Adjective	Aj	Dual	Du
Demonstrative	De	First	Fs
Relative	Re	Second	Sc
Personal	Ps	Third	Th
Diminutive	Dm	Indicative	Dc
Instrument	Is	Subjunctive	Sj
Proper	Pn	Jussive	Js
Adverb	Ad	Nominative	Nm
Interrogative	In	Accusative	Ac
Conjunction	Cj	Genitive	Ge
Preposition	Pp	Definite	Df
Vocative	Vo	Indefinite	Id
Conjunction	Co		
Exception	Ex		

Fig. 2: Tagset Abbreviations

4 Part-Of-Speech Tagging

4.1 Related Work

Part-of-speech tagging is the process of assigning a part-of-speech or other syntactic class marker to each word in a corpus [3]. Tagger is necessary for many applications, such as: speech synthesis system, speech recognition system, informational retrieval (IR) and parsing system. Many techniques have been used to tag English and other European languages corpora. Greene and Rubin [4] developed the first Rule-Based technique to tag Brown Corpus. Eric Brill's[5] interest in rule-based tagger. Garside[15] used hidden Markov Model to develop

CLAWS tagger. More recently, taggers that use combination of both Statistical and rule-based[6], Machine learning [7] and Neural Network [8,9] have been developed.

In terms of Arabic, small numbers of popular Part-of-Speech (POS) tagger have been developed. El-Kareh and Al-Ansary[10] described a hybrid semi-automatic tagger that uses both morphological rules and statistical techniques in the form of hidden Markov models. Abuleil and Evens[11] describe a system for building an Arabic lexicon automatically by tagging Arabic newspaper text. Shereen Khoja[12] described an Arabic part-of-speech called APT that uses statistical and rule-based techniques. Diab, Mona et al.[13] presented a Support Vector Machine (SVM) based approach to automatically tokenize, part-of-speech tag in Arabic text.

4.2 Proposed Arabic POS Tagging System

Our tagger is called AWTS - short for Arabic Word-Tagging System - and its main function is to take as input untagged Arabic text, and produce a POS tagged Text. An Overview of AWTS can be seen in figure 3.

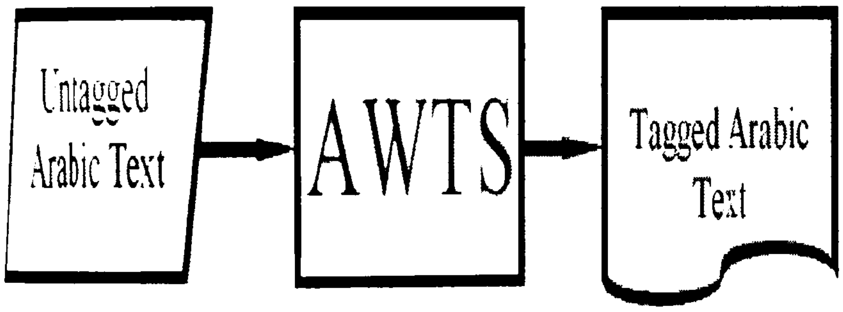


Fig. 3: An Overview of AWTS

The description of the AWTS modules shown in Fig 4.

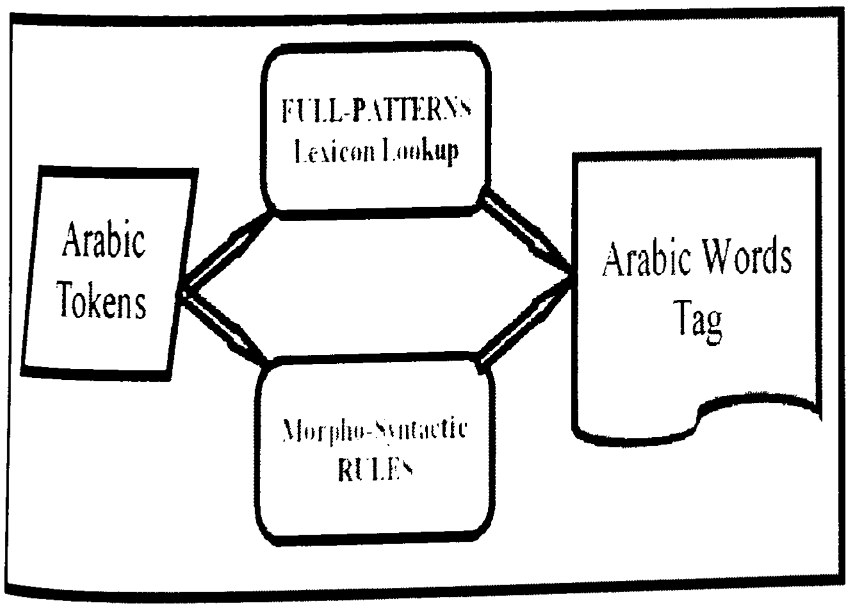


Fig 4: AWTS Modules

During tagging, the Arabic Token is first looked up in the Full-Patterns lexicon which contains the Full Patterns (Prefixes+Forms+Suffixes) of most Arabic Word. We introduce an algorithm describe how we match the tokens with its pattern. The pseudo code of proposed algorithm described in section 4.3. If the pattern of the token is found, then it is assigned the most likely tag of the word. If not, the word is then passed to the morpho-syntactic rules module to apply some linguistics rules to extract the most likely tag of the word. Some of these rules shown in section 4.3.

4.3 Proposed Approach

The proposed approach consists of two Parts: *Pattern-Base Approach and Linguistics Rules.*

Pattern-Base Approach, based on Full-patterns with diacritics. Arabic language has a rich morphological system that contains a lot of patterns. These patterns assign part-of-speech tag of the Arabic word. Some of patterns belong to Verb class, while the others belong to Noun class. Particle has no patterns in Arabic language. We generate automatically a Full-Patterns lexicon by collecting the Prefixes and Forms and Suffixes for most Arabic words.

Algorithm-1 shows the pseudo code to describe how we match the tokens with its pattern. Figure 5 shows an example, how to trace the steps of algorithm to match the pattern for the word " قَاتَلَ ", "*to fight*".

Let P=Full-Pattern, W=Inflected Word, T=Tag.

```
Step-1:  Return all P from lexicon where P=Len (W).
           Store results (Number of pattens) in N.
Step-2:  For I = 1 to N
           Compute the number of identical letters
           between P(I) and W. Store results in Sim.
           Next I
Step-3:  Return all P which have the Maximum (Sim)
           Store results in M.
Step-4:  For J = 1 to M
Step-5:  Convert each letter of " ف f " or " ع E " or
           " ل l " in P(I) with the corresponding letters
           of W.
Step-6:  If P(J) = W then Return P(J) , T(J).
           Go to Step 8
Step-7:  Next J
Step-8:  Exit
```

Algorithm-1: Pattern-Match-Algorithm

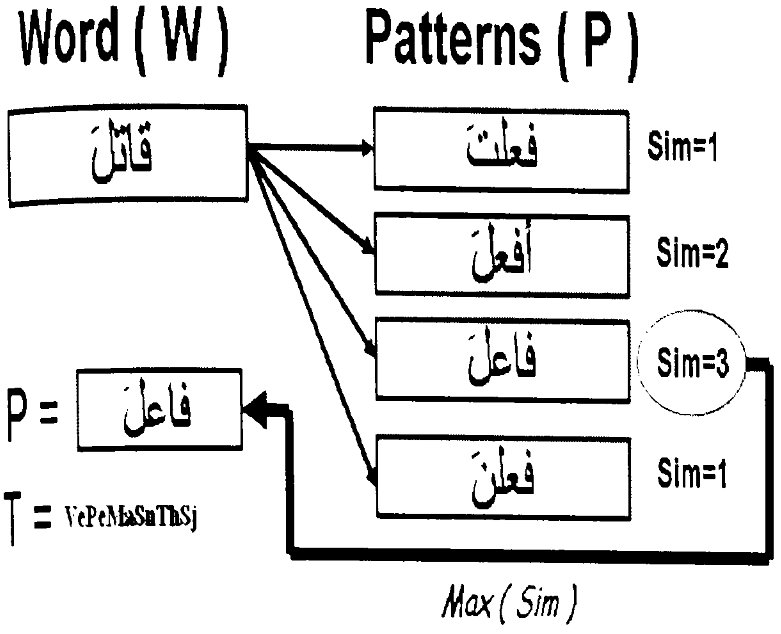


Fig. 5: Matching-Pattern Example

Linguistics Rules uses Syntactical Information and Morphological Information without regard to context and lookup tables to assign most likely tag to each unknown and ambiguous word in the text.

Some of these rules as examples are listed below:

Consider W = The word, T = The Tag

Rule-1: If W end with " يَ " or " يَة ", then
T = [**NuRe**].
For Example, the word " اردنيَ ", "Jordanian"

Rule-2: If W end with ً or ٌ or ٍ , then
T = [**NuCn**]
For Example, the word " رجلٌ ", "Man"

5 Results

We tested our system to tag the words using partial-diacritization documents from the holly Qur'an and another set chosen randomly from the proceedings of the Saudi Arabian National Computer Conference and other resources.

We ran our system on a group of these documents. The accuracy of our system has been calculated for tagging the words. The total accuracy about 81 \%, 19 \% in errors. Some errors of the system came from Arabized words which are translated as pronounced from other international languages, such as the word " كمبيوتر". These words do not have a root and a pattern. Others came from irregular verbs such as the word " ضلّ ". Also some words in Arabic language consider as primitive verbs, such as, " بئس ", " نعمَ " . These words not tagged correctly and need a special treatment.

6 Conclusion and Future Works

In this paper, we presented diacritics rule-based part-of-speech (POS) tagger which automatically tags a partially vocalized Arabic text. Also, we describe a morphosyntactic tagset that is derived from the ancient Arabic grammar, which is based on Arabic system of inflectional morphology. The tagset does not follow the traditional Indo-European tagset that is based on Latin but is instead based on the Semitic tradition of analysing language. These tags contain a large amount of information and add more linguistic attributes to the word. Also, we are currently collecting many rules to reduce the amount of errors and expanding our tagset to cover most categories word in Arabic language.

It's clear that an overall ambiguity in a vocalised text is quite lower than in an unvocalised text. Diacritics are used to prevent misunderstandings and reduce the ambiguity; diacritics play a great role to speed the tagging process without scarifying accuracy and remove a great deal of morpho-lexical ambiguity when the text is partial diacritization

References

[1] M. V. Mol, "The semi-automatic tagging of Arabic corpora," *COLING 94, USA, 1994.*

[2] M.A.Elaraby2000, "Alarge scale Computational processor of the Arabic Morphology and application.". *(Master's thesis), Cairo University, Egypt.*

[3] D.J.J.H.Martin., Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition. *Prentice-hall, USA, 2000.*

[4] B.Greene and G.Rubin., "Automatic grammatical tagging of English" *Department of Linguistics, Brown University, Providence, R.I., USA., 1971.*

[5] E. Brill, "A simple rule-based part of speech tagger" *Proceedings of the Twelfth International Conference on AI.(AAAI- 94),Seattle,WA, 1992.*

[6] S.J.DeRose., "Grammatical category Disambiguation by statistical optimization." *Computational Linguistics 14(1), 3139., 1988.*

[7] B. Daelemans and Gills, "A memory-based part of speech tagger generator." *Proceedings of the Fourth Workshop on Very Large Corpora, Copenhagen, Denmark, pp. 1427, 1996.*

[8] N. G. Marques, "A neural network approach to part-of-speech tagging" *Proceedings of the second workshop on spoken and written Portuguese, Curitiba, Brazil, p. 1-9, 1996.*

[9] H.Schmid, “Part-of-speech tagging with neural networks” *Proceeding of COLING-94. PP 172- 176, 1994.*

[10] El-Kareh and Al-Ansary., “An Arabic interactive multi-feature pos tagger.” *In Proceedings of the, ACIDCA conference, Monastir, Tunisia, pp 204- 210., 2000.Wu, C. and X.M. Wang, 2000.*

[11] S. Abuleil and M. Evens, “Discovering lexical information by tagging Arabic newspaper text” Workshop on Semitic Language Processing. *COLING-ACL.98, University of Montreal, Montreal, PQ, Canada, Aug 16 1998, pp 1-7.*

[12] S.KHOJA, “Apt: Arabic part-of-speech tagger” Proceedings of the Student Workshop at the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL2001), *Carnegie Mellon University, Pittsburgh, Pennsylvania. June 2001, no. 2.*

[13] K. H. Diab, Mona and D. Jurafsky, “Automatic tagging of Arabic text: From raw text to base phrase chunks,” *Proceedings of HLTNAACL, 2004.*

[14] G, Khojah and Knowels, “A tagset for the morphosyntactic tagging of Arabic,” Paper presented at Corpus Linguistics 2001, *Lancaster University, Lancaster, UK, March 2001.*

[15] Roger Garside, Geoffrey Leech, and Geoffrey Sampson (1987) The Computational Analysis Of English: a corpus-based approach. *Longman Group UK Limited.*

[16] Leech G, Wilson A 1996 Recommendations for the Morphosyntactic Annotation of Corpora EAGLES, *Report.*

[17] Transparent Language,
http://www.transparent.com/

Developing a tagset for automated POS tagging in Arabic

SHIHADEH ALQRAINY and ALADDIN AYESH
Centre for Computational Intelligence (CCI) - School of Computing
De Montfort University
Leicester – The Gateway
UNITED KINDOM
{alqrainy , aayesh}@dmu.ac.uk

Abstract: - Arabic language has much more syntactical and morphological information. Diacritics, which are marks placed over and below the letters of Arabic word, play a great role in adding linguistic attributes to Arabic word in part-of-speech tagging system. This paper describes a tagset that were built based on the inflectional morphology system which derived from traditional Arabic grammatical theory. The tagset developed represent an early stage of research related to automatic morphosyntactic annotation in Arabic language. This paper aims to present a general tagset for use in diacritics-based automated tagging system that is underdevelopment by the author.

Key-Words: - Part-of-Speech (POS), Arabic Language, Tagset, Diacritics, Syntactical, Morphological.

1 Introduction

A tag is a code which represents some features or set of features and is attached to the segment in a text. Single or complex information are carried by a tag [8]. In the case of POS Tagging, a POS tagset to categories and mark up the words of the target text is an absolutely necessary preliminary [3]. The development of a tagset to support diacritical based tagging system is at early stage. Little work has been done in developing Arabic tagset. The need for such a tagset comes from the fact that there is no standardized and comprehensive Arabic tagset.

An overview of Arabic language followed by diacritics in Arabic described in this paper. Tagset background and EAGLES guidelines overview presented. Finally we will present our tagset (Analysis and Hierarchy) followed by conclusion and future work.

2 Arabic Language

2.1 Background

The Arabic language is spoken in more than 20 Countries, from Egypt to Morocco and throughout the Arabian Peninsula. It is the native language of over 195 million people. Plus, at least another 35 million speak Arabic as a second language.

Modern Standard Arabic (MSA) is the official language throughout the Arab world, and its written form is relatively consistent across national boundaries. MSA is used in official documents, in educational settings, and for communication between Arabs of different nationalities. However, the spoken forms of Arabic vary widely, and each Arab country has its own dialect. Dialects are spoken in most informal settings, such as at home, with friends, or while shopping.

The Arabic language belongs to the Semitic family of languages, written from right to left. Arabic has been a literary language since the 6th century A.D., and is the liturgical language of Islam in its classical form.

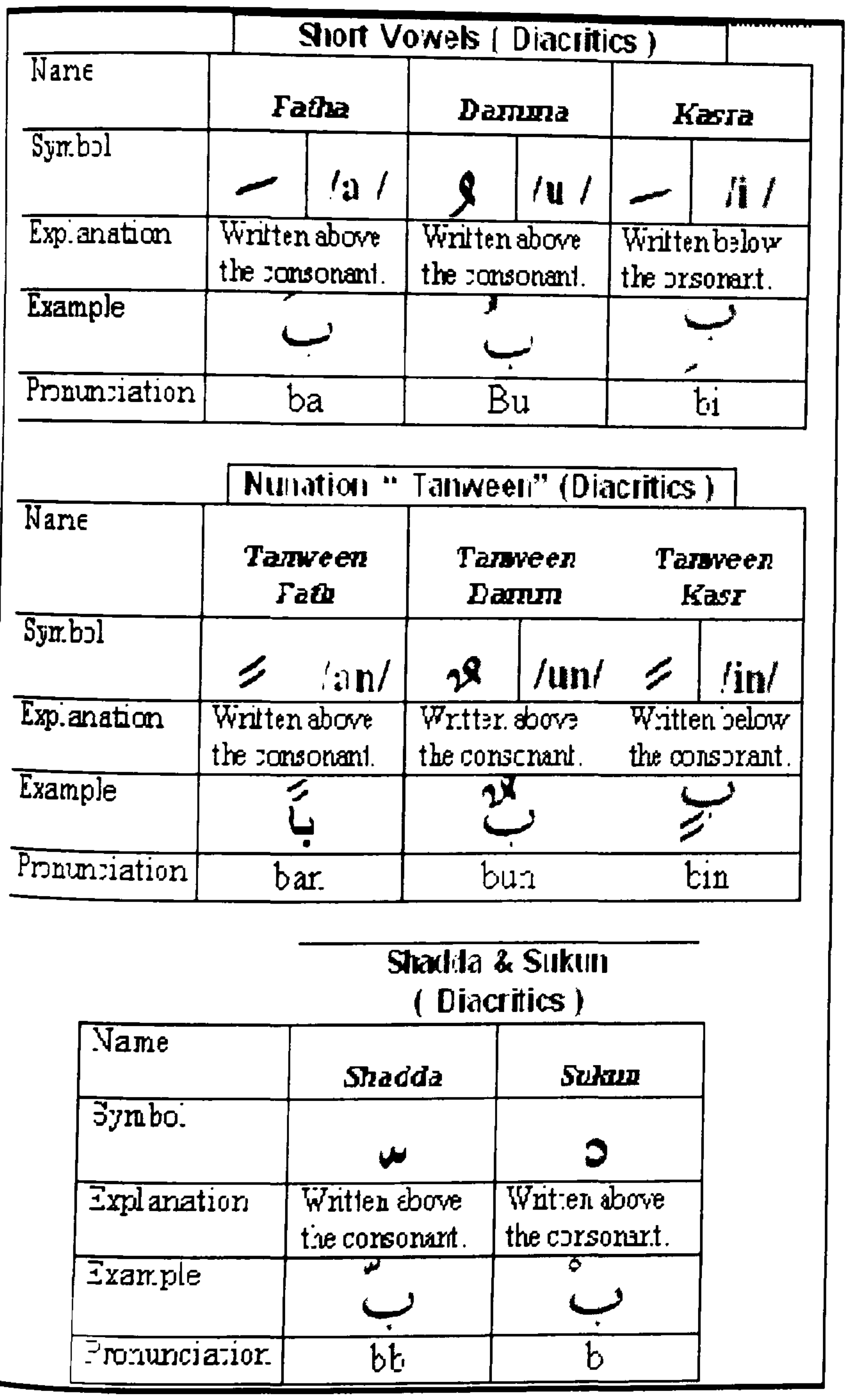
The Arabic writing system is quite different from the English system. The Arabic alphabet consists of 28 letters that change shape depending on their position within a word and the letters by which they are surrounded. Some Arabic letters must be connected to other letters; others may stand alone. Arabic vowels are indicated by marks (Diacritics) above and below the consonants. In many cases, these diacritics play the role of vowels in English and thus influence pronunciation. Additionally, there are no special forms, such as the use of capital letters in English, to indicate proper nouns or the beginning of a sentence [10].

2.2 Diacritics in Arabic

Diacritics are marks placed over and below the letters of Arabic word. This feature plays a great role in adding linguistic attributes to Arabic words which help us to assign the most likely tag of the word in POS tagging system and in indicating pronunciation and grammatical function of the words. It is particularly of interest for the purpose of this paper. Table 1 shows Arabic vowel diacritics.

The pronunciation of diacritized languages words cannot be fully determined by spelling their characters only; special marks are put above or below the characters (Diacritics) to determine the correct pronunciation and indicate the grammar function of the word within the sentence. For example, the word "كتب" without mark (Diacritic) may be pronounced to mean *"He wrote"* , *"It was written"*, *"books"*. The reader may refer to the context the word appears in to decide which of the words is actually intended. In such languages, two different words may have identical spelling whereas their pronunciations and meanings are totally different [2].

In Arabic, short vowels are not apart of the Arabic alphabet. They are used in both Noun and Verb in Arabic Language. They indicate the case of the noun and the mood of the verb.



Short Vowels (Diacritics)

Name	*Fatha*		*Damma*		*Kasra*	
Symbol	ـَ	/a /	ـُ	/u /	ـِ	/i /
Explanation	Written above the consonant.		Written above the consonant.		Written below the consonant.	
Example	بَ		بُ		بِ	
Pronunciation	ba		Bu		bi	

Nunation " Tanween" (Diacritics)

Name	*Tanween Fath*		*Tanween Damm*		*Tanween Kasr*	
Symbol	ـً	/an/	ـٌ	/un/	ـٍ	/in/
Explanation	Written above the consonant.		Written above the consonant.		Written below the consonant.	
Example	بً		بٌ		بٍ	
Pronunciation	ban		bun		bin	

Shadda & Sukun (Diacritics)

Name	*Shadda*	*Sukun*
Symbol	ـّ	ـْ
Explanation	Written above the consonant.	Written above the consonant.
Example	بّ	بْ
Pronunciation	bb	b

Table 1: Arabic vowel diacritics

3 Arabic Tagset and EAGLES guidelines

EAGLES [9] guidelines outline a set of features for Tagsets, these guidelines were designed to help standardize tagsets for what were then the official languages of the European Union.

EAGLES tags are defined as sets of morphosyntactic attribute-value pairs (e.g. Gender is an attribute that can have the values Masculine, Feminine or Neuter)[3]. The tagset discussed here is not being developed in accordance with the EAGLES guidelines for morphosyntactic annotation of corpora. Arabic is very different from the languages for which EAGLES was designed, and belongs to the Semitic family rather than the Indo-European one.

Following a normalized tagset and the EAGLES recommendations would not capture some of Arabic relevant information, such as the jussive mood of the verb and the dual number that are integral to Arabic. Another important aspect of Arabic is inheritance, where all subclasses of words inherit properties from the classes from which they are derived. For example, all subclasses of the noun inherit the "Tanween" nunation when in the indefinite which is one of the main properties of the noun [7].

3.1 Previous work on POS tagsets

There are numbers of popular tagsets for English, such as : 87-tag tagset used Brown Corpus , 45-tag Penn Treebank tagset and 61-tag C5 tagset, TOSCA tagset, ICE tagset, LUND tagset [5][3]. For Arabic also very small number of tagset had been built, El-Kareh S, Al-Ansary [1] described the tagset ,they classifying the words into three main classes, Verbs are sub classified into 3 subclasses; Nouns into 46 subclasses and Particles into 23 subclasses. Shereen Khoja [7] described more detail tagset. Her tagset contains 177 tags, 57 Verbs, 103 Nouns, 9 Paricles, 7 residual and 1 punctuation.

3.2 Proposed Arabic Tagset: Analysis

It is necessary to have a model of the language to create the linguistic categories of a tagset. An ideal approach would be to derive this model from the grammatical description of the language.

Since the grammar of Arabic has been standardized for centuries, it is logical to derive our morphosyntactic Arabic tagset from this grammatical tradition that has been used for around fourteen centuries by all students of Arabic.

Arabic grammarians and linguists have always used the Arabic system of inflectional morphology called "الإعراب" when teaching Arabic grammar to students. For example, given the sentence " لعبَ الولدُ " "the boy played", students would have to say that the first word is the indeclinable, indicative, perfect verb, while the second word is the nominative subject[7][3].

The proposed Arabic tagset in this paper is based on the inflectional morphology system. Arabic grammarians traditionally analyses all Arabic words into three main parts-of-speech. These parts-of-speech are further sub-categorised into more detailed parts-of-speech which collectively cover the whole of the Arabic language [4]. These are:

- **Noun:** A noun in Arabic is a name or a describing-word for a person, a thing or an idea. This includes not only the English equivalent of a noun, but also adjectives, proper nouns and pronouns.
- **Verb:** Verb: Verbs are the same in Arabic as they are in English in that they denote actions.
- **Particle:** Particles include prepositions, conjunctions, Exceptions, Vocative, Annulment, Subjunctive, and Jussive.

3.2.1 Noun

A noun in Arabic indicates a meaning by itself without being connected with the notion of time and refers to a person, place, thing, event, substance or quality.

Nouns are also divided into the following types: (Common, Demonstrative, Relative, Personal, Adverb, Diminutive, Instrument, Conjunctive, Interrogative, Proper, and Adjective).

The linguistic attributes of nouns that have been used in this tagset are:

- **Case:** Arabic nouns have three cases: nominative, accusative and genitive. For example, the words " الدرسُ الدرسَ ، الدرسِ " which mean "*the lesson*", indicate the above three cases respectively.

Without the case marker associated with the last letter of the above words (e.g short vowels), it's difficult to determine the case of that word.

- **State**: Arabic nouns are marked for definiteness and indefiniteness. Definiteness is marked by the article "ال", which means" *the*". For example, the words "الكتاب" and "كتاب" which mean" *the book*"," *a book*" indicate definiteness/indefiniteness respectively.

- **Number:** Arabic has three numbers: singular, dual, and plural. For example, the words "ولد","ولدان" and "أولاد" which mean " *a boy* ", " *two boys* " and "*boys* " indicate singular, dual, and plural respectively.

- **Gender:** Arabic nouns have three genders: masculine, feminine and neuter. Most common noun ends with "Tanween". Most feminine singular nouns end with a round Ta (marbuta). For example, the words "ملك","طائرة" and "جماعة" , which mean " *a king* ", " *a plane* " and " *group of people* " indicate masculine, feminine and neuter respectively.

- **Person:** Arabic nouns have three persons: the speaker (First person), the individual spoken to (Second person), and individual spoken of (third person). For example, the personal noun and "أنا" which mean " *I* ", " *You* " and " *He* " indicate First, Second, and third person respectively.

3.2.2 Verb

Arabic verbs are deficient in tenses. Moreover, these tenses do not have accurate time significances as in Indo-European languages [6].

The verb in the Arabic language implies a state or action and a notion of time combined with them and has several aspects: Perfect, Imperfect and Imperative. The Perfect verb indicates a state or a fact in the past. For example, the word "كتبَ" which means "*He wrote*". The Imperfect verb expresses an action still unfinished at the time to which reference is being made. For example, the word "يأكلُ" which means "*He is writing*". The Imperative verb indicates an action demanded to be carried out in the future. For example, the word "أكتب" which means "*you write*".

The linguistic attributes of Verbs that have been used in this tagset are:

- **Mood:** Arabic Verbs have three moods: Indicative, Subjunctive and Jussive. In Verbs, the words "كتبَ", "كتبتُ" and "كتبتِ" which mean " *He wrote* " , " *I wrote* " and " *You wrote* " indicate Indicative, Subjunctive, Jussive mood respectively.

- **Number:** Arabic has three numbers: singular, dual, and plural. For example, the words "قرأ", "يقرأن" and "قرأوا" which mean " *He read* ", " *(two people) read* " and " *they read* " indicate singular, dual, and plural number respectively.

• **Gender:** Arabic verbs have two genders: masculine, feminine. For example, the words "كتب" and "كتبت" which mean " He wrote "and " She wrote ".

• **Person:** Arabic verbs have three persons: the speaker (First person), the individual spoken to (Second person), and individual spoken of (third person). For example, the words which mean the words "كتب", "كتبت" and "كتبت" which mean " *He wrote* " , " *I wrote* " and " *You wrote* " indicate First, Second, and third person respectively.

3.2.3 Particle

In Arabic, particles are classified as one of the three main categories as part of speech, some of the particles used with Verbs and effective the mood of verb when precedes the Verb word. For example, the particles "لم" (Jussive), "كي" (Subjunctive), some of them used with Nouns. For example, the particles "في" (Preposition), "إلا" (Exception), and some used with both the noun and the verb. For example, the particle "و" (Conjunction).

3.3 Proposed Arabic Tagset: Hierarchy

We have based our Arabic tagset on inflectional morphology system. The traditional description of Arabic grammarians consider as a base to create the linguistic categories of Arabic tagset. Arabic grammarians describe Arabic as being derived from three main categories: noun, verb and particle. Figure 1 shows the tagset hierarchy.

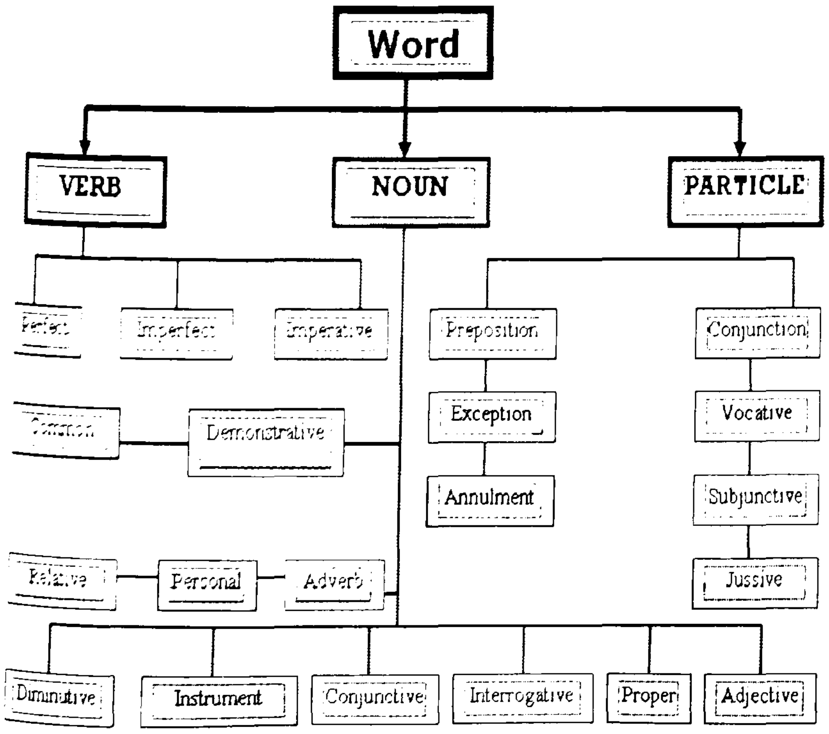


Fig. 1: Tagset Hierarchy.

The tagset has the following main formula:
[T , S , G , N , P , M , C , F] , Where:

T (Type) = {Verb, Noun, Particle}
S = Sub-Class {*Common, Demonstrative, Relative, Personal, Adverb,Diminutive,Instrument, Conjunctive, Interrogative, Proper and Adjective*}
G (gender)= {Masculine, Feminine, Neuter}
N (Number) = {Singular, Plural, Dual}
P (Person) = {First, Second, Third}
M (Mood) = {Indicative, Subjunctive, Jussive}
C (Case) = {Nominative, Accusative, Genitive}
F (State) = {Definite, Indefinite}

Figure 2 shows the Abbreviations which was used to define the words in our tagset.
A sample of our tagset shown in Table 2.

Word	Abb	Word	Abb
Verb	Ve	Annulment	An
Noun	Nu	Subjunctive	Sb
Particle	Pr	Masculine	Ma
Perfect	Pe	Feminine	Fe
Imperfect	Pi	Neuter	Ne
Imperative	Pm	Singular	Sn
Common	Cn	Plural	Pl
Adjective	Aj	Dual	Du
Demonstrative	De	First	Fs
Relative	Re	Second	Sc
Personal	Ps	Third	Th
Diminutive	Dm	Indicative	Dc
Instrument	Is	Subjunctive	Sj
Proper	Pn	Jussive	Js
Adverb	Ad	Nominative	Nm
Interrogative	In	Accusative	Ac
Conjunction	Cj	Genitive	Ge
Preposition	Pp	Definite	Df
Vocative	Vo	Indefinite	Id
Conjunction	Co		
Exception	Ex		

Fig. 2: Tagset Abbreviations

Let us try to explain the symbols of the tagset formula for a moment.
The symbols **[T , S , G , N , P , M]** consider as linguistic attributes for class Verb, while the symbols **[T , S , G , N , P , C , F]** consider as linguistic attributes for class Noun. For example , the word " كتب " which means " *He wrote* " has the following tag [**VePeMaSnThSj**], which means [*Perfect Verb , Masculine Gender , Singular Number , Third Person , Subjunctive Mood*].

4 Conclusion and Future Work

In this paper, we described a morphosyntactic tagset that is derived from the ancient Arabic grammar, which is based on Arabic system of inflectional morphology. The tagset represent an early stage for use in a word-class based automated tagging system that is underdevelopment by the author. The tagset does not follow the traditional Indo-European tagset that is based on Latin but is instead based on the Semitic tradition of analyzing language.

These tags contain a large amount of information and add more linguistic attributes to the word. Also, we are currently expanding our tagset to cover most categories word in Arabic.

Tag	Description
VePeMaSnThSj	Verb, Perfect, Masculine, Singular, Third Person, Subjunctive
VePeMaSnFsDc	Verb, Perfect, Masculine, Singular, First Person, Indicative
VePeMaSnSeSj	Verb, Perfect, Masculine, Singular, First Person, Subjunctive
VePeFeSnSeJs	Verb, Perfect, Feminine, Singular, Second Person, Jussive
VePeFeSnThJs	Verb, Perfect, Feminine, Singular, Third Person, Jussive
VePeNeDuSeSj	Verb, Perfect, Neuter, Dual, Second Person, Subjunctive
VePeMaDuThSj	Verb, Perfect, Masculine, Dual, Third Person, Subjunctive
VePeFeDuThSj	Verb, Perfect, Feminine, Dual, Third Person, Subjunctive
VePeMaPlFsSj	Verb, Perfect, Masculine, Plural, First Person, Subjunctive
VePeMaPlSeJs	Verb, Perfect, Masculine, Plural, Second Person, Jussive
VePeFePlSeJs	Verb, Perfect, Feminine, Plural, Second Person, Subjunctive
VePeFePlThJs	Verb, Perfect, Feminine, Plural, Third Person, Subjunctive
VePeMaPlThDc	Verb, Perfect, Masculine, Plural, Third Person, Indicative
VePiMaSnThDc	Verb, Imperfect, Masculine, Singular, Third Person, Indicative
VePiMaSnFsDc	Verb, Imperfect, Masculine, Singular, First Person, Indicative
VePiFeSnThDc	Verb, Imperfect, Feminine, Singular, Third Person, Indicative
VePiNePLFsDc	Verb, Imperfect, Neuter, Plural, First Person, Indicative
VePiMaDuThJs	Verb, Imperfect, Masculine, Dual, Third Person, Jussive
VePiFeDuSeJs	Verb, Imperfect, Masculine, Dual, Third Person, Jussive
VePiMaPlThSj	Verb, Imperfect, Masculine, Plural, Third Person, Subjunctive
VePiFePlThSj	Verb, Imperfect, Feminine, Plural, Third Person, Subjunctive
VePmMaSnSeJs	Verb, Imperative, Masculine, Singular, Second Person, Jussive
VePmNeDuSeSj	Verb, Imperative, Neuter, Dual, Second Person, Subjunctive
VePmFePlSeSj	Verb, Imperative, Feminine, Plural, Second Person, Subjunctive
VePmMaPlSeSj	Verb, Imperative, Feminine, Plural, Second Person, Subjunctive
NuAjMsSnNmId	Adjective Noun, Masculine, Singular, Nominative, Indefinite
NuAjMsSnAcId	Adjective Noun, Masculine, Singular, Accusative, Indefinite
NuAjMsSnGeId	Adjective Noun, Masculine, Singular, Genitive, Indefinite
NuAjMsSnNmDf	Adjective Noun, Masculine, Singular, Nominative, Definite
NuAjMsSnAcDf	Adjective Noun, Masculine, Singular, Accusative, Definite
NuAjMsSnGeDf	Adjective Noun, Masculine, Singular, Genitive, Definite
NuAjMsDuGeId	Adjective Noun, Masculine, Dual, Genitive, Indefinite
NuAjMsDuGeDf	Adjective Noun, Masculine, Dual, Genitive, Definite
NuAjFeSnNmId	Adjective Noun, Feminine, Singular, Nominative, Indefinite
NuAjFeSnAcId	Adjective Noun, Feminine, Singular, Accusative, Indefinite
NuAjFeSnGeId	Adjective Noun, Feminine, Singular, Genitive, Indefinite
NuAjFeSnNmDf	Adjective Noun, Feminine, Singular, Nominative, Definite
NuAjFeSnAcDf	Adjective Noun, Feminine, Singular, Accusative, Definite
NuAjFeSnGeDf	Adjective Noun, Feminine, Singular, Genitive, Definite
NuAjFeDuGeId	Adjective Noun, Feminine, Dual, Genitive, Indefinite
NuAjFeDuGeDf	Adjective Noun, Masculine, Dual, Genitive, Definite
NuAjMaPlAcId	Adjective Noun, Masculine, Plural, Accusative, Indefinite
NuAjMaPlGeId	Adjective Noun, Masculine, Plural, Genitive, Indefinite
NuAjMaPlNmId	Adjective Noun, Masculine, Plural, Nominative, Indefinite

NuAjMaPlNmDf	Adjective Noun, Masculine, Plural, Nominative, Definite
NuAjMaPlAcDf	Adjective Noun, Masculine, Plural, Accusative, Definite
NuAjMaPlGeDf	Adjective Noun, Masculine, Plural, Genitive, Definite
NuAjFePlNmId	Adjective Noun, Feminine, Plural, Nominative, Indefinite
NuAjFePlAcId	Adjective Noun, Feminine, Plural, Accusative, Indefinite
NuAjFePlGeId	Adjective Noun, Feminine, Plural, Genitive, Indefinite
NuAjFePlNmDf	Adjective Noun, Feminine, Plural, Nominative, Definite
NuAjFePlAcDf	Adjective Noun, Feminine, Plural, Accusative, Definite
NuAjFePlGeDf	Adjective Noun, Feminine, Plural, Genitive, Definite
NuIsMaSnNmId	Instrument Noun, Masculine, Singular, Nominative, Indefinite
NuIsMaDuGeId	Instrument Noun, Masculine, Dual, Genitive, Indefinite
NuIsMaPlNmId	Instrument Noun, Masculine, Plural, Nominative, Indefinite
NuIsMsSnNmDf	Instrument Noun, Masculine, Singular, Nominative, Definite
NuIsMsSnAcDf	Instrument Noun, Masculine, Singular, Accusative, Definite
NuIsMsSnGeDf	Instrument Noun, Masculine, Singular, Genitive, Definite
NuIsMaDuGeId	Instrument Noun, Masculine, Dual, Genitive, Indefinite
NuIsMaPlNmDf	Instrument Noun, Masculine, Plural, Nominative, Definite
NuIsMaPlAcDf	Instrument Noun, Masculine, Plural, Accusative, Definite
NuIsMaPlNmDf	Instrument Noun, Masculine, Plural, Genitive, Definite
PrPp	Preposition Particle
PrVo	Vocative Particle
PrCo	Conjunction Particle
PrEx	Exception Particle
PrAn	Annulment Particle
PrSb	Subjunctive Particle

Table 2: Sample of Arabic Tagset

Reference:

[1] El-Kareh and Al-Ansary, An Arabic interactive multi-feature pos tagger. *In Proceedings of the, ACIDCA conference*, Monastir, Tunisia, 2000, pp 204- 210.

[2] M. A. Elaraby 2000, A large scale computational processor of the Arabic morphology and application. *(Master's thesis), Cairo University, Egypt.*

[3] Andrew Hardie. Developing a tagset for automated Part-of-speech tagging in Urdu. *Proceedings of the Corpus Linguistics 2003 conference,* Lancaster University, UK, 2003.

[4] J. A. Haywood and H. M. Nahmad. *A new Arabic Grammar: of the written language*, LUND HUMPHRIES , USA, 2005.

[5] Daniel Jurafsky & James H.Martin. Speech and language processing: *An introduction to natural language processing, computational linguistics, and speech recognition.* Prentice-hall, USA., 2000.

[6] S. KHOJA, Apt: Arabic part-of-speech tagger. *Proceedings of the Student Workshop at the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL2001),* Carnegie Mellon University, Pittsburgh, Pennsylvania, no. 2, 2001 .

[7] Graside, Khojah and Knowels, A tagset for the morphosyntactic tagging of Arabic. *Paper presented at Corpus Linguistics 2001, Lancaster University, Lancaster, UK, March 2001*, and to appear in a book entitled "A Rainbow of Corpora: Corpus Linguistics and the Languages of the World", edited by Andrew Wilson, Paul Rayson, and Tony McEnery; Lincom-Europa, Munich., 2001.

[8] B Megyesi. Brill's rule-based part of speech tagger for Hungarian. D-level thesis *(Master's thesis) in Computational Linguistics, Stockholm University, Sweden.* 1998.

[9] Leech G, Wilson A 1996 *Recommendations for the Morphosyntactic Annotation of Corpora EAGLES Report.* http://www.ilc.pi.cnr.it/EAGLES96/annotate/

[10] *Transparent Language* http://www.transparent.com/ .