DEVELOPMENT OF SAN'ANI ARABIC

PARTS-OF-SPEECH TAGGER: A BI-GRUs-CRF MODEL

A thesis submitted to the University of Hyderabad in partial fulfilment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

IN

APPLIED LINGUISTICS

BY

Sabah Mohammed Mohammed Nasser Al-Shehabi

Reg. No: 15HAPH08

CENTRE FOR APPLIED LINGUISTICS AND TRANSLATION STUDIES
SCHOOL OF HUMANITIES
UNIVERSITY OF HYDERABAD
HYDERABAD, INDIA
DECEMBER, 2021

CENTER FOR APPLIED LINGUISTICS AND TRANSLATION STUDIES

UNIVERSITY OF HYDERABAD

Certificate

This is to certify that the thesis entitled "**DEVELOPMENT OF SAN'ANI ARABIC PARTS-OF-SPEECH TAGGER: A BI-GRUs-CRF MODEL"** submitted by Mrs. Sabah Mohammed Mohammed Nasser Al-Shehabi (15HAPH08) in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy in Applied Linguistics from the University of Hyderabad is a bona fide research work carried out by her under my supervision and guidance.

This thesis is free from plagiarism and has not been submitted previously, either in part or in full for the award of any academic degree or diploma to this or any other university or institution.

The student has the following publication(s) before submission of the thesis for adjudication and has produced evidence for the same in the form of acceptance letter or the reprint in the relevant area of her research:

1. Al-Shehabi, Sabah and Sharaf Addin, Mohammed. 2020. "A Grammatically Annotated Corpus for Sana'ani Arabic Dialect." *Test Engineering and Management*. Vol. 83: 4953 - 4961. The Mattingley Publishing Co., Inc.

2. Sharaf Addin Mohammed, Al-Shehabi Sabah. 2020. "Developing Social-Media Based Text Corpus for San'ani Dialect (SMTCSD)." In: Satapathy S.C., Raju K.S., Shyamala K., Krishna D.R., Favorskaya M.N. (eds) *Advances in Decision Sciences, Image Processing, Security and Computer Vision. Learning and Analytics in Intelligent Systems,* vol 3. Springer, Cham.

3. Al-Shehabi, Sabah, Mohammed Sharaf-Addin, K Rajyarama. 2021. "Pre-Processing and Annotation of Social Media Text for San'ani Arabic POS Tagging System." (accepted) *Aligarh Journal of Linguistics*, Vol. 11.

The student has made presentation in the following conferences:

1. Sharaf Addin Mohammed, Al-Shehabi, Sabah. 2019. "Developing Social-Media Based Text Corpus for San'ani Dialect (SMTCSD)." *International Conference on Emerging Trends in Engineering (ICETE)* held during 22-23 March 2019. Organized by University College of Engineering, Osmania University, Hyderabad.

2. Al-Shehabi, Sabah and Sharaf Addin, Mohammed. 2020. "A Grammatically Annotated Corpus for Sana'ani Arabic Dialect." *International Conference on Recent Challenges in Science, Engineering and Technology (ICRCSET-2020*) held during 28th & 29th February 2020, organized by Chalapathi Institute of Technology (CIT), Guntur, Andhra Pradesh.

Further the student has passed the following courses towards the fulfilment of the coursework requirement for Ph.D. in 2016-17.

Course Code	Course Name	Credits	Pass/Fail
AL-801	Research Methodology	04	Pass
AL-802	Current Trends in Applied Linguistics	04	Pass
AL821	Readings in Applied Linguistics	04	Pass
AL-831	Academic Writing for Doctoral Students	04	Pass

Prof. K. Rajyarama
Supervisor, CALTS
University of Hyderabad

Director

(CALTS)
University of Hyderabad

Dean

School of Humanities
University of Hyderabad

CENTER FOR APPLIED LINGUISTICS AND TRANSLATION STUDIES

UNIVERSITY OF HYDERABAD

Declaration

I, Sabah Mohammed Mohammed Nasser Al-Shehabi, herby declare that this title entitled "**DEVELOPMENT OF SAN'ANI ARABIC PARTS-OF-SPEECH TAGGER: A BI-GRUs-CRF MODEL"** submitted to the University of Hyderabad in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy in Applied Linguistics embodies veritable research work carried out by me under the guidance and supervision of Prof. K. Rajyarama, Centre for Applied Linguistics & Translation Studies, School of Humanities, University of Hyderabad. It is a research work which is free of plagiarism.

I also declare to the best of my knowledge that this thesis has not been submitted previously, in part or in full, to this or any other university or institute for the award of any academic degree. I hereby agree that my thesis can be deposited in Shodhgana/INFLIPNET.

A report of the plagiarism statistics from Indira Ghandi Memorial Library, University of Hyderabad is enclosed.

Date: December 26, 2021
Place: Hyderabad

Sabah Mohammed M. Nasser Al-Shehabi
Reg. no. 15HAPH08
Centre for ALTS

Countersigned by:

Professor K. Rajyarama
Supervisor, CALTS
University of Hyderabad

Dedication

I dedicate this thesis

To the soul of my baby girl "Raghad" a piece of me living in heaven. Mammy will never forget you,

To my exceptional parents who encouraged my dreams

To my sisters and brothers,

To my dear husband and lovely daughter "Jana"

for their love, encouragement, and endless support

Acknowledgement

First and foremost, my continuous thanks are due to Almighty Allah, the most gracious and the most merciful who blessed me with everything I ever needed to complete this thesis and still showering me with unlimited care, guidance, strength, and blessings all through the way. Additionally, may Allah's peace and blessing be upon The Prophet Mohammed, his family, and his companions.

I also would like to express my sincere gratitude and appreciation to my esteemed supervisor, professor K. Rajyarama who bestowed me with invaluable advice, continuous support, and patience. Thank you for providing positive encouragement and always having an open door and listening ear. It has been a great pleasure and honour to have you as my supervisor.

I owe my deepest gratitude to my research advisory members, Professor Uma Maheshwar Rao and Dr. Parameswari K., for their illuminating discussions and insightful recommendations. I consider myself very lucky to come across their knowledge and guidance.

I extend my sincere thanks to The Center of Applied linguistics and Translation Studies (CALTS), including the Director Prof. Bhimrao Panda Bhosale, and all the staff members of the CALT for their academic support and valuable suggestions. I am also thankful to the non-teaching staff for their whole-hearted support and co-operation.

My special thanks to Prof. N. Siva Kumar, Director of International affairs, Mr. Satyanarayana Murthy M, Coordinator of International Programs, and Mrs. Vadhana Ramanan, International Program Officer, for their valuable assistance and constant help and co-operation.

My special thanks to Dr. Manish Shrivastava at LTRC, IIIT, Hyderabad, my mentor for the Internship program held at LTRC, IIIT, Hyderabad. His insightful deliberation and assignments' recommendations were high-yielding.

My deepest gratitude to my father and mother for their unconditional support, love, encouragement, faith, and prayers but above all, thank you for painting my dreams bright and allowing

my wings to spread wide and fly high. I also wish to express my deepest thanks to my elder brothers, Mr. Mansour, Mr. Hani, and Mr. Yasser, who encouraged me thoroughly for my study. I owe a great deal to my sisters, Ms. Jameelah, Ms. Abeer, Dr. Shumoa, and Ms. Nowf, for their eternal care, understanding, and motivation.

I also would like to thank my family-in-law for being supportive and loving. My sincere thank are due to my dear relatives and friends. Their prayers and well wishes will never be forgotten.

I want to express my deepest gratitude to my beloved dream mate, husband, and best friend, Dr. Mohammed Sharaf Addin, for sharing our journey of identical dreams. Though he was working on his Ph.D. research, he never denied me academic and technical assistance.

It was his fruitful discussions, and insightful comments that saved me from wrong turns. Thank you for taking the time to read for me and support me in every way. The sweetest thanks are due to my baby girl and princess, Jana. Since her birth, she has become my absolute joy and sunshine, brightening my every day. Thank my small family for your presence, encouragement, and support, which sustained me this far.

I gratefully acknowledge the funding received towards my Ph.D. from the Indian Council for Cultural Relationship (ICCR) Ph.D. scholarship and my home university, Sana'a University, for their partial financial support.

Finally, I would like to acknowledge all those who helped me during this research work and have directly or indirectly contributed to the completion of this thesis. Thank you one and all.

Table of Contents

List of Tables

List of Figures

IPA Symbols Used in the Transcription

Consonants		Vowels	
Arabic Script	**IPA symbol**	**Arabic Script**	**IPA symbol**
ء	ʔ		a:
ب	b	و	u:
ت	t	ي	i:
ث	θ	◌َ فتحة	a
ج	dʒ	◌ِ كسرة	i
ح	ħ	◌ُ ضمة	u
خ	x		
د	d		
ذ	ð		
ر	r		
ز	z		
س	s		
ش	ʃ		
ص	sˤ		
ض	dˤ		
ط	tˤ		
ظ	ðˤ		
ع	ʕ		
غ	ɣ		
ف	f		
ق	g		
ك	k		
ل	l		
م	m		
ن	n		
ه	h		
و	w		
ي	j		

List of Abbreviations

1	First
2	Second
3	Third
ACL	Arabic Computational Linguistics
AMT	Arabic Morphosyntactic Tagger
ANNs	artificial neural networks
APT	automatic Arabic Parts of speech tagger
ATB	Arabic Tree Bank
BAMA	Buckwalter's morphological analyser
BI-GRUs	Bidirectional Gated Recurrent Units
BI-GRUs-CRF	Bidirectional-Gated Recurrent Units-Conditional Random Field
BI-LSTM	bidirectional–Long Short-Term Memory
BI-RNN	Bidirectional Recurrent Neural Networks
BNC	British National Corpus
BPNN	back-propagation neural network
BTEC	Basic Traveling Expression Corpus
CA	Classical Arabic
CATiB	Columbia Arabic Treebank
CGC	computational Grammatical Coder
CL	Computational Linguistics
CLAWS1	Constituent-Likelihood Automatic Word-Tagging System
CNN	Convolutional Neural Networks
CODA	conventional orthography for dialectal Arabic
CRF	Conditional Random Fields
DA	Dialectal Arabic
EA	evolutionary algorithms
ECA	Egyptian Colloquial Arabic
EGYA	Egyptian Arabic
ERTS	Extended Reduced Tagset
F	Feminine
GFA	Gulf Arabic

GRUs	Gated Recurrent Units
HMM	Hidden Markov Model
HSNA	Hassaniya Arabic
IRQA	Iraqi Arabic
LDC	Linguistic Data Consortium
LMNN	Levenberg-Marquardt neural network
LOB	Lancaster-Oslo/Bergen
LSTM	Long Short-Term Memory
LVA	Levantine Arabic
M	Masculine
MBL	Memory-based learning
ME	Maximum Entropy
MGHBA	Maghrebi Arabic
MLP	Multilayer Perceptron
MSA	Modern Standard Arabic
NER	Named Entity Recognition
NLP	Natural Language Processing
OOV	out of vocabulary
PATB	Penn Arabic Treebank
PL	Plural
POS	Parts of speech
POST	Parts-of-Speech Tagging
RNN	Recurrent Neural Network
RTS	Reduced Tag Set
SAMA	Standard Arabic Morphological Analyzer
SDNA	Sudanese Arabic
SG	Singular
STTS	(German Standard) Stuttgart/Tübinger Tagset
SVMs	Support Vector Machines
TBL	Transformational-based learning
TBR	Transformational-based Retagging
TTR	Token to Type Ratio
YMNA	Yemeni Arabic

Abstract

One of the essential pre-processing tasks for building and improving NLP applications is known as parts-of-speech tagging. The tagging process involves the assigning of an appropriate part of speech tag to each word/token in a text. It also plays a fundamental role in developing many natural language processing applications such as syntactic parsing, named-entity recognition, automatic translation, ontology engineering, question answering, and information retrieval.

In Arabic Natural Language Processing (NLP), the undivided attention of research was directed to Modern Standard Arabic (MSA) and occasionally Classical Arabic. The main bulk of research placed MSA in the spotlight, avoiding other Arabic forms. However, during the last decade, the situation has been changing. The prevalence of dialectal interchange through social media platforms gradually drives attention towards Arabic dialects.

Nowadays, work on dialectal Arabic NLP is still in elementary stages due to several challenges, including the paucity of data and resources. Such challenges, among others, influence the selection of the dialect/s to work on. As a matter of fact, Egyptian, Gulf, and Levantine dialects are primarily targeted while other Arabic dialects are barely touched. San'ani Arabic is no exception in this regard. The tools developed for processing San'ani Arabic dialectal text are almost not available.

In this thesis, we describe the process of developing a novel parts-of-speech tagger for San'ani Arabic. We adopted an innovative deep learning model which utilizes a Recurrent Neural Network (RNN) variant and a stochastic classifier. This model is known as the Bidirectional-Gated Recurrent Units-Conditional Random Field (BI-GRUs-CRF) model. To train the tagger, we had to overcome the challenge of data paucity, so we developed, pre-processed, and manually

annotated a social media-based corpus of 200,000 tokens of San’ani Arabic. The tagger was tested using 11,000 tokens of new/unseen data. The overall accuracy reported is 85.8%.

CHAPTER ONE
INTRODUCTION

1.1 Background

The main purpose of Computational Linguistics (CL) is the logical modelling of natural languages through the combination of the theoretical knowledge of Linguistics and the practical application of computer science. Computational linguistics deals with the automatic processing of natural languages (Abumalloh et al. 2016). One of the basic areas in the field of computational linguistics is parts-of-Speech tagging (POST/ POS tagging). parts-of-speech tagging plays an essential role in language processing, especially corpus annotation. It is considered as the basic need for most of the Computational linguistics applications (Abumalloh et al. 2016; Gahbiche-Braham et al. 2012; Mohamed and Kubler 2010; and Habash 2010). In literature, the term parts-of-speech tagging was defined extensively. A well-known definition of parts-of-speech tagging is given by Jurafsky and Martin (2000, 296) in their remarkable book "Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition". The definition is as follows "Part-of-speech tagging (or just tagging for short) is the process of assigning a parts-of-speech or other lexical class marker to each word in a corpus." This definition covers the concept of parts-of-speech tagging generally.

For the present study, parts-of-speech tagging can be defined as the process of assigning an appropriate parts-of-speech tag, such as noun, verb, adjective, etc., to each word/token in a natural language text based on its lexical and syntactic structure in the context. This process

results in enriching the raw text with grammatical annotation. To perform grammatical annotation, a set of parts of speech tags known as tagset is needed. A tagset can be defined as a group of tags that exhaustively cover a natural language's various parts of speech.

As parts-of-speech tagging is a fundamental step for computational linguistics systems, active research in the area has been ongoing in recent years (Abumalloh et al. 2016). Some of these applications that utilize parts-of-speech tagging are machine translation, parsing, information extraction, information retrieval, digital dictionaries, Speech Synthesis Systems, and Word Processing. Such wide use of parts-of-speech tagging tools implies the need for tagging tools capable of providing accurate natural text annotation.

Nowadays, natural language processing (NLP), the field that deals with all different types of handling natural languages computationally, demands the existence of corpora, which can be written or spoken. It can further be dealt with as annotated or parallel. Annotated corpora reflect some linguistic information about the language structure. In contrast, parallel corpora consist of the same text but in two or more different languages where at least one of the corpora is annotated to analyze the other corpora. These types are valuable sources for different basic language processing techniques, such as parts-of-speech tagging, which can be used to develop CL applications (Jurafsky & Martin 2000). A tagged corpus is more useful than an untagged corpus because more information can be used for theoretical and practical analysis.

Arabic Computational Linguistics (ACL) is a challenging area of research. It lacks a lot of essential resources. It also needs advancement to reach the standard level of English Computational linguistics systems. Rabiee (2011) reports that computational linguistics resources and applications of the Arabic language are less in number. They also need

improvement on different levels, which is difficult as most of the available resources and applications are either in close projects or are not freely available. Besides, Abumalloh et al. (2016) state that no free source corpus is available for Arabic, though there are some Arabic corpora created in the field such as LDC Arabic newswire corpus, Hayat newspaper corpus, Buckwalter Arabic Corpus, Penn Arabic Treebank Corpus, and some others. These pieces of evidence show the growing need for extensive research and development in the area.

In addition, there is a coverage issue as most ACL tools and systems targeted either modern standard Arabic or Classical Arabic. Little attention has been given to the different dialects of Arabic in the area of ACL. Habash (2010) states that the relationship between MSA and dialects of Arabic is based on two facts. The first is that both are very different from each other, and The Second is MSA is not the native language of any Arab speaker. These two facts emphasize the need for specialized NLP tools and systems for Arabic dialects.

Since it is not possible to work on all different dialects of Arabic, the present research aims at providing an automatic parts-of-speech tagging tool for one dialect of Arabic, which is San'ani Arabic, along with a reasonable size annotated corpus of the same variety. We adopted an innovative deep learning model that utilizes a Recurrent Neural Network (RNN) variant and a stochastic classifier to develop the tagger. This model is known as the bidirectional-Gated Recurrent Units-Conditional Random Field (BI-GRUs-CRF) model.

1.2 Arabic Language and Arabic Dialects

Arabic language today grabbed researcher in the NLP community as it is the native language of over 300 million people in twenty-six different countries and the liturgical language for over 1.2 billion Muslims throughout the world. The literary language, called Modern Standard Arabic (MSA), is the official form of Arabic. It differs from Indo-European languages

morphologically, syntactically, and semantically. It is a Semitic language. The Semitic languages are notable for their non-concatenative morphology. Morphologically, the Arabic root is unique, consisting of isolated consonants rather than syllables or words. These roots are usually three in number, known as triliteral root or less common four, known as quadrilateral root. Long vowels are added to fill the gaps in the consonantal root to construct words, while short vowels appear as diacritic marks over and under the text. MSA is based phonologically, morphologically, and syntactically on classical Arabic, but it is a modern form of it. Furthermore, MSA is the written form rather than the spoken form.

On the other hand, Arabic dialects are the spoken forms of Arabic. These dialects are not taught or standardized. They are mainly used for informal social communication. However, the situation is changing because of social media. As more and more native Arabic speakers gain access to the electronic form of communication, Arabic dialects are used in written communication. Habash (2010, 2) lists the following seven dialects of Arabic:

1. “Egyptian Arabic (EGY) covers the dialects of the Nile valley: Egypt and Sudan.
2. Levantine (LEV) Arabic includes the dialects of Lebanon, Syria, Jordan and Palestine.
3. Gulf Arabic (GLF) includes the dialects of Kuwait, United Arab Emirates, Bahrain, and Qatar. Saudi Arabia is typically included although there is a wide range of sub-dialects within it. Omani Arabic is included some times.
4. North African (Maghrebi) Arabic (Mag) covers the dialects of Morocco, Algeria, Tunisia and Mauritania. Libyan Arabic is sometimes included.
5. Iraqi Arabic (IRQ) has elements of both Levantine and Gulf.
6. Yemenite Arabic (Yem) is often considered its own class.

7. Maltese Arabic is not always considered an Arabic dialect. It is the only Arabic variant that is considered a separate language and is written with the Roman script."

The relationship between MSA and the Arabic dialect is what is known as diglossia. Diglossia is the phenomenon when two languages or varieties of the same language are used in different situations within the same speech community. Arabic is known as a good example of diglossia. Thus, the relationship between MSA and Arabic dialect can be characterized by the following two features:

- MSA and Arabic dialects are different varieties of the same language which are very different from one another.
- MSA is not the native language of any native Arabic speaker. Instead, it is the standard form taught in schools and used in formal spoken and written situations.

1.3 Aims and Objectives

The main aim of this research work is to develop an automatic parts-of-speech tagger for tagging San'ani Arabic. The following objectives have been identified to address this primary aim:

1. Constructing a grammatically annotated social media-based corpus of San'ani Arabic of at least 200k tokens to be used for training. (see Chapter four and five)
2. Adapting a suitable Arabic tagset to perform the data annotation and parts-of-speech tagging. (see Chapter five)
3. Building and training a deep learning-based parts-of speech tagger using BI-GRUs-CRF model. (see Chapter six)

4. Evaluating the BI-GRUs-CRF parts-of speech tagger output using testing data of extra 11k tokens (see Chapter Seven)

1.4. Research Questions

The research questions are inspired by the gap in the surveyed literature. It all begins with the question

1. Does dialectical/San'ani Arabic NLP resources exist?

As seen in the review of literature, few NLP tools were developed for San'ani Arabic dialect. This leads to the following questions:

2. Are there any Arabic NLP resources or tools that can benefit dialectical/San'ani Arabic tagging? More specifically,
    a. Which tag set available for MSA parts-of-speech tagging can be adopted for San'ani Arabic parts-of-speech tagging?
    b. Is there a reasonable size corpus of San'ani Arabic text? And if so, is it enriched with parts-of-speech annotation?
3. Can BI-GRUs-CRF model be used to develop an efficient part-of speech tagger for San'ani Arabic?

1.5. Research Methodology

The primary aim of this thesis is to develop a parts-of-speech tagger for San'ani Arabic. In the context of parts of speech tagging, there are approaches and methods to parts-of speech tagging. It can be either supervised or unsupervised, while methods vary between rule-based, statistical-based, hybrid, and other deep learning methods.

For our project, we adopted an innovative method to perform San'ani parts-of-speech tagging, namely the Bidirectional-Gated Recurrent Units-Conditional Random Fields (BI-GRUs-CRF) Model. It is a combination of supervised deep learning and statistical-based methods. It is a data-driven (machine-learning) approach using a type of Recurrent neural networks (RNN) known as Bidirectional Gated Recurrent units (BI-GRUs) along with a supervised stochastic-based method known as Conditional Random Fields (CRF). This model is specialized in sequence labelling of longer sequences, which is the case of parts-of-speech tagging, benefitting from the past and future information. The process of developing the San'ani Arabic parts-of-speech tagger went through the following main stages.

a. Corpus developing

The availability of data is an essential requirement in building our tagger. However, as illustrated in the literature review, few San'ani Arabic dialect data resources are available. Hence, we had to develop our corpus of San'ani Arabic, utilizing open data sources. We selected and collected data from popular social media platforms in Yemen, namely, Facebook and Telegram. The corpus size surpasses 200k tokens for model training and 11k tokens for model evaluation.

After the collection of data, **data pre-processing** and **grammatical annotation** took place. Since our data is social media-based, further pre-processing is needed to remove noise and standardize ill-formed data. The data pre-processing includes three important techniques, which are **noise cleaning, data tokenization** (word and sentence tokenization), and **text normalization**. All the three pre-processing stages were applied to the data systematically. Then the pre-processed corpus was enriched with parts-of-speech tags. The data annotation process

abides by Leech (1993) maxims of corpus annotation. The annotation was conducted manually and by a native speaker of San'ani Arabic following the guidelines of the Penn Arabic Treebank (PATB) (2008).

b. Tagset Selection

We need a tagset that comprehensively covers the target variety to perform parts-of-speech tagging. So, a survey of the available Arabic parts of speech tagset led us to the fact that no standardized parts-of-speech tagset is available for Arabic; however, the Bies/LDC/RTS tagset was found suitable. Therefore, we adapted the Bies tagset.

c. The tagger Building and Training

Our tagger was built using the BI-GRUs-CRF model. The pipeline of our project consists of five steps: data loading, Word Embedding, model building, model training, and prediction. The **data loading** step ensures that the data is prepared and labelled adequately to be loaded as a machine-readable input. The next step, i.e., **word embedding**, converts the text into a numerical type to be fed into the machine-learning model.

In the **model building** step, the BI-GRUs-CRF model layers are built. The CRF layer is included at the end of the model to enhance the output sequences. Masking is defined to manipulate sentence length, and the machine is informed to ignore the padding. The number of the hidden layers of the BI-GRUs are defined and created.

The next step is the **model training**, where the BI-GRUs-CRF model was trained using our social media-based corpus of San'ani Arabic described in Chapter Four and the adapted tagset described in Chapter Five. We chose to train the model using the "**rmsprop**" optimizer

since it automatically updates the learning rate. Then the final and optimal weights matrix is saved and utilized to make predictions achieving the last step in the pipeline.

d. The tagger evaluation

The BI-GRUs-CRF tagger was evaluated using the accuracy measures. Two test sets were prepared and tested compared to the tagger output. The size of both test sets is over 11k. They were pre-processed and annotated using the same guidelines and tagset of the training data to ensure a valid evaluation.

1.6 Justification and Likely Benefits

The proposed research targeted dialectical Arabic parts-of-speech tagging due to the necessity of this task in many applications such as automatic translation, ontology engineering, question answering, word processing, and information retrieval. Moreover, many NLP tasks are dependent on parts-of-speech taggers efficiency. For example, parsing task is highly influenced by parts-of-speech tagging as parsers need to get the accurate parts-of-speech of each token in a targeted text. Thus, the more efficient the parts-of-speech tagger, the more efficient the parser will be. Similarly, word sense disambiguation and sentiment analysis could make use of accurate parts-of-speech taggers.

In addition, parts-of-speech taggers play a fundamental role in the creation of lexicographical resources such as, dictionaries and thesaurus. Other useful implementations of parts-of-speech tagging systems are in the automatic extraction of noun phrases, compounds and Multi word units. Text-to-speech systems can also benefit from automatic parts-of-speech taggers.

1.7 Thesis Outline

Chapter Two is an overview of the San'ani Arabic structure. It starts by giving an introduction to the Arabic language and its forms. The different forms of Arabic are presented, showing the social status and their relationship to one another. Then the San'ani Arabic structure is described under four headings: Orthography, phonology, morphology, and syntax.

Chapter Three presents a review of the literature on the parts-of-speech tagging. The first section reflects the methods of parts-of-speech tagging and provides a classification of the same. The second gives a chronological review of Parts-of-speech tagging in non- Arabic Languages. The fourth section surveys the history of the Arabic parts-of-speech tagging. The fifth and sixth sections investigate parts-of-speech tagset available for Arabic and a review of Dialectal Arabic Corpora, respectively.

Chapter Four describes the data collection and pre-processing. It is divided into four main sections. The first deals with data collection. The second presents the framework of data pre-processing in three stages: data cleaning, text normalization, and tokenization. The third section gives a statistical analysis of the developed corpus in all stages. The fourth section describes the corpus genre.

Chapter Five deals with the adopted tagset and the data annotation. In this chapter, the tagset is described along with justification for using the same. Then the grammatical annotation is introduced. Finally, the annotation statistics are presented.

Chapter Six describes the tagging model, i.e., the BI-GRUs-CRF Model for tagging. Moreover, the mathematical representation is given along with the used algorithm. In addition, the implementations, as well as, user interface are introduced.

Chapter Seven presents model evaluation and error analysis of the output. It explains the output accuracy rates and comments on the causes and types of the errors. It also suggests solutions to further improvement.

Chapter Eight gives the conclusion and summary of the work. It also suggests future expansions.

CHAPTER TWO
AN OVERVIEW OF SAN'ANI ARABIC

2.1 Overview of Arabic Language and its Forms

Arabic Language is registered as one of the six main languages globally. It contains three main forms and more than thirty dialects. This section aims at introducing Arabic Language and its forms with special emphasis on dialectal Arabic/San'ani Arabic. It consists of four sub-sections. The first sub-sections deal with Arabic Language forms typology. The second is allotted to Classical and Modern Standard Arabic, while the third presents dialectal Arabic under which San'ani Arabic is introduced.

2.1.1 Arabic Language forms

Arabic Language is one of the most spoken languages of the world. One of the markers of Arabic Language is the diglossic nature of the Language (Habash 2010), where two varieties (MSA and Dialectal Arabic (DA) exist side-by-side and are closely related. MSA is a predominant variety over dialectal Arabic informal settings, restricting almost all written content to the standard form. However, recently and with the advent of technology and the vast spread of social media networking sites, a strong presence of DA is noticed, and more individual-driven data becomes accessible and available as users of these sites feel free and encouraged to jot down their thoughts, interact or comment about their daily social life in their own dialects.

2.1.2 Classical Arabic (CA) and Modern Standard Arabic (MSA)

The start of CA is believed to be from the sixth century (Ryding, 2005). This era can be referred to as the pre-Islamic, and it was distinguished with a highly sophisticated poetic language. At that time, Arab tribes cared deeply about poetic language, as it was linked with tribal esteem. So, they used to have competitions in the art of public recitation and composition of poems and poetry. The ode */qasʕi:dah/* was characterized by being written in a standard poetic language. According to Arberry (1957), the scheme of the ode was highly conventional, where the length could reach 60 couplets, and each line ends with an identical rhyme. This form of highly poetic language no longer exists today (Ryding, 2005).

In the seventh century, the prophet Mohammed, peace and blessings of Allah be upon him, was powered with the holy book Qur'an for nearly 11 years (622-632 A) (Ryding, 2005). The advantage given to the Arabic language when selected as the language of the Holy Qur'an added holiness to the powerful poetic language. Since this time, with the spread of Islam, Arabic has become the center of centuries of religious study.

From the seventh through the twelfth century, Arabic was introduced as an international language due to the expansion of the Islamic empire. It was considered the international language of science, diplomacy, administration, and research. Later on, the Islamic empire suffered from several invasions by Crusades and Mongols and some regions' independence movements. Eventually, the Islamic empire got weak, which influenced the language.

From the thirteenth to the eighteenth century, the early Islamic CA was still the language of literacy, but different regions' spoken verities were used in everyday communication. So diglossia established itself with two forms of Arabic, i.e., CA as the written form and regional

dialects as the spoken ones, which were not written nor preserved. By the end of the eighteenth century, the Modern period took place. In this period, literacy spread as well as universal education concepts. Influenced by the western writing style and several types of literary forms, linguists started differentiating MSA from CA (Ryding, 2005). They distinguished MSA from CA in style and vocabulary. CA is style, and syntax is described as a complex, intricate form of formal language. At the same time, MSA is seen as more modern and suitable for journalistic broadcasting and other modes of formal education and media. Despite all the differences, CA and MSA are very similar, considering that MSA is originally derived from CA. This close similarity ensured the continuity of literacy and Islamic traditions. Thus, CA is categorized as a heritage form of the standard and MSA as the modern form.

2.1.3 Dialectal Arabic

Arabic dialects are classified into many broad categories based primarily on their regional locations. The broad regional dialects of Arabic are Egyptian Arabic (EGYA), Gulf Arabic (GFA), Levantine Arabic (LVA), Hassaniya Arabic (HSNA), Iraqi Arabic (IRQA), Sudanese Arabic (SDNA), Maghrebi Arabic (MGHBA), and Yemeni Arabic (YMNA). EGYA includes all the Arabic dialects spoken in Egypt. GFA consists of the Arabic dialects in KSA, UAE, Kuwait, Oman, Bahrain, and Qatar. LVA contains Arabic dialects spoken in Syria, Palestine, Lebanon, and Jordan. HSNA presents the dialects in Mauritania, Western Sahara, southwestern Algeria, and Southern Morocco. IRQA covers dialects spoken in eastern Syria and Iraq. SDNA contains dialects in Sudan and Southern Egypt. MGHBA includes dialects in Tunisia, Libya, Algeria, and Morocco. Finally, YMNA covers the dialects of Arabic spoken in Yemen and Southern KSA (Habash 2010; Alshutayri and Atwell 2018; Biadsy et al. 2009). Further division of the above categories is based on regional and social status.

2.2.2.1 San’ani Arabic

San’ani Arabic is one of the three main dialects spoken in Yemen (. It belongs to the Yemeni dialects spoken in the South of the Arabian Peninsula, namely, Yemen and south of the Kingdom of Saudi Arabia. It is mainly spoken by 30 percent of the whole population of Yemen, which would approximate 9 million speakers (Sharaf Addin and Al-shehabi 2020). San’ani is considered a spoken, informal variety, where MSA is the standard written form for all Arabic speakers. These two forms are used in complementary distribution, which is known as diglossia. Though San’ani Arabic has common linguistic features with Classical Arabic and MSA, it shows a linguistic peculiarity of its own. In the following section, we will show some of the disguising linguistic features of San’ani Arabic with occasional reference to MSA.

2.2.2.1.1 Who Speaks San’ani Arabic, and where it is spoken?

As shown in Figure 2.1 San’ani Arabic is mainly spoken in the North of Yemen. The approximate population that speaks San’ani is 9 million speakers who belong to nine different governorates, as shown in Table 2.1. It is considered the second most spoken dialect in Yemen after Ta'izzi-Adeni Spoken Arabic.

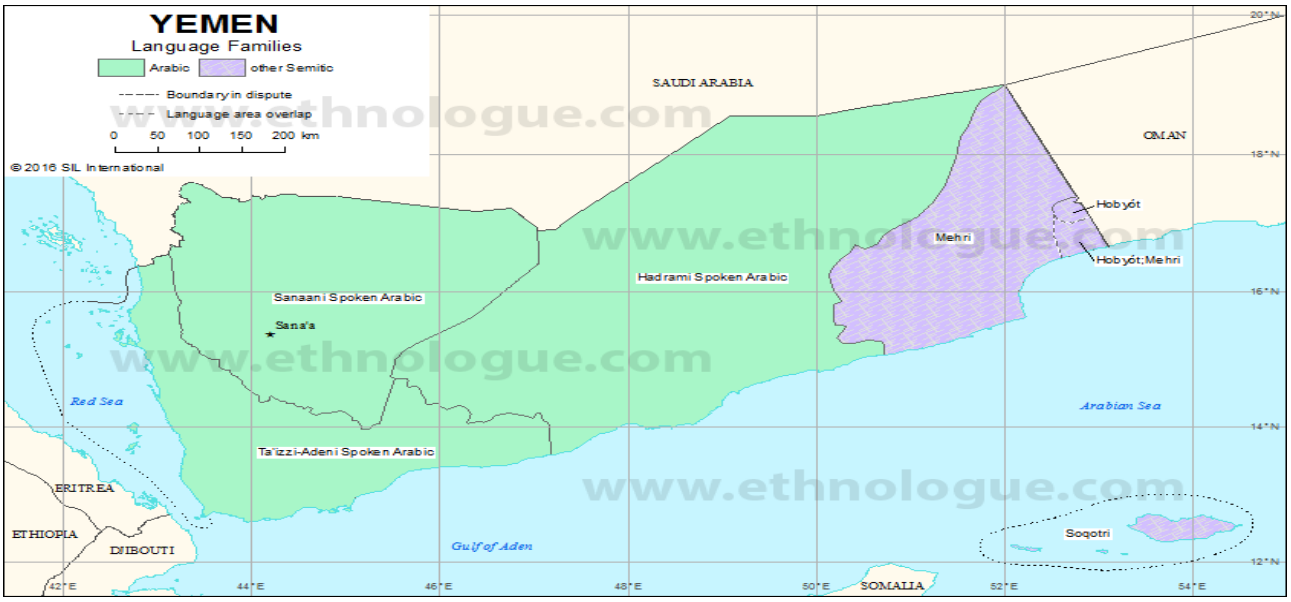


Figure 2.1 Yemen spoken languages and dialects

Source: Ethnologue: Languages of the World https://www.ethnologue.com/country/ye/maps

Table 2.1 ***Population of San'ani Arabic***

Division	Capital city	Population 2013 census
'Amran	'Amran	1,123,651
Al Jawf	Al Hazm	663,147
Al Mahwit	Al Mahwit	732,360
Amanat Al Asimah	Sana'a	1,174,767
Hajjah	Hajjah	1,887,213
Ma'rib	Ma'rib	504,696
Sa'dah	Sa'dah	987,663
Sana'a	Sana'a	2,279,665
Total		9,353,162

2.2 An Overview of the Structure of San'ani Arabic

2.3.1Orthography

San'ani Arabic does not have a standard orthographic system, but it uses Arabic script. This is because of the diglossic situation discussed earlier in section 2.2, where MSA is considered the formal written form and other dialectal varieties are considered as the informal spoken forms. Fortunately, with the emergence of social media platforms, personal blogs, etc., people started interacting and writing in their own spoken dialects. In the following, we will introduce the Arabic orthographic system.

The Arabic alphabet consists of twenty-eight letters where two of which are semi-vowels plus diacritic marks. These letters are constructed using only 18 shapes and dot/s to form them. For example, the shape ب produces three different letters with the use of dots: /ب/ "b"/ت/ "t" and /ث/ "ɵ." Moreover, in cursive writing, the shapes of Arabic letters change depending on the letter's position (i.e., separate or connected and if connected in which position beginning, middle, or end). Table 2.2 represents the Arabic alphabet and the letters' shape.

Table 2.2 *Arabic letters*

Name	IPA equivalent	Independent	w-initial	w-medial	w-final
hamza	ʔ	ء			
/ʔalif/	a:	ا	ا	ـا	ـا
/ba:ʔ/	b	ب	بـ	ـبـ	ـب
/ta:ʔ/	t	ت	تـ	ـتـ	ـت
/ɵa:ʔ/	ɵ	ث	ثـ	ـثـ	ـث
/ʤa:ʔ/	ʤ (ʒ in SA)	ج	جـ	ـجـ	ـج
/ħa:ʔ/	ħ	ح	حـ	ـحـ	ـح
/xa:ʔ/	x	خ	خـ	ـخـ	ـخ

/da:l/	d	د	د	ـد	ـد
/ða:l/	ð	ذ	ذ	ـذ	ـذ
/ra:ʔ/	r	ر	ر	ـر	ـر
/za:j or za:ʔ/	z	ز	ز	ـز	ـز
/si:n/	s	س	سـ	ـسـ	ـس
/ʃi:n/	ʃ	ش	شـ	ـشـ	ـش
/sˤa:d/	sˤ	ص	صـ	ـصـ	ـص
/dˤa:d/	dˤ	ض	ضـ	ـضـ	ـض
/tˤa:ʔ/	tˤ	ط	طـ	ـطـ	ـط
/ðˤa:ʔ/	ðˤ	ظ	ظـ	ـظـ	ـظ
/ʕajn/	ʕ	ع	عـ	ـعـ	ـع
/ɣajn/	ɣ	غ	غـ	ـغـ	ـغ
/fa:ʔ/	f	ف	فـ	ـفـ	ـف
/qa:f/	q	ق	قـ	ـقـ	ـق
/ka:f/	k	ك	كـ	ـكـ	ـك
/la:m/	l	ل	لـ	ـلـ	ـل
/mi:m/	m	م	مـ	ـمـ	ـم
/nu:n/	n	ن	نـ	ـنـ	ـن
/ha:ʔ/	h	ه	هـ	ـهـ	ـه
/wa:w/	w	و	و	ـو	ـو
/ja:ʔ/	j	ي	يـ	ـيـ	ـي

Unlike English, Arabic script goes from the right to the left. The diacritic markers go above or under the intended letters; however, nowadays, MSA writers in general and San'ani writers in specific ignore diacritics totally and do not use them whatsoever. The native speaker can guess these diacritics. It is important to state that there is no distinction between upper- or lower-case letters in Arabic, and writing is always cursive.

2.3.2 Phonology

A glimpse of a phonological language system is essential to draw a clear picture of its structure. The phonological establishments in a language can influence its orthography as well as morphology. This section describes the essential phonological components of San'ani Arabic, including Phonemic Inventory, Syllable Structure, and Phonological Alternations.

2.3.2.1 Phonemic Inventory

This section introduces the phonemic inventory of San'ani Arabic. It explains both the consonantal and vowel systems.

2.3.2.1.1 The Consonantal System

San'ani Arabic shares the phonemic inventory of MSA with certain changes. San'ani Arabic consonantal system consists of twenty-seven consonants. Out of these consonants, twenty-two are plain while the rest, i.e., five, are pharyngealized. The plain consonants are [b, m, f, θ, ð, t, d, s, z, n, r, l, ʒ, ʃ, j, k, g, x, ɣ, w, ʔ, h] and the pharyngealized are [ðˤ, tˤ, sˤ, ɦˤ] along with the voiced pharyngeal fricative [ʕ]. Pharyngealization is expressed by a superscript [ˤ]. Table 3 shows the consonantal inventory of San'ani and MSA guided by the place of articulation and manner of articulation. As shown in Table 2.3 San'ani Arabic distinguishes itself from MSA as follows:

- The voiceless uvular plosive /q/ is replaced with a voiced velar plosive /g/ as in /*ga:la*/ 'he said' (Qafisheh, 1990).

- The MSA pharyngealized voiced alveolar stop /dˤ/ is replaced with a voiced alveolar fricative/ðˤ/ in pronunciation, however, in the writing the orthographic shape [ض] is retained.
- The postalveolar affricate /ʤ/ is prounced as the English voiced postalveolar fricative /ʒ/, as in: /*ʤamal* / "camel".
- In foreign words that contains the voiceless bilabial stop /p/ and the voiced labiodental fricative /v/ phonemes are replaced by the voiced bilabial stop /b/ and the voiceless labiodental fricative /f/, respectively as /p/ and /v/ are not considered a part of the consonantal system.
- Gemination[1] of consonants is characterized by doubling the consonant or ignoring it; as in /madd jaduh/ "stretched (3S.M) his hand".
- Geminate consonants are represented as double letters, as in: /*xabba*:z/ 'baker', and /*sawwa*:*g*/ 'driver' (Watson, 1993).

[1] Gemination is also known as lengthening /doubling. In Arabic, it is known as */tashdi:d/* where consonants are doubled in both spelling and pronunciation. It means that the geminated consonant is articulated with double strengths. Gemination is signaled by the diacritic [ّ] which is known as */shaddah/*. It appears in Arabic script as a superscript. However, in Arabic dialects it is abandon and in writing, writers tend to either duplicate the consonant or leave it for the native reader to guess it out.

Table 2.3 *San'ani Arabic and MSA Consonant Inventories*

		Bilabial	Labiodental	Interdental	Alveolar	Postalveolar	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	Plain	b			t	d		k	g^ q+		ʔ
	Emph.				tˤ	dˤ+					
	Nasal	m				n					
	Trill					r					
Fricative	Plain		f	Θ ð	s	z	ʃ	x	ɣ	ħ	ʕ h
	Emph.			ðˤ	sˤ						
Affricate						ʤ					
Nn Glides (Approximant)							j		w		
Liquid (Lateral Approximant)						l					

(+)=found in MSA only

(^)= found in San'ani Arabic only

2.3.2.1.2 The Vowel System

San'ani Arabic has the same vowel inventory of MSA but it distinguishes itself with additional specifications. As shown in Table 4 the vowel system consists of six vowels divided into three pairs. Each pair contains a short vowel and its corresponding long vowel. The first pair i.e., /i, i:/consists of the short high front vowel /i/ and the long high front vowel /i:/. The second

consists of the short high back vowel /u/, and the long high back vowel, /u:/. The last pair contains the short low central vowel, /a/ and the long low central vowel, /a:/.

In addition, San'ani vowel system is distinguished by the following specifications:

- An additional vowel pair that contains the short mid front vowel /e/ and its corresponding long vowel /e:/.
- The use of the long mid back vowel /o:/is established only in loan words (Qafisheh 1990, 174) as in/*ʤa:lo:n*/ 'gallon.'
- /aj/ and /aw/ are the two diphthongs that are used in San'ani Arabic. According to Watson (1993), the /a/ is more open in /aw/ than in /aj/. as in the interrogative particle /*ʔajn*/ "where" and in the coordinating conjunction /*ʔaw*/ "or". Table 2.4 shows the vowel inventory of San'ani Arabic.

Table 2.4 The vowel inventory of San'ani Arabic

	Short			Long		
	Front	Central	Back	Front	Central	Back
High	i		u	i:		u:
Mid	e			e:		o:
Low		ɑ			a:	
Diphthongs			aj, aw			

2.3.2.2 Syllable Structure

San'ani Arabic has three syllable structures which are of the types: monosyllabic, disyllabic and trisyllabic. Watson (2008) classified them as light (one syllable), heavy (two syllables), and super heavy (three syllables) as shown in Table 2.5 The first two i.e.,

monosyllabic and disyllabic occur at all positions while trisyllabic i.e., super heavy, is restricted in position as the following rules:

- CVCC and CV:C occur stem final position
- CVCCC and CV:CC occur at word final position
- Ca:C occurs stem final only when h- or n- are add as suffixes initially.

Table 2.5 *Syllable Inventory in San'ani Arabic*

Light syllables	**Heavy syllables**	**Super heavy syllables**
CV	CVC	CVCC
	CV:	CV:C CVCCC/CV:CC

***Source**:* This syllable inventory is cited from Watson (2008, 2)

2.3.2.3 Phonological processes

According to Watson (1993) phonological processes in San'ani Arabic can be classified into two categories: processes represented in transcription and processes not represented in transcription. The first i.e., the phonological alternations in transcription are those which take place across morphemes within the syntactic word. The second type, on the other hand, refers to the processes which occur at morphemes and word boundaries. Each type of these can further be divided into number of sub-types as shown in Figure 2.2.

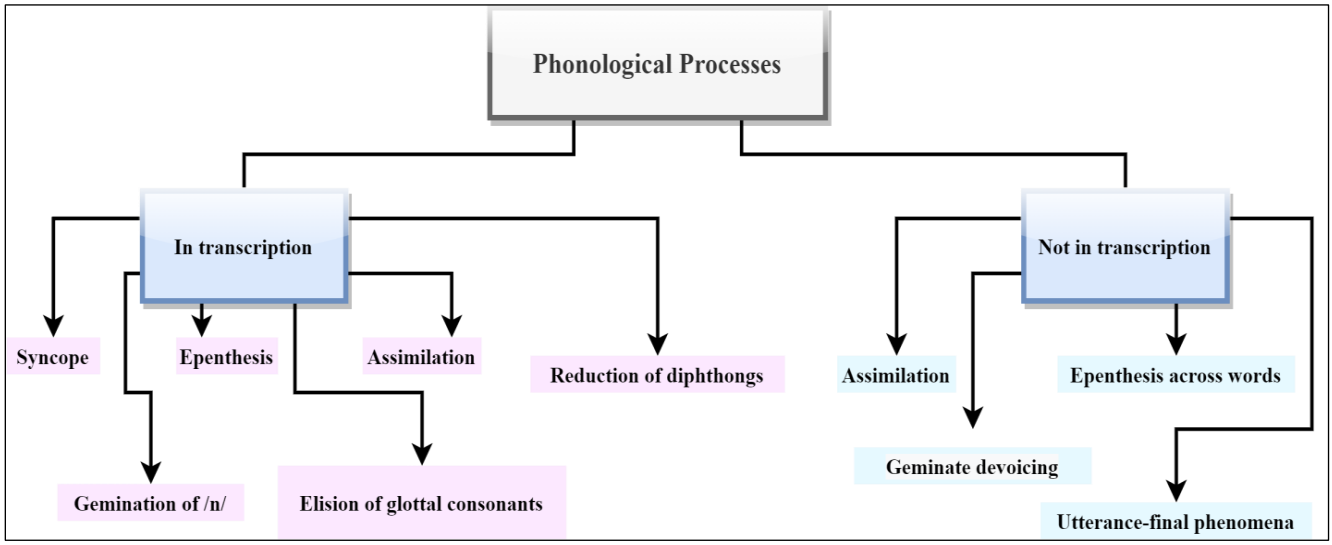


Figure 2.2 Phonological processes in San'ani Arabic

2.3.3 Morphology

San’ani Arabic is morphologically rich. It shares with MSA a complicated morphology however, it diverges from MSA morphology showing distinguishing characteristics. This section handles essential morphological aspects of San’ani Arabic. These aspects are inflectional morphology, derivational morphology and Grammatical Parts of Speech.

2.3.3.1 Inflectional Morphology

Inflectional morphology is defined by several linguists for instance, Aronoff gives the following definition to Inflectional classes “a set of lexemes whose members each select the same set of inflectional realizations” (Aronoff, 1994, 65). Carstairs-McCarthy also describes it as “a set of words (lexemes) displaying the same paradigm in a given language” (Carstairs-McCarthy, 1994, p. 739). In other words, inflectional morphology can be described as the study

of the morphosyntactic features that distinguish lexical forms of the same grammatical category such as number, gender, person and so on. Usually, only open classes are inflected with such categories.

Since Arabic language is highly inflectional, San'ani Arabic is not an exception. Mainly there are eight inflectional grammatical features which are aspect, mood, voice, person, number, gender, case and definiteness. Verbs inflect for aspect, mood, voice and person, gender and number while other open classes inflect for gender, number, state and case. In the following each of these inflectional features is discussed.

1. **Aspect**

Aspect loosely refers to time marking of verbs. Traditional Arabic grammar distinguishes two aspects: perfect (past) and imperfect (non-past). In San'ani , verbs inflect for either one of the two aspects however perfect is considered as the more basic aspect (unmarked). The perfect verbs are listed in lexicons as the basic lexical entries (Watson, 1993).

As shown in Table 2.6, the perfect markers in San'ani Arabic are suffixes to be attached to the end of verbs, while imperfect markers are confixes, i.e., combinations of both prefixes and suffixes to be attached to the beginning and end of verbs. Moreover, in Arabic aspects inflection involves the attachment of subject agreement markers, i.e., person, number and gender to the verb stem which is the case of San'ani Arabic. The following examples show both perfect and imperfect aspect along with agreement markers:

/daxal-at ʕasˤtˤ-at al-ʕasˤi: tˤ wa- grab-at-ih/ "entered (3 F perfective) made (3 F perfective) the porridge and served (3 F perfective) it"

/tu-dxal tu-ʕasˁtˁ al-ʕasˁi: tˁ wa- ti-grab –ih/"enters (3 F imperfect) makes (3 F imperfect) the porridge and serves (3 F imperfect) it"

Table 2.6 *Perfect markers in San'ani Arabic*

Perfect aspect			
Person		**Singular**	**Plural**
1		*-t*	*-na*
2	M	*-t*	*-tu*
	F	*-ti*	*-tajn*
3	M	*-*	*-u*
	F	*-at* *-it* (weak verbs)	*-ajn*

Imperfect aspect (non-past) includes both Continuous/habitual and future time. Imperfect markers are listed in Table 2.7.

Table 2.7 ***Imperfect markers in San'ani Arabic***

Imperfect aspect					
Person		**Continuous/habitual**		**Future**	
		Singular	**Plural**	**Singular**	**Plural**
1		*a-* *bajta-* *bajna-*	*na-* *bi-na-*	*ʃa-* *ʕad-*	*ʕa-na-*
2	M	*ti-*	*ti- -u*	*ʕa-*	*ʕa- -u*
	F	*ti- -i*	*ti- -ajn*	*ʕa- -i*	*ʕa- -ajn*
3	M	*yi-*	*ji- -u*	*ʕa-*	*ʕa- -u*
	F	*ti-*	*ji- -ajn*	*ʕa-*	*ʕa- -jan*

It is worth mentioning; that aspect might influence the morphological pattern of the verbal stem (*wazn*) causing certain vowel changes. For instance, the triliteral verb of the stem pattern CVCVC in perfect changes to CCVC in imperfect, as in:

/liʕib-t / “I played” (1SG) perfect aspect

/ʔa-lʕab / “I play” (1SG) imperfect aspect

2. Mood

Along with aspects Arabic verbs are inflected for mood. In MSA, perfect verbs are inflected for indicative mood and imperfect verbs are inflected for one of three moods; indicative, subjunctive and jussive in addition to the imperative. In San’ani Arabic, on the other hand, perfect and imperfect verbs inflect only for indicative mood besides imperative. Table 2.8 presents the indicative perfect markers used with the verb /*samaʕ/* “heard” (3 SG M)

Table 2.8 ***Indicative Perfect markers in San’ani Arabic***

Indicative Perfect markers			
person		singular	plural
1		*samaʕ-t*	*samaʕ-na*
2	M	*samaʕ-t*	*samaʕ-tu*
	F	*samaʕ-ti*	*samaʕ-tajn*
3	M	*samaʕ*	*samaʕ-u*
	F	*samaʕ-at*	*samaʕ-ajn*

The imperfect aspect,as mentioned earlier, expresses the non-past i.e., present (continuous/habitual) and future. The indicative mood conjugates the imperfect form of the verb as shown in table 2.9 using the verb pattern CCVC */ji-**smaʕ** /* "hears" (3 SG M)

Table 2.9 *Indicative Imperfect markers in San'ani Arabic*

Indicative imperfect markers					
Person		**Continuous/habitual**		**Future**	
		Singular	**Plural**	**Singular**	**Plural**
1		*a-smaʕ*	*na-smaʕ*	*ʃa-smaʕ*	*ʕa-na-smaʕ*
		bajta- smaʕ	*bina-smaʕ*	*ʕad-smaʕ*	
		bajna- smaʕ			
2	M	*ti- smaʕ*	*ti-smaʕ -u*	*ʕati-smaʕ*	*ʕati-smaʕ-u*
	F	*ti-smaʕ-i*	*ti-smaʕ-ajn*	*ʕati-smaʕ-i*	*ʕati-smaʕ-ajn*
3	M	*ji-smaʕ*	*ji-smaʕ-u*	*ʕaji-smaʕ*	*ʕaji-smaʕ-u*
	F	*ti-smaʕ*	*ji-smaʕ-ajn*	*ʕati-smaʕ*	*ʕati-smaʕ-jan*

Imperative mood is derived from the imperfect verb form without the person agreement prefix. Instead, imperative mood has a functional second person antecedent. To form verbs in imperative mood, the imperfect verb stem is used as the stem of the imperative but the prefix of the imperfect is replaced with a glottal stop /ʔ/ followed by one of these high vowel */i/*or */u/*. The choice of the vowel depends on the environment. The following re-write rule explains the vowel choice:

(1) Rule of vowel choice in Imperative verb formation u/→— CCu..
i/ elsewhere

Since imperative mood manifests itself functionally in 2-person recipient, the gender and number agreement markers are attached as suffixes as shown in Table 2.10:

Table 2.10 ***Imperative markers in San'ani Arabic***

Imperative markers					
Person/		**Singular**		**Plural**	
2	M	*ʔi-/ʔu-*		*ʔi/ʔu*	*-u*
	F	*ʔi-/ʔu-*	*-i*	*ʔi/ʔu*	*-ajn*

It is fair to say that the imperative prefix is always /i/ unless the verb stem contains u, then the prefix vowel has to be /u/ to create vowel harmony. The following examples represent the imperative formation rule along with the agreement markers addition.

Imperfect	**Gloss**	**Imperative**	**Gloss**
ji-ktub	"writes (3 SG M)"	*ʔu-ktub*	"write (2 SG M)"
ji-smaʕ-ajn	"hear (3 PL F)"	*ʔi-smaʕ-ajn*	"hear (2 PL F)"

In addition in weak (defective and hollow) verbs roots, i.e., roots which contain ya: /j/ ي, wa:w /w/ و or hamzah /ʔ/ ء in its structure , their aspect, mood and subject conjugation involves the addition of various irregularity to the stem where additional vowel changes occur. The reason is phonological as such verbs are influenced by their surroundings. For instance, the verb */wagaf/*"stand up" (3 M SG) is derived from the triliteral root */wgf/* which is a first weak verb (initial position).Table 2.11 shows the following conjugation in San'ani :

Table 2.11 ***The weak verb conjugation (initial position)***

Person		Perfect		Imperfect					
				Continuous/habitual		Future		Imperative	
		Singular	**Plural**	**Singular**	**Plural**	**Singular**	**Plural**	**Singular**	**Plural**
1		*wigaf-t*	*wigaf-na*	*a-wgaf* *bajta-wgaf* *bajna-wgaf*	*na-wgaf* *bina-wgaf*	*ʃa-wgaf* *ʕad- awgaf*	*ʕa-nu- wgaf*		
2	M	*wigaf-t*	*wigaf-tu*	*tu-wgaf*	*tu-wgaf -u*	*ʕatu-wgaf*	*ʕatu-wgaf-u*	*ʔu-wgaf*	*ʔu-wgaf-u*
	F	*wigaf-ti*	*wigaf-tajn*	*tu-wgaf-i*	*tu-wgaf-ajn*	*ʕatu-wgaf -i*	*ʕatu-wgaf-ajn*	*ʔu-wgaf-i*	*ʔu-wgaf-ajn*
3	M	*wigaf*	*wigaf-u*	*ju-wgaf*	*ju-wgaf-u*	*ʕaju- wgaf*	*ʕaju-wgaf-u*		
	F	*wigif-at*	*wigif-ajn*	*tu-wgaf*	*ju-wgaf-ajn*	*ʕatu- wgaf*	*ʕaju-wgaf-jan*		

3. Voice

San'ani verbs inflect for voice which is of two-way distinction; active and passive. The active voice is considered the unmarked and is represented by the usual verb forms. The passive voice, on the other hand, is classified into three types based on the forming pattern. These passive types are apophonic passive, medio-passive and derived medio-passive.

The apophonic passive is formed by certain internal vowel changes in transitive verbs. It is the standard type which distinguishing MSA as well as Yemeni Arabic (Watson, 1993). The San'ani apophonic passive has a set of vowel patterns for passivation of active verb: [u-i, u-a].

the choice of these pattern is decided by the aspectual status of the verb i.e., perfect or imperfect. Usually, perfect verbs use the first pattern [u-i,], and imperfect verbs use the second [u-a]. Check the following examples:

Active	**Gloss**	**Passive**	**Gloss**
xalag	"created (3 SG M)"	*xulig*	"he was born"
kasar	"broke"	*kusir*	"was broken"
jaʕraf	"knows" (3 SG M)	*juʕraf*	"is known"
tixbiz	"bakes" (3 SG F)	*t/juxbaz*	"is baked"

The second type is the medio-passive which refers to the passive where actor is not implied or cannot be figured out. Moreover, the passive that does not have active equivalent is also a medio-passive. Check the following examples:

/tˁiwil/ "It became tall (M)"

/ximid/ "it got cooked (M)"

The derived medio-passive is the one which is formed by the addition of affixes to the active form. It is more common in dialectal varieties than apophonic passive (Al-Toma 1969). In San'ani the morpheme [-t] or [-n] are used as prefixes to the perfect form and infixes in the imperfect verb to form the derived medio-passive. An additional glottal stop [*ʔ*] or a vowel is attached initially to the perfect form only. The following examples show the derived medio-passive.

Active	**Gloss**	**Passive**	**Gloss**
/xalag/	"created (3 SG M)"	*/ʔi-nxalag /*	"he was born"
/kasar/	"broke"	*/ti-kasar/ʔin-kasar/*	"was broken"
/jaʕraf/	"knows (3SG M)"	*/ji-n-ʕrif/*	"is known"
/yu-xtˤub/	"engages (3 SG M)"	*/ti-n-xatˤib/*	"she gets engaged"

4. Person

San'ani verbs and personal pronouns inflect for person. There are three-person distinction; first, second and third person. 1 person expresses the speaker (*ʔana:* "I" and *ʔiħna /ʔaħna* "we"). It has number distinction of singular /plural but no gender. The second person refers to the addressee (*ʔant* "you (SG M)," *ʔanti:/ʔinti:* "you (SG F)," *ʔantu:/ʔintu:* "you (PL M)," *ʔantajn "*you PL F"). The third person refers to the absent, i.e., neither the speaker nor the addressee (*hu:* "he", *hi:* "she" *hum* "they (PL M)" *hin* "they (PL F)." The second and third persons have number and gender distinction as shown above. When verb stems show inflection of any inflectional category, there need to be covert or overt subject or object agreement markers. (cf. Table 2.6 and 2.7).

5. Gender

In Arabic as well as San'ani gender is classified as masculine or feminine. Moreover, gender is morphological rather than natural where the category to which gender refers is semantically arbitrary unless it refers to a real being (Ryding, 2005). Nouns, verbs, adjectives

and pronouns inflect for gender in which it is unmarked on nouns. Generally, gender is visible where gender affixes are added to the inflected word however sometimes gender is invisible and can only be reflected in agreement.

6. Number

Though MSA shows three-way distinction of number, i.e., singular, dual and plural, San'ani Arabic abandoned the use of dual number except in narrow cases of noun inflection[2]. In San'ani nouns, verbs, adjectives and pronouns inflect for number. Verbs show number inflections which is associated with other subject agreement markers when they inflect for aspect, mood or voice. (c.f. Table 2.6 and 2.7).

7. Definiteness

Definiteness is an inflectional category that applies to both nouns and adjectives. In San'ani, the prefix /-al/ is added to the beginning of nouns or adjectives to denote definite state. Similarly, the absence of the prefix /-al/ denotes the indefinite; as in:

Definite noun	**Indefinite noun**
/al-bnit/ "the girl"	*/ bnit /*"a girl"
/al-kari:m/"the generous (3 SG M)"	*/kari:m /*"generous (3 SG M)"
/al-ʤuba:/"the roof"	*/ʤuba:/*"a roof"
/al-ri:ʤa:l/"the men"	*/ ri:ʤa:l /*"men"
/al-ħumijʃ/ "the hard working (3 SG)"	*/ ħumijʃ/*"hard working (3 SG)"

[2] In nouns, the dual is used to indicate the number of an object is two. Though there is no contrast in numbers between singular, dual, and plural in SA, the dual is usually showed only for weight, measurement, or time (Watson, 2008). The suffix /-ajn/ is the dual suffix which forms the dual by attaching to the singular nouns of weight, measurement, or time, as in, /*ʃahr*/ 'a month' and /*ʃahrajn*/ 'two months, /*sa:ʕah*/ 'an hour' and /*sa:ʕatajn*/ 'two hours', /*jawm*/ 'a day' and /*jawmajn*/ 'two days', /gɪrʃ/ 'a riyal' and /*gɪrʃajn*/ 'two riyals'. In some other cases, the words /*ʔɪənajn*/ 'two masc.' and /*əɪntajn*/ are attached before the plural form to indicate the dual, as in /*ʔɪənajn rɪʤa:l*/ 'two men' and */əɪntajn bana:t*/ 'two girls/daughters.'

Proper nouns are considered definite. So definite prefix/-al/ is usually added to common nouns. Ryding (2005) listed two more way of expressing definiteness in Arabic:

- using the annexation
- suffixing a possessive pronoun

Annexation[3]

Arabic has a syntactic structure that is known as */ʔidˤa:fa/* "addition" annexation or genitive construct state. It consists of /mu*dˤa:f/* which is the first term that is indefinite and */mudˤa:f ʔili:h/*that is the second definite term. The first term */mudˤa:f/* is made definite by the addition to the second definite tern which can be a proper noun. For example, the word */bint/* "girl" is indefinite and the word */ʔlħa:rah/* "the neighbourhood" is definite. The term */bint ʔlħa:rah /* " the girl from the neighbourhood" is made definite by adding both term together.

Suffixing a Possessive Pronoun

Here indefinite nouns can be made definite by adding a suffixes Possessive pronoun that is similar to English possessive pronouns, however in Arabic possessive pronouns are bound morphemes; As in:

/kita:bu-hum/ " their books (M)"	*/kitab/* "a book"
/sija:rat-hin/ " their car (F)"	*/sija:ra/* "a car"

[3] Annexation is a construction that is unique to Semiotic languages, though some scholars might consider it a type of compounding, but it differs from compounding, as it functions at the level of syntax rather than morphology. for more information refer to (Ryding, 2005 Chapter 7)

2.3.3.2 Derivational Morphology

Derivational morphology refers to the study of formation of new words which differ from the original in syntactic class or semantic meaning. In other words, it analyses the processes by which new words are formed in a language such as affixation, compounding and so on. In San'ani Arabic, new words are mainly produced by derivation or what is known in Arabic grammar as */ʔiftiqa:q/*. Derivation is applied using root and pattern interleaving to produced new words. Arabic root-and- pattern system is briefly discussed in the next section.

2.3.3.2.1 Root-and-Pattern System

It is a system that consists of a root of consonants or consonant radicals and a pattern of vowels and sometimes other consonant that interconnect together to form a word (Ryding 2005). Both the root and the pattern cannot stand alone so they are bounded to each other to function. For example, the word */maddrasih/* "school" came from the root /d-r-s/ and the pattern /maCCaC-ih/ In this way by merging the root with different patterns a hug number of words are created in Arabic and its forms. A few more examples of the words that can be created from the root **/d-r-s/** are shown below:

Word pattern	**Actual word**	**Gloss**
CaCaC	*daras*	"studied (3 SG M)"
Ca:CaC	*da:ras*	"studied with (3 SG M)"
CuCiCa	*duris-a*	"was studied (PASS)"
CiCa:C-ih	*dira:s-ih*	"studying"
CuCu:wC	*Duru:ws*	"lessons"

Ca:CiC	*da:ris*	“learner”
uCCuC	*udrus*	“study (2 M SG imperative)”
maCCaC-ih	*madras-ih*	“school”
maCa:CiC	*mada:ris*	“schools”
naCCuCu	*na-drus-u*	“study (1 PL)”
maCCu:C	*madru:s*	“studied”
CuCCajjiC	*mudrajis*	“teacher”

As shown in the above example how the root /d-r-s/ is merged with the pattern in the first column to produces different words and word forms. After derivation the inflectional categories can be added as affixes; as in /*madras-ih/* “school”*/mada:ris/ “schools”*.

Similarly, the same pattern can be integrated with different roots to create words of the similar syntactic classes and semantic field. For instance the pattern: CiCa:Cih

Root	**Actual word**	**Gloss**
d-r-s	*dira:sih*	studying
k-t-b	*kita:bih*	writing
n-ʤ-r	*niʤa:rih*	carpentry
t-ʤ-r	*tiʤa:rih*	trading
f-l-ɦ	*fila:ɦih*	farming
d-b-x	*diba:xih*	cooking
x-b-z	*xiba:zih*	baking
x-j-tˁ	*xija:tˁih*	sewing
n-ɦ-t	*niɦa:tih*	sculpting

ʤ-z-r *ʤiza:rih* butching

As seen in the above examples, when the pattern CiCa:Cih interacts with multiple roots, gerund nouns are derived. It is important to state, that both roots and patterns are abstract representations where neither of them can function independently (Ryding 2005).

2.3.3.2.2 Other Word Formation Processes

San'ani Arabic, similar to MSA, employs other word formation techniques beside derivation which form the fast majority of vocabulary. Some of these techniques are:

- **Coinage*/naɦt/***

It refers to the process where two words are merged into one word. According to Ryding (2005) this process of compounding which is not common in Classical Arabic; however, it is used in MSA to coin new terms and for the need of translation. Words produced from coinage can consist of the whole composing words such as: */raʔsma:l/* "capital" which is built from */raʔs* / "head" and */ma:l/* "money". Other words are composed from part of the first word plus the second word such as */kahrumaɣna:tˁjisi/*"electromagnetic" which consists of */kahru/* a part of */kahraba:ʔi/* "electrical" plus*/maɣna:tˁjisi/*"magnetic".

- **Compounding */ *tarki:b/***

Owens (1988) defines compounding as "the joining of two words together to form a unit that is like a single word" (Owens 1988, 123)". In Arabic it is known as */tarki:b/* which relates to the a processes of making up a single word or a noun phrase or by joining two or more words together. The justification behind the formation of words using this process is related to the need of rapid translation of technical and scientific terms from other languages into Arabic (Ryding

2005). Such a: */umm ʔarbaʕah wa-ʔarbaʕjin/* "centipede", which is made from the combination of */umm/* "mother" plus */ʔarbaʕah wa-ʔarbaʕjin/* "forty-four". Another example that is used in San'ani is */bint bunuwt/* "virgin".

- **Arabicization */at-taʕri:b/***

Arabicization is called */at-taʕri:b/*. It is defined by The American Heritage Dictionary of English Language (2016) as "To make Arabic in form, style, or character." The process of Arabicisation can be described as the processes of reshaping foreign words to match Arabic structure and requirement. In Arabic and San'ani , borrowing, figurative translation and blending are considered as means of Arabicization (Ghanim, 2014). Some examples of Arabization in San'ani Arabic are:

/talafizju:n/	"television"
/radjuw/	"radio"
/mikja:ʤ/	"make-up"

2.3.3.3. Parts of Speech in San'ani Arabic

Traditional Arabic grammar divides Parts of Speech into three main classes: Nouns, Verbs and Particles. Furthermore, traditional grammarians include adjectives, pronouns, numerals and adverbs in the noun class. Besides, verb class is sub-divided into imperfective and perfective. Moreover, particles class contains prepositions, conjunctions and interjections. In San'ani Arabic this three-way distinction is approved by linguists as general categories that

branch further into sub categories. For example, Watson (1993) adopted the three-way distinction in analysis of San'ani providing a hierarchical classification.

In Watson's classification (1993), she further divides nouns into: substantives, adjectives and verbal derivatives and pronouns and circumstants. Substantives are divided into common nouns, which sub-classify into: concrete nouns and abstract nouns, and proper nouns. Adjectives and verbal derivatives are classified into Adjectives and verbal derivative. Adjectives are divided into basic adjectives and elative. Pronouns and circumstants are further divided into pronouns that subdivide into Personal, demonstrative and locative demonstrative pronouns. The second main class is verbs which are divided into core and deficient verbs. The last main class is particles which are divided into the following types:

- conjunctions
- adjunctions
- conditional particles
- negative particles
- prepositions
- verbal particles
- adverbial particles
- vocative particles ya:
- determiners
- hypotactic particle ʔinn
- nominaliser ma:

The classification is depicted clearly in Figure 2.4.

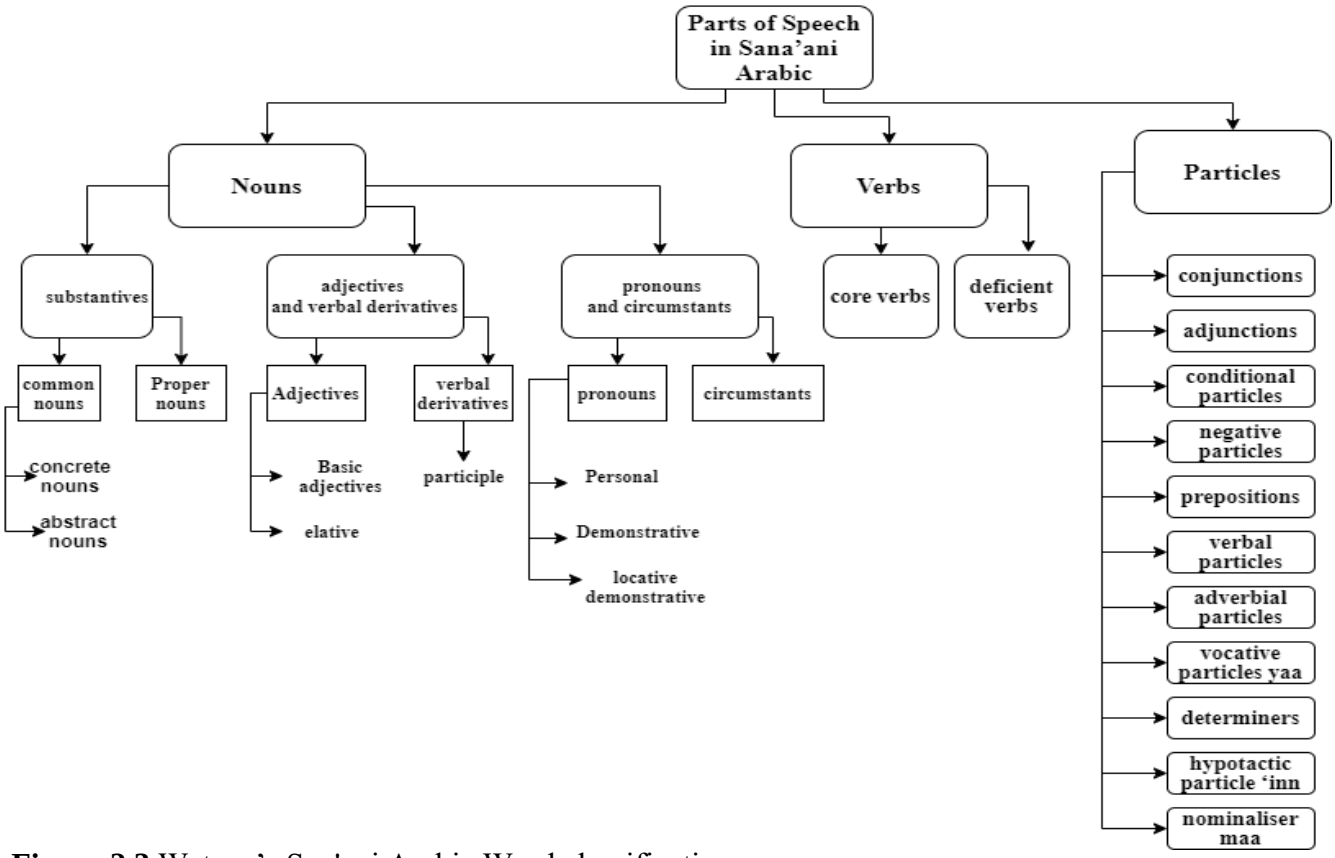


Figure 2.3 Watson's San'ani Arabic Word classification

In Modern Linguistics, the traditional Arabic grammar's three way classification was not good enough. A clear-cut distinction in parts of speech proves to be essential for different linguistic analyses and applications. Thus, linguists deviated from the traditional way, offering their own word classification. Such as Aboul-Fetouh (1969: 35) who distinguishes six parts of speech in Arabic: nouns, pronouns, verbs, adjectives, adverbs and particles.

In the field of computational linguistics, words' classification plays a fundamental role in various NLP tasks. For instance, Arabic computational linguists rely of word classification in designing tagsets to annotate date with Parts of Speech. So, a well-defined distinction of word

categories is required. So, our classification of San'ani Parts of Speech is inspired by (Watson, 1993) classification but with a more linear division that provides a sharp cut of Parts of Speech.

1. The noun

Nouns also termed substantives, are the core nouns in the language are classified as common nouns */ʔismu-lʤins/* and proper nouns */ʔismu-lʕalam*/. The common noun category is subdivided into concrete nouns */ʔism ʕajn*/ and abstract nouns */ʔism maʕna:/*. The concrete sub-type is either countable or uncountable. Table 2.12 shows core nouns typology in San'ani Arabic with examples.

Table 2.12 *Types of core nouns in San'ani Arabic*

	Common nouns			**Proper nouns**
Sub-Type	**Abstract nouns**	**Concrete nouns**		
		Countable	**Uncountable**	
Example	*/ʕilm/*	/*galam*/	*/ħali:b*/	*/rayad , ʤana:/*
Gloss	"knowledge"	"pen"	"milk"	"Raghad, Jana" (female names)

2. Pronouns

Pronouns belong to a closed class category that is distinguished from nouns. The most prominent feature of closed classes, including pronouns, is that they do not inflect for definiteness, which distinct them from core nouns. In fact, pronouns do not utilize inflection to show their morphological properties as they possess their inherent features, which are as follows.

- As opposed to other parts of speech, pronouns have a three-way person distinction: first, second, and third.

- They also have a two-way number distinction singular and plural. There is no gender distinction in first-person pronouns, but second and third-person pronouns show feminine and masculine distinction.

Additionally, pronouns in San'ani are divided into personal and demonstrative. Personal pronouns are sub-divided into subject, object, and possessive pronouns. Demonstrative pronouns are proximal, locative, and indefinite. The different types of pronouns are presented below.

Personal Pronouns

Generally, Arabic personal pronouns are subject, object, and possessive pronouns, and San'ani Arabic is no exception. Subject pronouns are free words that can stand alone independently however object, and possessive pronouns are bound, in the sense that they do not stand alone; they have to be attached to another word.

Subject Pronouns

As their name suggests, subject pronouns occur as the subject of sentences. Table 2.13 lists subject pronouns in San'ani Arabic, illustrating person, number, and gender distinction.

Table 2.13 *Subject pronouns in San'ani Arabic*

Subject Pronoun	Gloss	Person	Number	Gender
/ʔana:/	'I'	1	SG	-
/ʔiħna:/	'we'	1	PL	-
/ʔant/	'you'	2	SG	F
/ʔantu:, ʔintu:/	'you'	2	PL	M
/ʔanti:, ʔinti/	'you'	2	SG	F
/ ʔantajn/	'you'	2	PL	F
/hu:/	'he'	3	SG	M

/hum/	'they'	3	PL	M
/hi:/	'she'	3	SG	F
/hin/	'they'	3	PL	F

Object Pronouns

Object pronouns are objects in the form of bound morphemes attached to transitive verbs or prepositions as their objects. for example, */ʔana: katab-tu-hum/* "I wrote them" */-hum/* is an object pronoun attached to the end of the verb base */katab-tu/*. In */hum ʕamalu la-na: ħaflah/* "they made for us a party" */-na:/* is an object pronoun attached to the preposition */la-/*. Table 2.14 lists objects pronouns in San'ani Arabic along with their distinction.

Table 2.14 *Object pronouns in San'ani Arabic*

Object Pronoun	Gloss	Person	Number	Gender
/-ni:/	"me"	1	SG	-
/-na:/	"we"	1	PL	-
/-ak/	"you"	2	SG	M
/-kum/	"you"	2	PL	M
/-iʃ/	"you"	2	SG	F
/-kin/	"you"	2	PL	F
/-ih/	"him"	3	SG	M
/-hum/	"them"	3	PL	M
/-ha:/	"her"	3	SG	F
/-hin/	"them"	3	PL	F

Possessive pronouns

Possessive pronouns are bound morphemes attached to nouns and show agreement in gender and number. They express possession, and the host nouns are considered definite through possession. In form, possessive pronouns and object pronouns are almost the same; however,

they function differently. Moreover, object pronouns are only attached to verbs or prepositions, while possessive are attached to nouns. For instance, in /*qalam-iʃ*/ "her pen," /-iʃ/ is a third-person singular possessive pronoun but in /ʔana: *kalamt-iʃ*/ "I told You (3 SG F)" and /*qarʔat la-iʃ*/ "I read for her" is an object pronoun. Table 2.15 presents possessive pronouns in San'ani Arabic.

Table 2.15 *Personal possessive pronouns in San'ani Arabic*

Possessive Pronoun	Gloss	Person	Number	Gender
/-i:/	"my"	1	SG	-
/-na:/	"our"	1	PL	-
/-ak/	"your"	2	SG	M
/-kum/	"your"	2	PL	M
/-iʃ/	"your"	2	SG	F
/-kin/	"your"	2	PL	F
/-ih/	"his"	3	SG	M
/-hum/	"their"	3	PL	M
/-ha:/	"her"	3	SG	F
/-hin/	"their"	3	PL	F

Demonstrative pronouns

Demonstrative pronouns */ʔasma:ʔ al-ʔiʃa:ra(t)/* are free determiners that modify nouns or occur individually to express distance (Ryding 2005). In San'ani Arabic, they are classified into three types demonstrative, locative demonstratives, and indefinite demonstrative pronouns. Demonstrative pronouns are of two sub-types the first is formed with the prefix */ha:/*, while the second without it. Morphologically, they show number and gender distinction. The singular, near demonstrative pronoun, is considered the unmarked variant. Table 2.16 and 2.17 show the two form of San'ani Arabic demonstrative pronouns and their distinction.

Table 2.16 *Demonstrative pronouns with /-ha:/*

Demonstrative Pronouns with *-ha:/*	**Gloss**	**Number**	**Gender**	**Distance**
/ha:ða:/	“this”	SG	M	Near
/ha:ði/	“this”	SG	F	Near
/ha:ðawla:, hawla:/	“these”	PL	M or F	Near
/ha:ða:-k/	“that”	SG	M	Far
/ha:ði-k/	“that”	SG	F	Far
ha:ðawla:-k/	“those”	PL	M or F	Far

Table 2.17 *Demonstrative pronouns without /-ha:/*

Demonstrative Pronouns without *-ha:/*	**Gloss**	**Number**	**Gender**	**Distance**
/ðajja:/	“this”	SG	M	Near
/tajjih/	“this”	SG	F	Near
/ðawlajja, ðawla:ʔi:/	“these”	PL	M or F	Near
/ðajja:-k/	“that”	SG	M	Far
/tajji-k/	“that”	SG	F	Far
/ðawlajja-k, ðawla:-k/, ʔawla:-k, ʔawla:ʔi-k/	“those”	PL	M or F	Far

Locative Demonstrative Pronouns

Locative demonstrative pronouns are used to identify the distance of places as near or far. Morphologically, unlike demonstrative pronouns, they do not display gender or number distinction. Table 2.18 lists Locative demonstrative pronouns in San’ani Arabic.

Table 2.18 ***Locative demonstratives in San’ani Arabic***

Locative Demonstrative	**Gloss**	**Distance**
/ha:na:, hinijjih/	“here”	Near
/ha:na:-k(a), hinjjaka/	“there”	Far

Indefinite Demonstratives (IDs)

Indefinite demonstratives have similar properties as locative demonstratives. In this sense that, they distinguish near and far only. Table 2.19 shows indefinite demonstratives in San'ani Arabic.

Table 2.19 *Indefinite demonstratives in San'ani Arabic*

Indefinite Demonstratives	Gloss	Distance
/ha:kaða:/	"like this"	Near
/kaða:/	"like this"	Near
/ha:kaða:-k/	"like that"	Far
/kaða:-k/	"like that"	Far

3. Adjectives

Adjectives are words used to describe nouns or pronouns. In Arabic as well as San'ani Arabic, Adjectives inflects for gender, number, and definiteness. For instance, the adjective */ʔa-tˁawjl-a:-t/* "the tall-(PL F)" contains the definiteness marker /ʔa/, plural and feminine markers /a:-t/. In San'ani, both comparative and superlatives adjectives are called elatives. They are formed using these three patterns /ʔaCCaC, ʔaCCa, ʔaCaCC/. for example, the adjective /ɣa:lj/ "expensive" is converted into elative using the second pattern /ʔaCCa/, so it becomes /ʔaɣla/ "more expensive."

4. Participle (gerund)

Verbal derivatives in San'ani Arabic are "descriptive words derived from particular stem classes, or forms, of a verbal root" (Ryding 2005, 102). Two participles are found in San'ani Arabic and they are based on a distinction in voice: Active or Passive.

The active participle refers to the doer of the action and the passive participle refers the entity having undergone the action of the verb. Active participles are derived from the basic verbal pattern or Form I[4] (i.e., CVCVC pattern), of the triliteral verb by placing it in the pattern /Ca:CiC/ for singular forms. The passive participle is formed using the pattern /maCCu:C/, as shown in Table 2.20.

Table 2.20 *Formation of singular AP and PP in San'ani Arabic*

Form I	**Active participle**	**Passive Participle**
/*daras*/ 'to learn'	/*da:rɪs*/ 'learner (M SG)'	/*madru:s*/ 'learned'
/*labɪs*/ 'to wear'	/*la:bɪs*/ 'wearer (M SG)'	/*malbu:s/* 'worn'
/*saħar*/ 'to fascinate'	*/sa:ħɪr*/ 'fascinator (M SG).'	/masħu:r/ 'fascinated'
/katab/ 'to write'	*/ka:tɪb/* 'writer (M SG)'	/maktu:b/ 'written'
/*daʕas*/ 'to tread'	/*da:ʕɪs/* 'treader (M SG)'	*/madʕu:s*/ 'trod'

[4] The most common patterns of this form in SA include: /CaCC, CiCC, CuCC, CaCCih, CaCaC, CaCu:C, CuCu:C, CaCi:C, CaCaCih,maCCaCih, maCCaC, miCCa:C, CiCa:Cih, CaCa:C, CiCa:C/, (Watson, 1993, p. 436).

5. Numeral

In San'ani Arabic, numerals are divided into cardinal and ordinal numbers. The Cardinal and Ordinal numbers from 1to 12 are shown in Table 2.21. Ordinal numbers inflect for definiteness, gender, and number. Moreover, cardinal numbers from eleven to nineteen are affixed with the addition of the suffix/ -ar/ only when they are followed by a noun, as in */ʔaθna:ʕaʃ-ar kita:b/* "twelve books".

Table 2.21 ***San'ani Cardinal and Ordinal Numbers from 1to12***

Cardinal	**Gloss**	**Ordinal**	**Gloss**
wa:ħid	"one"	*ʔawal*	"first"
ʔaθni:n	"two"	*θa:nij*	"second"
θala:θih	"three"	*θa:liθ*	"third"
ʔarbaʕah	"four"	*ra:biʕ*	"fourth"
xamsih	"five"	*xa:mis*	"fifth"
sitih	"six"	*sa:dis*	"sixth"
sabʕah	"seven"	*sa:biʕ*	"seventh"
θamanijih	"eight"	*θa:min*	"eighth"
tisʕah	"nine"	*ta:siʕ*	"ninth"
ʕaʃarih	"ten"	*ʕa:ʃir*	"tenth"
xada:ʕaʃ ħada:ʕaʃ	"eleven"	*ħada:ʕaʃ*	"eleventh"
ʔaθna:ʕaʃ	"twelve"	*ʔaθna:ʕaʃ*	"twelfth"

6. Verbs

In Arabic as well as San'ani Arabic, the verb typology is based on the number of radicals, i.e., the number of consonantal roots. There are three types of verbs bi-radical, tri-radical, and quadri-radical. The bi-radical verb is the one that consists of only two consonantal roots, such as */q-m/* "the notion of standing or doing". On the other hand, the tri-radical contains three consonantal roots as in */d-x-l/* "the notion of entering or getting in," and they constitute the majority of verb entries. The last type, i.e., quadri-radical, is a four consonantal root, e.g., */d-ħ-r-ʤ/* "the notion of rolling". In Arabic, to form verbs, these radicals are combined with vowels according to certain patterns known as */wazn/*. In the case of the tri-radical type, there are ten patterns for verb formation, while the quadri-radical type has only four patterns.

Classification of Verbal Roots

Another famous classification for verbal roots is based on the presence of semi-vowels, or what is known as the weak letters /ʔ/, /w/, or /j/, in the verbal root. They are classified into intact */sʕaħıħ/* and defective */muʕtal/* roots. The intact verbs are those which do not have weak letters in their construction, which are further divided into three sub-types. The defective verbs are those which contain weak letters in their construction and are further classified into two sub-types. Table 2.22 clearly shows this division of verbs along with examples.

Table 2.22 *Verbal root classification in San'ani Arabic*

	Intact Roots			**Defective Roots**		
Sub-type	**sound roots**	**hamzated roots**	**geminated roots**	**quasi-sound, hollow**		**defective roots**
				Root-initial	**Root-middle**	
Example	*d-r-s*	*q-r-ʔ*	*m-d-d*	*w-ʤ-d*	*sˁ-a:-m*	*ʤ-r-j*

Gloss	"the notion of studying"	"the notion of reading"	"the notion of stretching"	"the notion of finding"	"the notion of fasting"	"the notion of running"

7. Prepositions

A preposition can be defined as a word that precedes a noun or a pronoun to express a relation with another word. In general, Arabic identifies two types of prepositions: non-derived prepositions or true prepositions */ħuru:f al-ʤarr/* and the derived prepositions or semi-prepositions */ðʕuru:f maka:n wa-ðʕuru:f zama:n/*"adverbs of place and time." San'ani Arabic is no exception. The non-derived prepositions have two types bound and free. The bound prepositions are mono-radical, while free prepositions can be biradical or triradical. Table 2.23 shows San'ani Arabic non-derived prepositions.

Table 2.23 *San'ani Arabic non-derived prepositions*

Bound prepositions	**Gloss**	**Free prepositions**	**Gloss**
bɪ-	"in, with"	*mɪn*	"form"
lɪ-, la-,la:	"to, for"	*ʕala:*	"on, on to, above"
fɪ-	*"in"*	*fɪ:*	"in"
ka-[5]	"like"	*maʕ*	"with"
		ʕan	"from"

8. Coordinating Conjunctions

[5]It is only used with the demonstrative pronoun */ðajja:/ "this"as /ka-ðajja:/* "like this."

Coordinating Conjunctions are known in Arabic as the connectives */ħuru:f al-ʕatf/* are words or phrases that connect clauses, sentences, or other parts of the discourse together. Al-Batal (1994) defines this term as "any element in a text which indicates a linking or transitional relationship between phrases, clauses, sentences, paragraphs or larger units of discourse, exclusive of referential or lexical ties" (Al-Batal, 1994, 91).

In San'ani Arabic, Watson 1993 distinguished two types of connectives: conjunctions and poly-syndetic conjunctions; each contains four conjunctions, as seen in Table 2.24.

Table 2.24 ***Connectives in San'ani Arabic***

	Connective	**Gloss**
Conjunction	*wa-*	"and"
	aw/awla:/walla:	"or"
	fa-	"and then, and so, yet, but, and also, moreover, and therefore, in conclusion".
	bass, la:kn	"but"
Polysyndetic	*ja: ... ja:*	"either … or"
Coordination	*(ja:)ɔamma: ... (ja:) ɔamma:/aw/awla:/walla:*	"either …or"
	sawa: ...aw/walla:	"whether … or"
	sawa ... wa-	"both … and"

As shown in table 30, some conjunctives have alternative allomorphs. Thus, the conjunctive /*aw/* 'or' has two allomorphs, /*awla:*/ and /*walla:*/.

9. Adjunctions/subordinating conjunctions

Subordinating conjunctions/ Adjunctive adverbs are connectors that attach clauses or sentences together. Watson (1993, 339-40) provided ten types, including the following: time,

concession, Universal conditional-concession, Alternative conditional-concession, place, Manner and comparison, and Reason and purpose. Table 2.25 provides Watson division of adjunctions with examples.

Table 2.25 *Watson division of Adjunctions in San'ani Arabic*

Type	adjunction	Gloss
Time	*law-ma:*	"when, while, until"
	law	"when, until"
	lamma:/lamman/amma:	"when, until"
	tɪʤa:h-ma:	"before"
	gabl-ma:	"before"
	baʕd-ma:	"after"
	ʔawwal-ma:	"at first, as soon as, in the past"
	ħɪ:n-ma:	"when, at the time that"
	ħɪ:n	"when, at the time that"
	ʕɪnd	"when, at the time that"
	ʕɪnd-ma:	"when, at the time that"
	ħa:l-ma:	"when, at the time that"
	ka-ma:	"just, as, when"
	ħatta:	"until"
	wagt-ma:	"when, at the time that"
	jawm (-ma:)	"the day that, when"
	sa:ʕat-ma:	"the hour that, when"
Concession	*bajn-ma:*	"while"
	bɪ-rayɣm-ma:	"in spite of"
	ʕala:-ma;	"considering that"
	ʔɪnna-ma:	"but, however"
	badal-ma:	"instead of"
	bɪ-du:n-ma:	"without"
Universal conditional-concession	*mahma:*	"however, whatever"
	ʔajn-ma:	"wherever"
	ʔajn	"wherever"
	ʔajjaħɪ:n-ma:	"whenever"
	kull-ma:	"whenever"
	man	"whoever"
	ma:	"whatever"

Manner and comparison	*ka-ma:*	"as"
	mɪθəl-ma:	"as, like"
	ʕala:-ma:	"as, depending on"
	sa:ʕ-ma:	"as, like"
	kam-ma:	"as much as"
Place	*ħajθ-ma:*	"where"
Reason and purpose	*mɪn sɪbb/ʕasɪbb*	"because, for, so that, so"
	ħatta:	"in order that"
	ma: da:m	"so long as, since"
Concession	*wa-law*	"even if, even though"
	wa-ʔɪn	"although"
Negative condition	*(ma:…) ʔɪlla:*	"not … unless"

These conditions are cited from Watson (1993, 339-40).

10. Adverbs

Adverbs in San'ani Arabic are classified by Watson (2008) into four types: temporal adverbs, local adverbs, manner adverbs and degree adverbs. Temporal adverbs refers to the adverbs which express time such as: /*ðalħi:n*/ "now", /*ʔams*/ "yesterday". Local adverbs refers to the adverbs that denotes location such as /*ha:na:/* "here"*/ha:na:k/* "there". Manner adverbs express the way of the action or haw it is done such as: */ha:kaða:/* "like this"*/bisa:ʕ/* "quickly". Degree adverbs express the degree of emphasis or quantity; such as: */gawijah/* "very" /*xajira:t*/ "a lot".

11. Particles

The class of particles in San'ani Arabic constitutes the greatest variety of types of closed-list systems. They belong to a closed class system that has its unique linguistic properties. In terms of form, particles are always indefinite, mostly mono-radical or biradical.

Morphologically, particles do not show inflection as they are inherently invariable. Moreover, as function words, they do not carry an actual semantic meaning; however, they depict effects such as negation, contrast, and emphasis. Syntactically, unlike other open classes, particles do not take the place of a subject or a predicate in a sentence (Watson 1993). Some of the particle types in San'ani Arabic are negative, conditional, vocative, and interrogative particles.

A. Negative particles

They are particles used to express negation in any part of a predicate. Some examples of negation particles in San'ani are: /*ma:ʃi:*/ "no", /*ma:*/ "not" and /*la:/ma: ....wa-la:/* "'neither… nor' " as in */ma:daxal wa-la: xaraʤ/ "neither he went out nor he stayed in ".*

B. Conditional Particles

In San'ani Arabic there are four conational particles /*ʔıða: , ʔın , (ʔı)la,,* and *law/* .They all can be glossed as "if".

C. Vocative Particles

There is a single vocative particle in San'ani Arabic which is */ja:/*. It usually occurs before the name to be called as it denotes the supposed verb of call */ʔuna:di /* "I am calling." '

D. Interrogative particles

In San'ani Arabic Arabic interrogative particles are the question words which occur initially in a clause or a sentence to verify or ask about something. They are similar

to the wh-word in English. Some examples are: /*ʔajjaħɪ:n*/ "when", /*ʔajn/wajn/* "where", /*kajf/* "how", /*lɪlma:/* "why" and /*kam/* "how much/many" .

12. Interjections

Interjections refer to the expressions that illustrate feelings such as: */ja: lat^{ʕ}i:f/* "oh my God", */aha:/* "Sound for assertion" */ja:sala:m/* "wow".

2.3.4. Syntax

This section provides a general overview of Arabic syntax in general and San'ani in specific. It defines some basic syntactic terms. Then it depicts the different types of sentences in Arabic. It finally comments on the Syntactic divergence between MSA and San'ani Arabic.

2.3.4.1. Definitions of Basic Syntactic Terms

a) Syntactic word

According to Watson (1993), a syntactic word is defined as a word that can function independently in syntactic construction, such as the conjunction /*wa-*/ "and". However, any dependent word, such as verbal subject or annexing pronouns, cannot be called syntactic. Beeston (1970) justifies it because such words cannot start an utterance and fail to separate from the preceding word with a sensible word. For instance: The verb */niktub/* "we write consists of the dependent subject pronoun */na-/* and the verb */katab/*. */na-/* is considered a bound morpheme that functions only attached to the verb.

b) Clause

It refers to the syntactic structure, consisting of at least a predicate. However, generally, it contains a subject and a predicate (Wastson, 1993) as in:*/fatahat alba:b/* "she opened the door"

/fatahat/ is a verb, and a verb phrase, while */alba:b/* is a noun and noun phrase. Interestingly intransitive verbs can function as a syntactic word, a phrase, and a clause simultaneously. For instance,*/xaraʤ/* "he went out" is a word, a verbal phrase, and a clause at the same time.

c) Complex Clause

A complex clause is a clause that contains at least one subordinate and one superordinate clause. It can also have more than two clauses as in:

/ ʔiftahij talafuwniʃ ʕala: sibb lawma: ʔatasil liʃ tiʤa:wbi/

"switch on (2 F S) your phone, so that when I call you, you answer (2 F S)"

d) Sentence

The term sentence in traditional Arabic grammar is considered as a central unit where the grammatical theory function (Al-Kohlani 2015), so two terms fall under the sentence in Arabic, which are: */jumlah/* "sentence" and */kala:m/* "speech". The first term, i.e., */jumlah/* depicts a dependent clause that consists of a predicate and does not need to be informative. The second term */kala:m/*, however, refers to the syntactic structure which is independent syntactically; i.e., it can stand alone and can express a complete thought (Ibn Jinni 1983, 17). This means that the term sentence is equivalent to */kala:m/*.

e) Compound Sentence

It is a syntactic structure that comprises of two or more sentences joined by a conjunction (Wastson, 1993), as in:

/alʤahal xaraʤu jilʕabu balguri:h wa-lbana:t ʔftaʤaʕi:n wa-sa:ħi:n gawijah/	*"the boys went out playing with fireworks and the girls got afraid and cried loudly"*

2.3.4.2. Sentence Types

The traditional Arabic grammar divides sentences into two types: nominal and verbal sentences. This distinction is determined by the word class at the beginning of a sentence. When the sentence starts with a noun, it is called a nominal sentence, and when it starts with a verb, it is called a verbal sentence. So, in traditional Arabic grammar terminology, there are two types of sentences:

- Nominal Sentences
- Verbal Sentences

Syntactically San'ani Arabic acts freely and deviates from many Classical Arabic and MSA syntactic rules. For example, the San'ani Arabic word order is more flexible in general. For instance, adjectives in MSA are to come after nouns and not to precede them, but in San'ani Arabic, they can come prior to nouns for emphasis (Watson 1993). For example, the adjective /*kabi:rih*/ "large" in /*ʇajja:rih kabi:rih*/ 'a large airplane' can also come before the noun as in /*kabi:rih ʇajja:rih*/ 'airplane large'. Besides, nouns and adjectives can be separated by the indefinite demonstrative /*hakað*a/ 'like this' e.g., /*ʇajja:rahha kað a kabi:rih*/ literally 'airplane like this large' which means "a large airplane like this".

2.3 Summary

This chapter contains an introduction to Arabic language history and its forms. It clarifies the relationship between CA, MSA and Dialectal Arabic. The main parts of the chapter are allotted to San'ani Arabic. First it is introduced geographically and publicly. Then, the structure

of San’ani Arabic is described to reflect a clear view of the nature of San’ani Arabic dialect. The structural overview is divided into four main parts orthography, phonology, morphology and syntax of San’ani Arabic which are presented to enable a vivid understanding of the dialect in hand.

CHAPTER THREE
LITERATURE REVIEW

This Chapter investigates parts-of-speech tagging literature. It consists of five sections. The first section introduces parts-of-speech tagging approaches and methods. The second section provides a chronological survey of the history of parts-of-speech tagging into two sub-sections. The first sub-section reviews parts-of-speech tagging in non-Arabic languages, while the second reviews parts-of-speech tagging in Arabic. The third section examines the existing Arabic tagsets. The fourth section investigates dialectal Arabic corpora and discusses the research gap and the proposed solution. The last section is a summary of the Chapter highlighting essential remarks.

3.1 Parts-of-Speech Tagging Methods

In automatic parts-of-speech tagging, many issues arise while assigning tags to words in a running text. For instance, when tags are attached to words, they are applied to both words and punctuation marks, which is not easy for the system to differentiate. Another challenge is the ambiguity where a word has more than one possible part-of-speech tag. For example, the same word can be a noun and can be a verb as well. These issues make it a challenging task for the machine. Therefore, researchers experimented using different methods to perform automatic parts-of-speech tagging, to resolve these ambiguities and achieve optimal results.

Many scholars investigated parts-of-speech tagging approaches in general and concerning their languages (Van Guilder 1995; Hasan 2006; Rathod and Govilkar 2015; Kumawat and Jain 2015; and Awwalu, Abdullahi and Evwiekpaefe 2020). However, most scholars followed the

general classification proposed by Van Guilder (1995). Therefore, we here present his classification.

Van Guilder (1995) classified automated parts-of-speech tagging into supervised and unsupervised. His distinction was regarding the degree of automation of the training and tagging process. For him, supervised taggers typically depend on a pre-tagged corpus to serve as the basis for building any tools to be used when tagging, for instance, the tagger dictionary, the word/tag frequencies, the tag sequence probabilities, and/or the ruleset. However, unsupervised models do not require a pre-tagged corpus but instead, use computational methods to induce tagsets automatically. Then based on the automatic grouping, it either calculates the probabilistic information required by statistical tagging systems or induces the context rules required by rule-based systems. Figure 3.1 illustrates Van Guilder's classification of the approaches of Parts-of-speech tagging.

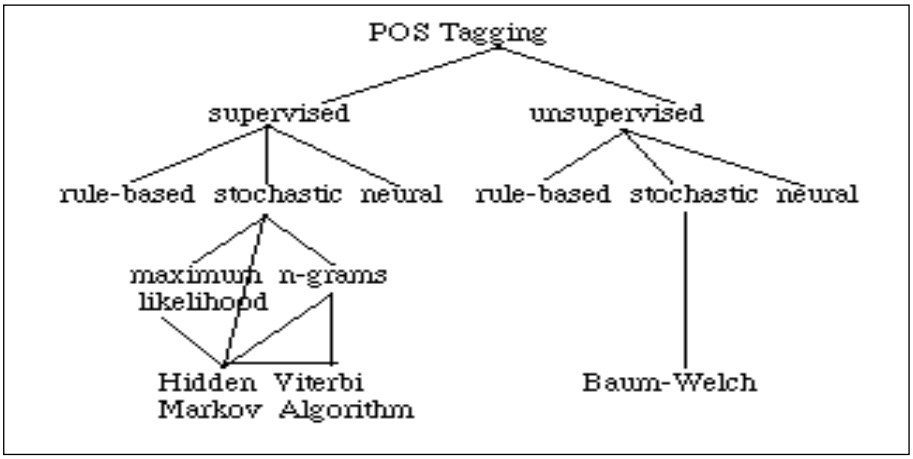


Figure 3.1 *Van Guilder's classification of the approaches of Parts of Speech Tagging*

Source: Van Guilder, 1995 .Automated Parts-of-speech tagging: A brief overview, Handout for LING361, http://ccl.pku.edu.cn/doubtfire/NLP/Lexical_Analysis/Word_Segmentation_Tagging/POS_Tagging_Overview/POS%20Tagging%20Overview.htm

Adopting Van Guilder's classification, parts-of-speech tagging methods of today are depicted in Figure 3.2. These methods are classified into supervised and unsupervised. The rubric of the supervised parts-of-speech tagging technique depends on using pre-tagged data as a prerequisite. While in the unsupervised parts-of-speech tagging technique, there is no need for pre-tagged data. Here we highlight the types of tagging schemes commonly used today, although no particular system will be discussed.

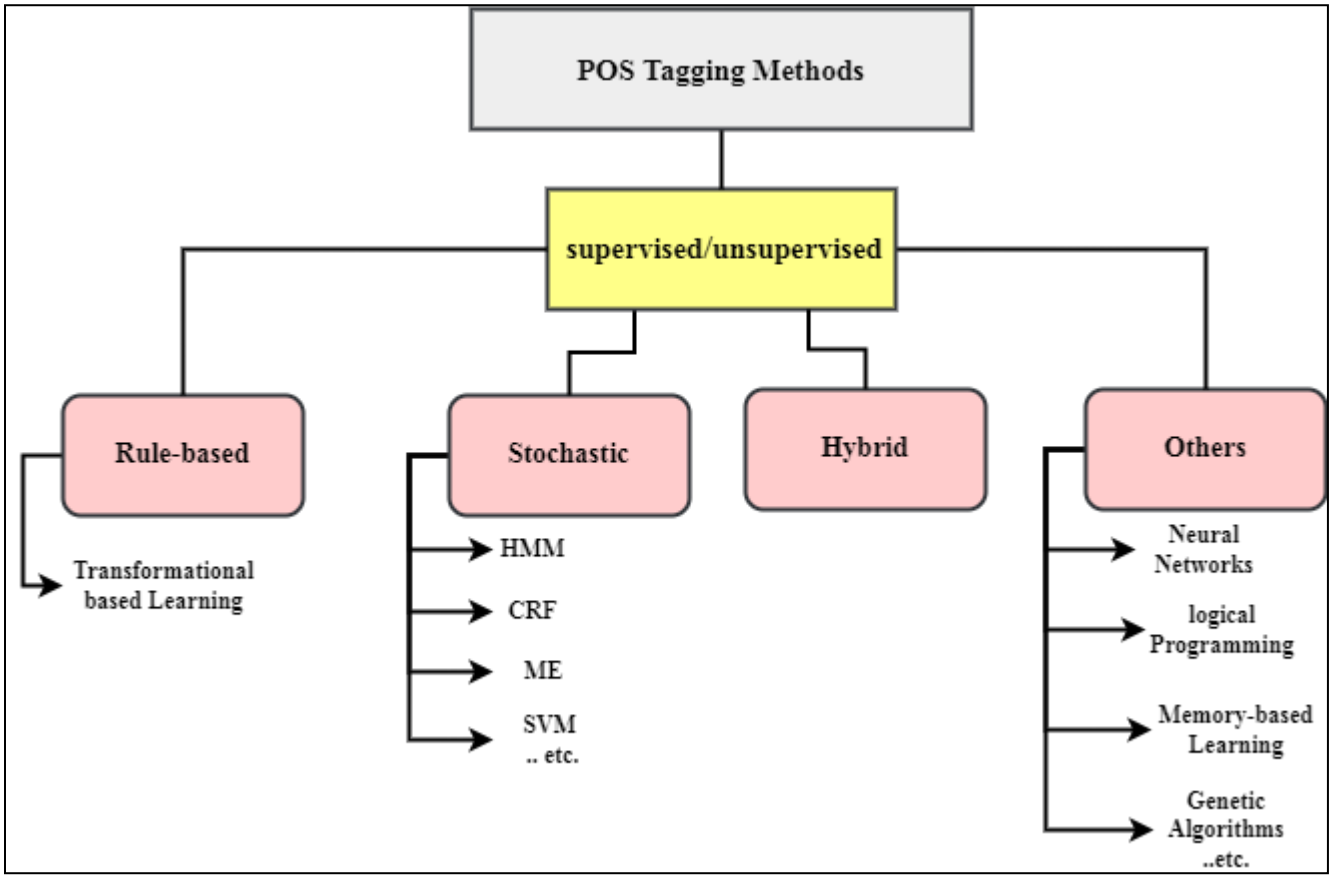


Figure 3.2 *Parts-of-speech tagging methods*

3.1.1 Rule-based parts-of-speech tagging

It is the earliest algorithm for automatic parts-of-speech tagging. There are two stages involved in this technique. The first is applying a dictionary lookup to assign a list of possible parts-of-speech tags. The second involves using a large list of hand-written disambiguation rules

to declare a single correct tag for each word in a sentence. Usually, contextual information is used to formulate a set of linguistic rules that identify the proper tag to be assigned to a word in a sentence. Such rules are known as context frame rules Linda Van Guilder (1995) provided the following example:

det - X - n = X/adj

It is illustrated as: if an X word / an unknown word occurs between a determiner and a noun, respectively, it is tagged as an adjective. Additional linguistic information that is used for rule-based tagging is morphological information.

The rule-based tagging technique is most commonly applied to supervised training. Lately, researchers have been trying to generate the disambiguation rules automatically instead of a hand-written list of rules.

3.1.1.1 Transformational-based tagging

It is an advanced application of the rule-based method (Khoja 2003). The idea lying behind transformational-based learning (TBL) is to assign the most likely tag to a word and then go back and correct the mistakes. TBL is considered a supervised learning method that involves the following stages of operation:

1. First, it gives every word the most-likely tag

2. Then, it examines the transformation and selects the most improved tagging

3. Finally, it replaces the incorrect tags accordingly.

3.1.2 Stochastic Parts-of-speech tagging

It can also be called statistical parts-of-speech tagging. This technique uses probabilities to assign tags. It may use lexical information as well as contextual information. There are several algorithms and models to calculate and disambiguate parts-of-speech tags, such as:

- **Hidden Markov Model**

Hidden Markov[6] Model (HMM) is one of the most popular statistical models for parts-of-speech tagging. Basically, it uses tag sequence probabilities and word frequency measurements. It makes poor use of contextual information as it assumes context independence. In fact, its probabilistic depends on the n previous tags only. HMM probabilistic can be presented using the following formula:

P (word | tag) * p (tag | previous n tags)

HMM is usually implemented using the **Viterbi Algorithm[7]**. Viterbi Algorithm is known as the n-gram approach, which denotes the fact that a correct tag for X word is identified by the probability of the previous context, i.e., when it occurs with n previous tags. It is used to get the most likely sequences of hidden states.

[6] For more information on Hidden Markov Model refer: Güngör, Tunga. Parts-of-speech tagging. In: Indurkhya Nitin, Damerau Fred J, eds. Handbook of Natural Language Processing. 2nd ed. Boca Raton, FL: Chapman and Hall/CRC Press; 2010, 205–236

[7] For more information on Viterbi Algorithm refer:
Brill, Eric & Marcus, M. 1993. Tagging an unfamiliar text with minimal human supervision. ARPA Technical Report.

- **Maximum Entropy Model**

Maximum Entropy (ME) Model is based on the ME principle. As the name suggests, the principle states that the most reasonable probability to model certain data is the one with the highest entropy (Guiasu and Shenitzer 1985). The significant advantage of ME model to parts-of-speech tagging is that it is flexible in accommodating context compared to HMM, which makes restricted use of contextual information (Güngör 2010). Güngör states that the contextual features used by ME models can be simple or complex and not necessarily independent.

- **Conditional Random Fields**

Conditional Random Fields (CRF) is a discriminative model that calculates conditional probability distribution as opposed to generative models, such as HMM, which aim for a joint probability distribution. In mathematical notation CRF represents the probability of $\boldsymbol{P}(\boldsymbol{y}|\boldsymbol{x})$ where *P* is the probability of, $\boldsymbol{y}$ is the output vector and $\boldsymbol{x}$ is the input sequence. Generative classifiers, on the other hand, attempts to calculate the $\boldsymbol{P}(\boldsymbol{y},\boldsymbol{x})$. CRF is explained in detail in Chapter Six, section 6.2.2.

.

3.1.3 Hybrid Parts-of-speech tagging

Sometimes Parts-of-speech tagging is performed by combining rule-based methods along with stochastic methods. Such tagging method is known as hybrid tagging. When applying hybrid tagging, statistical tagging is applied first, and then the rule-based component takes care of resulting errors. The CLAWS is one of the earliest and most famous English hybrid taggers

developed by Garside (1987). This tagging technique is known for achieving higher accuracy rates than statistical and rule-based tagging.

3.1.4 Others

In addition to the above-discussed methodologies, several innovative techniques are introduced to parts-of-speech tagging. Researchers utilize artificial intelligence and machine learning techniques for parts-of-speech tagging. Some of these techniques are:

- **Neural networks**

Neural networks, which are also known as artificial neural networks (ANNs), belong to machine learning and are one of the deep learning algorithms. Neural networks techniques can be classified as supervised or unsupervised.

ANNs are designed to mimic humans' brain behaviour, whereas their structure reflects the signals between biological neurons. In general, neural networks are composed of multi-layers of nodes where the input layers are at the bottom, the output layer is at the top, and the middle layers are hidden. The nodes are connected with weighted links and have a threshold. The threshold is what allows the output of one layer to proceed to the next layer.

Neural networks application in parts-of-speech tagging performs sequence labelling. The input consists of all the information of a current token's possible tags and a certain number of preceding and following tags. At the same time, the output displays the token's appropriate tag. The weights on the connection are what allow the labelling, so they are adopted. By the end of the learning, the weights associated with the tags are saved as they are used to perform the tagging. Recurrent Neural Network, a type of ANN, is explained in detail in Chapter Six, section 6.2.1.

- **Memory-based learning**

  Memory-based learning is also called instance-based learning. It works through identifying similarities between the new test data and the training data, i.e., it searches in the training data for the most similar items to the test items and then gives an appropriate prediction. One of the significant advantages of MBL is that it can store the training set.

- **Logical programming**

  It is a programming paradigm that is based on formal logic. It can be used instead of imperative programming languages in implementing parts-of-speech taggers. It has the advantage of transparency and easy understanding of the model.

- **Genetic Algorithm**

  It is a search-based optimization method that is based on the concepts of genetics and natural selection. It belongs to the evolutionary algorithms (EA). It is efficient in producing solutions to optimizations and search problems.

3.2 History of Parts-of-speech tagging

This section presents a historical review of the research done on parts-of-speech tagging. It discusses studies in chronological order into two sub-sections. The first sub-section deals with the history of parts-of-speech tagging in Non- Arabic Languages, where the focus is given to the prominent contributions in the field. The second sub-section investigates parts-of-speech tagging in the Arabic language.

3.2.1 History of Parts-of-speech tagging in non-Arabic languages

3.2.1.1 1960s

Parts-of-speech tagging started in the early 60s. It was not known as parts-of-speech tagging and was usually a component of more extensive systems rather than separate systems. The role of parts-of-speech tagging in NLP as a fundamental step in text processing was acknowledged during this decade.

However, parts-of-speech taggers were mostly built using rules due to the vast influence of Chomsky's theory of language innateness (Khoja 2003).

Harris (1962) developed a program capable of decomposing a string of sentences into its elementary analytic components, taking into consideration the placement of adjuncts. It is known as the sentence recognizer. According to him, a sentence structure can be described using three equally powerful analyses. They are String Analysis, Constituent analysis, and Transformational analysis. The first processing stage is a dictionary lookup, where every component gets a category. If more than one category is provided, specific tests are performed to choose the most probable category based on certain cues such as neighbouring words and context. The second stage of the program provides a plausible category by analysing the sequence of categories to reflect well-formed sentences. The tagset used is not listed, but the main categories are reported to be used.

A computational Grammatical Coder (CGC) is what **Klein and Simmons (1963)** called their developed parts-of-speech tagger. It is a part of a fully functioning syntactic analysis system. Klein and Simmons considered the CGC as an alternative for using large lookup lexicons. They criticize the need for such dictionaries, which necessitates the storage of a large

number of entries of at least 25,000 or 27,000 and their information. CGC is built using a set of small size dictionaries containing less than 2000 entries to cover function words. The tagset used consists of 30 tags.

The input is passed through several stages of processing. If a token is an exception in one stage, it is stored as an exception to be passed to the next test stage to get resolved and so on. The first step of the CGC pipeline is a dictionary look to tag function words with their unique tags. Then a capitalization test is performed to tag capitalized words within the sentence. After that, suffix tests are used, checking the words' final characters to tag the rest of the text.

Some words are assigned more than one tag. Such words are directed to the next step, the context frame test so that, each word is assigned a single suitable tag. The CGC was tested using a scientific text from the same corpus used reporting 90% accuracy in the evaluation stage. The remaining unresolved text is tagged using further syntactic and semantic analysis if needed.

Stolz et al. (1965) introduced one of the earliest statistical parts of speech taggers. It was known as WISSYN grammatical coder. The system structure resembles that of Klein and Simmons (1963). Like Klein and Simmons, dictionaries of small sizes are used to process function words and frequent lexical words. The WISSYN grammatical coder processes the input through a number of stages. The first stage is the dictionary lookup, which tags all the function words and closed classes such as pronouns and articles. Nearly 60 to 70% of the words are tagged in this stage. On the other hand, the open classes are run through the next stages for processing. The second stage is the use of morphological cues, namely suffixes stored in small dictionaries, to match the input and assign a suitable tag. The third stage is ad-hoc rules, where the most likely tag is predicted based on the context. Through this stage, up to 10% of words are

tagged. The fourth stage uses a set of previously calculated conditional probabilities to predict an appropriate tag considering the previous and following three classes. This stage defines nearly 20% of ambiguous grammatical classes. It can be noticed that the first two stages process the word in isolation while the last two stages deal with the sentence as a whole. Stolz et al. used a manually tagged corpus of 28,500 words to calculate the probability. The tagset used consists of 18 tags, out of which two refer to punctuation marks. The system accuracy reported is 92.8%.

3.2.1.2 1970s

In the seventies, research had little interest in parts-of-speech tagging. The attention was mostly directed to corpus linguistics. The following studies present the prominent contributions made in Parts-of-speech tagging.

Greene and Rubin (1971) established a rule-based Parts of Speech tagger called TAGGIT. They used the model followed by (Klein and Simmons, 1963). However, the size of the lexicon and tagset is bigger. The lexicon, which was called the word list, contains 3000 entry and the tagset, known as the tag system, consists of 71 tags. The pre-processing stage handles multiword units by joining them together to get a single tag. Like Klein and Simmons (1963), Greene and Rubin used Suffix lists to tag tokens following the same procedure explained earlier.

Additionally, rules were used to tag capitalized words, numbers, and so on. The rest of the tokens which are not covered by the lexicon, rules, or suffix list are given three tags which are namely Noun/singular (NN), Verb (VB), and Adjective (JJ). The following stage relied on what is known as Context Frame Rules, where 3300 positive and negative context rules are used to disambiguate words with more than a single tag. The context frame rules consider two proceeding and following tags to the target tag. The format of these rules is as the following:

$A\ B\ ?\ D\ E \rightarrow C$

It means if there is an ambiguous tag *?* proceeded by *A* and *B* tags and followed by *D* and *E* tags, respectively, then the ambiguous tag is *C* (Khoja 2003). These rules are extracted using 900 manually annotated sentences from the Brown corpus (Kučera & Francis 1967). Eventually, TAGGIT was used to tag the Brown corpus reporting a 77% accuracy rate as described by Kučera and Francis (1982) (Jurafsky& Martin 2020).

Bahl and Mercer (1976) developed a stochastic parts-of-speech tagger. Viterbi algorithm along with HMM were used for tagging. Their tagger was trained on 40,000 tokens, reporting a 98.6% accuracy rate (Khoja 2003).

3.2.1.3 1980s

In the 80s, highly accurate parts-of-speech tagging systems were built using probabilistic approaches. In addition, neural networks methods for Parts-of-speech tagging were introduced. The following studies reflect the progress made during this decade.

Garside (1987) proposed the Constituent-Likelihood Automatic Word-Tagging System (CLAWS1). CLAWS1 was developed between 1981 and 1983. It is built using a hybrid method, i.e., both statistical and rule-based methods. A basic form of HMM is used to calculate and predict lexical and contextual probabilities. At the same time, the rule-based method is used to tag exception words, multiword units, clitics, and so on. They used 200k tokens from the Brown corpus to account for the probabilities and train the model. The Tagset used contains 133 tags that are adopted from the tagset used by Greene and Rubin (1971). The main aim for developing CLAWS1 was to tag the Lancaster-Oslo/Bergen (LOB) corpus (Khoja, 2003). The accuracy reported was 96-97%.

It is worth mentioning that Lancaster University kept the CLAWS project under continuous improvement producing several versions. The current Versions of the CLAWS, i.e., CLAWS4, is applied to tag the British National Corpus (BNC) of 100 million words, achieving a 97% accuracy rate (Khoja 2003).

Church (1988) proposed a stochastic parts-of-Speech tagger called PARTS. It is similar to CLAWS; however, it misses the rule-based component. The tagger adopts a trigram model calculating lexical and contextual probabilities using a linear time dynamic programming algorithm. PRATS was trained using the Brown corpus. Khoja (2003) stated that Marcus et al. (1993) used PRATS for tagging the Penn Treebank, reporting a 95-97% accuracy rate.

One of the earliest neural network applications on parts-of-speech tagging is **Benello et al. (1989)**. The tagger was built using a backpropagation neural network of 560 units and two layers of modifiable connections. It depicts human behaviour to tag a text. In addition, it uses context where six windows of a word are used to perform tagging. The training was done using 900 sentences of the Brown corpus belonging to the Romance genre. Moreover, words from the training data were also looked up in the Brown corpus to account for the possible tags. The accuracy reported on the test set is 95%.

3.2.1.4 1990s

During the 1990s, the importance of Parts-of-speech tagging became well established. In fact, parts-of-speech tagging was acknowledged as a fundamental step of different linguistic analyses that facilitate most high-end NLP tasks. Therefore, researchers' attention was drawn to develop and improve Parts-of-speech tagging systems using available resources, which are pretty much the case of English. For non-English languages, since resources were lacking, work was

directed to data collection and annotation to be used for NLP tasks. Guided by the early literature, English tagging techniques were successfully applied to other languages, especially Western European languages (Khoja 2003). Moreover, Technological advancement in computer sciences at that time equipped the field with new techniques to be used in different NLP tasks, including parts-of-speech tagging.

Cutting et al. (1992) developed an unsupervised statistical tagger for English. The tagger used HMM for complete flexibility in selecting the training data as justified by (Cutting et al.). The idea of proposing a tagger without relying on the availability of a decent size annotated corpus was meant for other languages that lack data resources. As English, by that time, had two annotated corpora, namely Brown and LOB corpora.

Though the tagger does not need a tagged corpus, it requires a lexicon and untagged corpus for training. The tagger was trained using 3,000 sentences. There are four main modules in the tagger: a tokenizer, a lexicon, a training module, and a testing module. The tokenizer processes the input splitting it into words, and identifies sentence boundaries. Then tokens are passed to the lexicon to be tagged. Untagged words are checked for suffixes to get possible tags; otherwise, a default class is given that contains multiple tags of all the possible open classes. After that, the training probabilities are calculated, and text is processed using a bigram HMM and Viterbi algorithms. The Brown corpus is used for training as well as testing. The accuracy reported reached 96%.

Brill (1992) build a parts-of-speech tagger using a new technique of rule-based tagging. His technique involves automatically extracting the rules from a tagged (training) corpus using a transformation-based error-driven learning (TBL) algorithm. The tagger starts with a lexicon

lookup. The lexicon was built from the Brown corpus. After checking the words in the built-in lexicon, tags are given to familiar words; others are tagged based on the suffixation. The suffixation rules are acquired from the training corpus. Moreover, capitalized words, which are not found in the lexicon, are tagged as proper nouns. Then a corpora comparison is conducted to gather the errors. The errors are listed in the form of an error triple shown in the following format <*taga, tagb, number* >. It reads as *taga* is mistakenly used in place of *tagb*, (number of times). TBL then extracts the rules that reduce the errors allowing the rate of accuracy to maximize.

In addition, several modifications were applied to the tagger in 1994 and 1999 (Brill 1994; Brill & Pop 1999). These modifications made the tagger more efficient and increased the accuracy rate. The final accuracy rate reported is 96%.

Kupiec (1992) developed a statistical parts-of-speech tagger using unsupervised HMM. It is similar to the tagger developed by (Cutting et al. 1992). Kupiec used a 200k+ inflected forms dictionary extracted from the Brown corpus along with a tagset of 42 tags (Khoja, 2003). Kupiec used word classes instead of word types that are listed in the dictionary so that he could get rid of data redundancy. This step reduced the 200k+ form into 202 classes, allowing the new words to be included without re-training.

The project pipeline consists of three steps. First, the text is tokenized and normalized. Then, each token is tagged using the dictionary. If words are not found in the dictionary, affixation is used to assign tags to them. Finally, the tagger is trained using the Baum-Welch algorithm. At the same time, the Viterbi algorithm is used to identify the most likely tag or sequence of tags. Unknown words are tagged using the suffixation rules included in the model.

The Brown corpus was used for training. The accuracy rate reached 96.36%. Interestingly, the tagger was applied to the French Language. Kupiec stated that the French tagger was as good as the English tagger, if not better.

Merialdo (1994) experimented on supervised and unsupervised Parts-of-speech tagging to evaluate the performance of both approaches. He used a trigram HMM to experiment. He used both tagged and untagged text to train the Trigram HMM. The corpus used consists of 42,186 sentences, and the lexicon was built of the training corpus words. Merialdo found that training using tagged corpus makes the model more efficient than training on untagged data. He also reported that Maximum Likelihood estimation could negatively influence the performance or accuracy of the tagger (Khoja 2003).

Net-tagger is a neural network-based tagger that was developed by **(Schimd 1994)**. The tagger consists of multilayer perceptron (MLP) networks and a lexicon. The MLP output layer includes all the possible tags in the tagset, and the most likely tag during tagging is activated while others are deactivated. The input layer contains the token lexical probabilities and the following tokens probabilities. On the other hand, the preceding token is already tagged, which enables the use of its activated tag instead of its probability.

In addition, the lexicon is similar to the one used by (Cutting et al., 1992). It consists of a full-form lexicon, a suffix lexicon, and a default entry. During the lookup stage, first, the token is searched into the full-form lexicon; if found, a tag is returned. Otherwise, the capitalization is converted, and the search is repeated. If the token is not found again, then the suffix lexicon is searched to figure out the tag; if not applicable, then a default entry is given. Schimd creates the lexicon from 2 million words of the Penn Treebank Corpus. He first calculated the frequency of word/tag pairs. Any pair's frequency equals 1% is excluded.

The training of the Net-tagger is done on 2 million tokens of the Penn Treebank. The tagger was tested with a 100k token which is not a part of the training data. The accuracy reported reached 96.2%.

3.2.1.5 2000s

In the 2000s, work in parts-of-speech tagging continues to focus on improving parts-of-speech tagging in terms of accuracy and coverage. Several attempts are made to develop language-independent taggers to tag different languages. This section presents some examples of parts-of-speech tagging contributions made in the 2000s.

Gimpel et al. (2010) developed a parts-of-speech tagger for English social media data, namely English tweets. They collected and manually annotated 1,827 tweets (26,436 tokens). The data was divided into 14,542 tokens for training, 4,770 tokens for development, and 7,124 tokens for testing. For annotation, Gimpel et al. proposed a coarse tagset of 25 tags divided into 17 standard parts of speech categories and eight social media categories such as URLs, emoticons, Twitter hashtags, and so on. The system was built using the Conditional Random Fields (CRF) model. The authors incorporated a set of features into the model, including word type feature, suffix feature, capitalization pattern feature, features for domain-specific properties, and external linguistic resources. The system was tested and compared against Stanford Tagger (Toutanova et al. 2003), reporting an 89.37% accuracy rate which reflects a 25% error reduction than Stanford Tagger.

Neunerdt et al. (2013) worked on non-standard German text that is collected from social media. They developed 36,000 token social media text corpus called Web Train. Then they annotated the corpus using the German standard Stuttgart/T¨ubinger Tagset (STTS) (Schiller et al., 1995). The tagset consisted of 54 tags and was used without extension. Moreover, the authors

evaluated four state-of-the-art parts-of-speech taggers' performance on social media text. The four taggers are Tree Tagger (Schmid 1999), TnT (Brants 2000), Stanford (Toutanova et al. 2003) and SVM Tool (Giménez & Màrquez 2004). The training of the taggers was done in 10-fold cross-validation using Web Train and other corpora. Neunerdt et al. (2013) reported that training with in-domain data improves the overall performance of more than five percent. At the same time, training with joint-domain data leads to performance improvement, which is approximately between two and seven percent. In addition, the Tree Tagger outperformed other tested taggers with a 93.72% accuracy rate. Table 3.1 summarizes the history of parts-of-speech tagging in non-Arabic languages.

Table 3.1 *Summary of the history of parts-of-speech tagging in non-Arabic languages*

Author/s	**year**	**Approach/method**	**Language**	**Accuracy**	**Tagset & corpus**
Harris	1962	Rule based	English		-
Klein and Simmons	1963	Rule based	English	90%	30 tags
Stolz et al..	1965	Stochastic approach	English	92.8%	18 tags corpus size is 28,500 words
Greene and Rubin	1971	Rule-based	English	77%	71 tags the Brown corpus
Bahl and Mercer	1976	HMM	English	98.6%	
Garside	1987	Hybrid	English	96-97%	133 tags 200k from the Brown corpus
Church (1988)	1988	stochastic	English	95-97%	the Brown corpus

Benello et al..	1989	neural networks (back propagation)	English	95%	900 sentences from the Brown corpus
Cutting et al.	1992	Unsupervised HMM	English	96%	
Brill	1992	Rule-based	English	96%	the Brown corpus
Kupiec	1992	Unsupervised HMM	English	96%	the Brown corpus
Merialdo	1994	stocahastic	English		the Corpus consisted of 42,186 sentences
Schmid	1994	neural networks	English	96.2%	2 million words of the Penn Treebank Corpus
Gimpel, et al.	2010	CRF	English	89.37%	25 tags the Corpus consisted of 1,827 tweets
Neunerdt et al.	2013	taggers' comparison	German	93.72	STTS of 54 tags the Web train corpus of 36,000 token

3.2.2 History of parts-of-speech tagging in Arabic

As explained earlier, the development of parts-of-speech taggers started as early as the 60s. However, it was not until the early 2000s that researchers-initiated work on Arabic Parts-of-speech tagging for Arabic. It is safe to say that work on parts-of-speech tagging in Arabic was behindhand compared to English and other European languages. Arabic NLP history can be

traced back to the 80s where the focus was given to the morphological analysis and development of rule-based analyzers for Arabic (Darwish et al. 2021).

For Arabic, parts-of-speech tagging slowly acquired researchers' attention as a fundamental task for developing NLP applications such as parsing and information retrieval. This delay of progress in Arabic NLP in general and Parts-of-speech tagging, in particular, is mainly caused by the lack of resources and expertise in the field (Sproat 2007). So, to build parts-of-speech taggers for Arabic data collection, pre-processing and annotation have to be prepared ahead.

This section provides a detailed description of Parts-of-speech tagging history and progress made during the last two decades in Arabic. Table 3.2 summarizes work on Arabic parts-of-speech tagging in these decades.

3.2.2.1 2000s

During this decade, the studies conducted on parts-of-speech tagging were mainly directed to Classical Arabic and MSA. This is justified by the phenomenon of diglossia of the Arabic language, which results in scares of written informal data, i.e., dialectal data. Moreover, researchers utilized supervised stochastic approaches mostly followed by rule-based and hybrid. Tagset selection is influenced by corpus annotation and analysis schemes. We noticed that some studies used fine grain tagsets while others used coarse-grain either as collapsed from fine grain tagsets or proposed ones. At the begging of this decade, researchers had to collect and annotate their data; however, later on, researchers made use of available data resources such as PATB, LDC, and others.

The reported results of the taggers show that the accuracy is usually in the high 90%. Such high accuracy suggests that applying parts-of-speech tagging to Arabic was successful, but improving obtained results is challenging.

Khoja (2001) developed a parts-of-speech tagger for MSA. She called it APT "automatic Arabic Parts of speech tagger". This tagger is considered the first tagger for Arabic (Abumalloh 2016). She also developed a tagset of 131 tags based initially on traditional Arabic grammar but then derived from the BNC English tagset (Abumalloh, 2016). The corpus used for training consists of 50,000 tokens. It was annotated using the initial small tagset. For testing, four other corpora were collected and used. In addition, Khoja adopted a hybrid method to build the tagger where she used both a lookup lexicon and a stemmer as initial rule-based tagging phase and then developed a statistical tagger based on the Viterbi algorithm. The accuracy of the tagging was reported as 86%.

Freeman (2001) tries to apply the Brill tagger to Arabic using a machine learning approach. He reports several challenges dealing with the Arabic Language. Some of these challenges are word ambiguity due to abandoning short vowels and scares of data resources, such as corpora and machine-readable lexicons. Hence freeman developed a MSA corpus that contains more than 3,000 tokens (Abumalloh et al. 2016). He also created a tagset of 146 tags, inspired by the English Brown corpus (Elhadj 2009).

The Stanford Arabic Parts of Speech tagger is introduced by the Stanford Natural Language processing group. **Toutanova et al. (2003)** developed the actual tagger applied to English using the supervised Maximum Entropy approach. Later the tagger was improved to support other languages, including Arabic. The Arabic model of the tagger is trained on the Penn Arabic Treebank (PATB) and uses an augmented Bies tagset of 25 tags for tagging (Alosaimy &

Atwell 2017). The tagger exploits the context of the proceeding and following context with a representation of the dependency network. The tagger inputs a segmented Arabic text using the Stanford Arabic Word Segmenter (Diab et al. 2013). The accuracy reported is 96.5% (El-haj & Koulali 2013)

Among the tasks done by **Diab et al. (2004)** is Parts of Speech tagging. They select a supervised, data-driven approach to perform parts-of-speech tagging on MSA text. They utilize the Support Vector Machines (SVMs) algorithm to process the text. Parts-of-speech tagging is accomplished using annotated part of the Arabic TreeBank. The 24 collapsed tagset in the Arabic Treebank distribution, known as The Reduced tagset, is used. They claim a 95.49% accuracy rate. This system was also tested on English using English TreeBank achieving 94.97% accuracy.

Habash and Rambow (2005) argue that using a morphological analyzer in parts-of-speech is the solution for morphologically rich languages such as Arabic. Thus, they use the morphological features classifiers of the morphological analyzer output to improve Parts of Speech tagging. Then Support Vector Machines (SVMs) are used for tagging. The corpus used for training and testing is from Penn Arabic Treebank. The size of the Data used is 120,000 tokens used for training and 12,000 tokens for development and testing. The tagset used was developed by them as a reduced tagset that consists of the following 15 tags: V (Verb), N (Noun), PN (Proper Noun), AJ (Adjective), AV (Adverb), PRO (Nominal Pronoun), P (Preposition/ Particle), D (Determiner), C (Conjunction), NEG (Negative particle), NUM (Number), AB (Abbreviation), IJ (Interjection), PX (Punctuation), and X (Unknown). The idea behind this system is to use linguistic features to aid choosing from the morphological analyzer output a precise tag of each token. For this purpose, Habash and Rambow (2005) use

ALMORGEANA morphological analyzer developed by (Habash 2007). This morphological analyzer reflects the output informs of lexeme and feature format rather than stem and affix format. for the task of tagging, the accuracy rate reported is 97.6% using the Penn Treebank tagset and 98.1% using their simplified tagset.

Duh and Kirchhoff (2005) present a minimally supervised HMM-based parts-of-speech tagger for Egyptian Colloquial Arabic. They exploit existing resources, which are "Call Home" Egyptian Colloquial Arabic (ECA) corpus, the LDC Levantine Arabic (LCA) corpus, the LDC MSA Treebank corpus, and the LDC-distributed Buckwalter stemmer for MSA. Moreover, a unified tagset consisting of 17 tags was used to collapse the fine-grain tagsets used in the earlier mentioned resources. A baseline tagger is initially developed utilizing a trigram HMM. The baseline tagger is later improved by adding affix features, leading to the tagging of out of vocabulary (OOV) words and constraining the Lexicon. Thus, the baseline accuracy is improved from 62.76% to 69.83%.

Al Shamsi and Guessoum (2006) developed an Arabic parts-of-speech tagger using Hidden Markov Model (HMM). They choose a fine grain tagset to incorporate more morphosyntactic information as their tagger aims to perform named entity extraction. Linguistically, they used the Arabic phrase structure to disambiguate parts of speech of the text. As the text needed pre-processing, they developed a tokenizer to separate punctuation marks from the text. They adopted the Buckwalter stemmer for stemming (Buckwalter 2002) and corrected the result manually. The built HMM tagger has unigram, bigram, and trigram language models and uses lexical and contextual probabilities. The tagset used consists of 55 tags and a 9.15 MB training corpus of MSA. The claimed accuracy of this tagger is 97%. They also reported a decrease of accuracy to 55% when non stemmed text is used.

Tlili-Guiassa (2006) developed a hybrid tagger integrating rule-based tagging and memory-based learning. The rule-based part is applied to predict the tag of each word, while MBL is used to verify each tag and correct any tagging errors. The accuracy rate is reported as 86% (Abumalloh et al. 2016). APT tagset developed by (Khoja et al. 2001) is modified and extended to suit the purpose intended. The method used is justified to tackle the challenges of variation and typographic errors, which directly influence tokenization and reduce tagging accuracy.

Zribi et al. (2007) Proposed parts-of-speech tagger for Arabic Vocalized text developing what they call "a multi-agent system of tagging." This system used the combined approach to integrate five taggers into one. The purpose of their multi-agent architecture is to improve accuracy. They developed a fine grain tagset consisting of 465 complex tags, which they called hyper-tags. The hyper-tags have a reduced tagset of 223 tags for inflected form and 65 for enclitics. The reduced tagset is known as micro-tags. They used a morphological analyzer to run the input through it, and they also developed a training corpus for a supervised tagging technique. They also used statistical tagging methods for their first four built taggers: HMM tagger, Unigram Tagger, bigram tagger, and trigram tagger. The fifth tagger is built based on "Sentence Pattern Based Agent". The idea behind the fifth tagger suggests providing a model of each sentence consisting of the valid tag of each word. Thus, the tagger tags each word in a sentence by checking the morphological analyzer output and the training sentences' models. The results claimed improvement of tagging accuracy to 98 % using micro- tags and 96 % using hyper-tags.

Alqrainy et al., (2008) develop a pattern (wazn) based algorithm to tag fully and partially-vocalized Arabic text. The developed algorithm utilizes a lexicon of each token's

possible tags (lexical information). It starts by providing the same length patterns of the processed token, and then it reduces the number of patterns by checking the similarity of characters. After that, it chooses the most similar pattern and mirrors it, adding the affixes to the pattern and storing it as a new pattern. Finally, the algorithm assigns the correct tag from the lexicon, corresponding to the chosen pattern. Alqrainy et al. tested 5000 semi-vocalized Arabic verbs and nouns, reporting a 91% accuracy rate.

Elhadj et al. (2009) introduces an HMM parts-of-speech tagger for classical Arabic, i.e., the language of Holy Quran based on Arabic sentence structure. The main aim of the tagger is to be used for tagging textual corpus of Holly Quran built by (Elhadj et al., 2009). The tagger is built using both linguistic and statistical processing. Each processing is applied in one of two levels. The first level is linguistic processing, where text is first normalized, tokenized into words, and morphologically analysed, segmenting words to their composing prefixes, stems, and suffixes. This level output serves as the input to the next level. The importance of the first-level processing lies in the idea of reducing the size of the tags needed.

On the other hand, the second level utilizes the Arabic sentence structure to operate the statistical model built to identify the morphological properties of words. Arabic sentence structure defines the permissible sequence of words providing them with appropriate tags. HMM reflects the sentence structure where HMM states represent a possible tag, and the sentence syntax controls the transitions between tags. Elhadj et al. also developed a tagset based on a hierarchical analysis of Arabic parts of speech so that it is possible to expand whenever it is needed. The tagset consists of 13 tags. In addition, a classical Arabic corpus is collected from books of the third century. It consists of 56312 tokens, out of which 6439 are types. The accuracy rate reported is 96%.

Algahtani et al (2009) utilize transformational based learning to perform Parts-of-speech tagging on Modern Standard Arabic text. Their work is an implementation of the Brill tagger (Brill, 1994) on segment level MSA text. the stem-affix segmentation is then used for tagging. Affixes are used as cues for tagging while tokens free of affixes are tagged from the lexicon. Unknown words on the other hand are tagged using Buckwalter's morphological analyser (BAMA) (Buckwalter, 2002) and then a bigram module is created to decide on the most probable tag. Finally, the remaining of the unknown tokens are tagged as NNP (proper noun). For training, Algahtani et al (2009) used the Arabic Tree Bank (ATB) corpora along with the collapsed tagset. The original size of the ATB corpora is 770k token but after segmentation it calculates as 920k. Algahtani et al (2009) reported an accuracy rate of 96.9% using ATB1 and 96.1% using the whole ATB.

Habash et al. (2009) present MADA+TOKEN as a toolkit for Arabic processing among which Parts-of-speech tagging is included. It is a Perl based system that utilizes third party software tools which are namely SVM Tool, SRI's Language Modelling Toolkit and LDC's Standard Arabic Morphological Analyzer (SAMA) (Habash et al. 2012a). The data used for training and testing of SVM model of MADA is taken from the Penn Arabic Tree Bank (PATB). To choose from the BAMA analysis, the system checks for 19 features out of which 14 are morphological. Based on these features the list of the provided analysis is ranked. Habash et al. (2009) developed MADA tagset which contains 34 tags. The accuracy reported for Parts-of-speech tagging is 96%+.

Albared et al. (2010) propose a Bigram Hidden Markov Model (HMM) parts-of-speech tagger for Both Classical Arabic and Modern standard Arabic (MSA). They used small amount of data to train and test the HMM based tagging tool. The size of the training corpus was 26631

tokens divided into 23146 tokens for training and 3485 tokens for testing. Moreover, the corpus contains Classical Arabic text as well as MSA text. Albared et al. develop a tagset inspired by Arabic TreeBank Parts of Speech guidelines (Maamouri et al. 2009). It consists of twenty-three tags. In addition, the proposed tagger makes use of several smoothing techniques to handle sparseness problem which are namely, Laplace estimation, Kneser-Ney smoothing and Modified Kneser Ney Smoothing. To choose the most probable tags the Viterbi algorithm is implemented. Moreover, a successive abstraction scheme is used to handle unknown words where lexical probabilities of prefixes and suffixes is calculated and used. Though this paper describes the preliminary results of the proposed tagger, they report 95.8% accuracy rate.

3.2.2.2 2010s

In this decade, besides MSA and CA, researchers targeted informal Arabic dialects. The advancement in technology and the use of social media in communication encouraged Arabic speakers to use their dialect in written communication. So dialectal data could be collected and analysed . Improving tagging accuracy is challenging, so researchers on MSA parts-of-speech tagging tried to improve tagging using different innovative methods such as combining taggers. In addition, the selection of tagging approaches heads toward artificial intelligence and machine learning. In tagset selection, preference was given to coarse tagsets more than fine grain.

In this subsection, researchers' influential contribution during the 2010s is presented. The following studies showcase the progress made during this time duration.

Köprü (2011) developed an HMM parts-of-speech tagger for Arabic. The tagger is built without a morph analyzer or a lexicon. The tagger is data-driven and language-independent, allowing it to be used for other languages. The corpus used for training is PATB developed by (Maamouri et al. 2004), and the tagset used is a coarse one consisting of 17 tags. As a result, an

accuracy of 95.57% is reported. The system was also tested on other languages reflecting similar accuracy levels. What is unique about this tagger is that it is language and tagset independent.

Alabbas and Ramsay (2012), inspired by the tagger combination technique, which is applied in several languages, attempted to improve Arabic parts-of-speech tagging accuracy by integrating three Arabic tagging systems. These systems are AMIRA 2.0 (Diab 2009), MADA 3.1 (Habash 2010), and MXL (Ramsay & Sabtan 2009). The corpus they used is Penn Arabic Treebank (PATB) (Maamouri & Bies 2004), which they considered a gold-standard Arabic corpus. The PATB fine-grained tagset was used, and another collapsed tagset. i.e., a coarse-grained tagset of the fine-grained, which consists of 39 tags. They dealt with each tagger's unique tagset using Transformational-based retagging (TBR) to improve tagging accuracy. In the TBR, an extra template is included. It checks the first and last three characters of words and other templates that check affixes to fit the Arabic language properly. Besides, they tried evaluating several integration techniques modifying some to fit the morpho-syntactic nature of Arabic. In addition to Transformational-based retagging (TBR), they used back-off strategies. Their result reported an improved accuracy that reached 99.5% using the coarse-grained tagset rather than the fine-grained one.

Al-Sabbagh and Girju (2012) built a supervised Transformational based Parts of Speech tagger targeting dialectal Arabic, namely Egyptian Arabic, collected from the tweeter platform. The annotation scheme was a function based on the grammatical function of words rather than morpho-syntactic features. Al-Sabbagh and Girju (2012) modified the Buckwalter tagset producing 49 tags. These tags are used in annotation as single and complex tags. Their social media corpus consists of 423,691 tokens and 70,163 types. Al-Sabbagh and Girju (2012) claim that grammatical function is more reliable than the morpho-syntactic scheme for tokenization

and parts-of-speech annotation purposes. The evaluation was performed on tokenization and tagging individually and collectively. The accuracy reported was in the 80s.

Ali and Jarray (2013) use the genetic algorithm to develop a parts-of-speech tagger for MSA. Since they use a supervised approach, they extracted a training corpus from EASC[8] and Watan[9] Corpora and annotated it manually. The authors report using a reduced tagset that consists of 22 tags (without the punctuation tag) to aid the tagger's work. Ali and Jarray explain that context size and training corpus size are highly influential factors. The accuracy of the tagger is 94.5% (Othmane et al., 2017).

Hadni et al. (2013) introduce a hybrid parts-of-speech tagger for Arabic, i.e., HMM integrated with Rule-based tagging. They follow the rule-based method introduced by (Taani & Abu-Al-Rub 2009). Taani's and Abu-Al-Rub's rule-based method consists of three steps: a lexicon, a morphological analyzer, and a syntax analyzer. After processing the text through these three steps, if a word is misclassified or unclassified, it heads toward the HMM analyzer for disambiguation. Hadni et al. used KALIMAT[10] Corpus (of MSA) and the Quranic Arabic[11] (of Classical Arabic) Corpus for training and testing. However, the tagset used consists only of three tags (Noun, Verb, and Particle). The reported accuracy rates are 98% for the KALIMAT corpus and 94.4% for the Quranic Arabic Corpus.

Muaidi (2014) applies a Levenberg-Marquardt neural network (LMNN) to parts-of-speech tagging of MSA. He claims that LMNN is better than the traditional back-propagation neural network (BPNN) as it is more efficient and effective. A corpus of 24,810 tokens is

[8] M. El-Haj, "Easc corpus."2013 [Online]. Available: http://privatewww.essex.ac.uk/ melhaj/form.htm
[9] Watan, "Watan 2004 corpus," 2004.[Online]. Available: http://sourceforge.net/projects/arabiccorpus/files/watan-2004corpus/
[10] http://bit.ly/16jO3Ks
[11] Quranic Arabic Corpus: http://corpus.quran.com

collected and manually annotated using the ARBTAGS tagset developed by (Alqrainy and Ayesh 2006). The tagset consists of 161 detailed tags and 28 general tags. The reported accuracy rates are 98.83 % on the training set and 90.21% on the test set.

Albogamy and Ramsay (2015) evaluate three well-known Arabic taggers, namely AMIRA (Diab 2009), MADA (Habash et al. 2009), and Stanford Log-linear (Toutanova et al. 2003). These taggers were applied to Arabic tweets. They implement some improvements based on detailed error analysis. The improvements suggested are done as pre-and post-processing steps. The pre-and post-processing steps are applied by using normalization and external knowledge. Albogamy and Ramsay collect a corpus from tweeter for training which consists of 390 tweets (5454 words). They also proposed a tagset to map the different tags of the used taggers to their unified tagset. Generally speaking, the accuracy reported prior to improvements is (49-65%) and (96-97%) posterior to the improvements of the three selected taggers.

Hamdi et al. (2015) proposed developing a tagger for Tunisian dialect using MSA resources to handle the lack of dialectal resources. They processed the text by converting Tunisian text into pseudo-MSA via three steps. Firstly, Tunisian words are morphologically analysed, then lexically transferred, and finally Generated as MSA forms. In addition, they use the following MSA resources: MAGEAD morphological analyzer and generator (Habash & Rambow 2006) as well as three lexica which are a lexicon of verbs, a lexicon of deverbal nouns, and a lexicon of particles. After the conversion, the tagger is used to disambiguate the parts of speech of each token. The tagger is based on a trigram HMM. It is trained on the Penn Arabic Treebank (PATB) Part 3 (Maamouri et al. 2004) using the Columbia Arabic Treebank (CATiB) tagset, which consists of only six tags (Habash and Roth 2009). The accuracy reported is 89%

Btoush et al. (2016) developed a rule-based parts-of-speech tagger for MSA. The tagging process goes through two phases. After the text is split into tokens, the first phase is the lexicon phase, where a token is looked up in the lexicon. A tag is outputted if there is a match; otherwise, the token heads to the second phase. In the second phase, morphological information, i.e., affixes, is used to decide the appropriate tag guided by the written rules. Finally, if there is no cue matching, the token is tagged as unknown. The tagset used consists of three tags only: verb, noun, and determiner (V, N, and DET).

Abumalloh et al. (2018) applied the artificial neural network (ANN) method to their MSA Tagger. They proposed a MSA grammar-based tagset, which mainly consists of 18 tags. Each tag consists of three letters. The first represents the main parts of speech, i.e., noun, verb, or particle, the second refers to the subclass, and the third represents the gender of each token (feminine/masculine). The tagger was trained using the backpropagation training algorithm, where a dataset consisting of 20,620 tokens was used for both training and testing. The tagger accuracy at the time of testing reached 89.04%.

AlKhwiter and Al-Twairesh (2020) developed a parts-of-speech tagger for tagging Arabic tweets. The taggers were built using Conditional Random Fields (CRF) and bidirectional –Long Short-Term Memory (BI-LSTM). Arabic tweets collected are written in Gulf Arabic (dialectal Arabic) and MSA. While preparing the corpus, Arabic dialects other than Gulf Arabic were illuminated in the pre-processing stage. AlKhwiter and Al-Twairesh used MADARi (Obeid et al. 2018), a morphological annotation tool and spelling corrector, and MADAMIRA (Pasha et al., 2014), which is a morphological analyzer. The data is manually annotated following the guidelines proposed by (Habash et al. 2012a, 2018). A tagset consisting of 44 tags was proposed for the tagging task. The tagset contains unique tags to capture tweeter text properties such as (#)

hashtag, (RT) retweet, etc. In addition, a hashtag behaviour analysis is conducted, which is claimed to influence the tagging task. After datasets annotation, three datasets were produced. They are namely the 'mixed', 'MSA' and 'GLF' with 3000, 1000, and 1000 tweets, respectively. After running the datasets through the proposed taggers, the BI-LSTM tagger achieves a higher accuracy rate. The accuracy rate reported is 96.5% for the mixed dataset. Table 3.2 summarizes the history of Parts-of-speech tagging in Arabic during the last two decades (2000-2020).

As observed in the Arabic parts-of-speech tagging literature, few taggers and annotation tools were developed to target Arabic dialects. Actually, in Arabic Natural Language processing, attention has been given to classical Arabic and MSA. It was clearly shown in this section, and, as far as our knowledge, there is no parts-of-speech tagger for San'ani Arabic dialect. Thus, the present study attempts to address the literature gap by providing an automatic machine learning tagging tool based on an innovative deep learning model for Parts-of-speech tagging of San'ani Arabic.

Table 3.2 *Summary of the work done on Arabic parts-of-speech tagging during the last two decades (2000-2020)*

Author/s	year	Approach/method	Language	Accuracy	Tagset & corpus
Khoja	2001	hybrid	MSA	86%	Fine grain tagset consists of 131 tags
Freeman	2001	TBL	MSA	_____	Fine grain tagset consists of 146 tags
Toutanova et al.	2003	Maximum Entropy approach	MSA	96.5%	the PATB corpus Bies(RTS) tagset of 24 tags
Diab et al.	2004	a supervised machine learning perspective using SVMs	MSA	95.49%	Part of the Arabic Treebank Bies (RTS) of 24 tags

Habash & Rambow	2005	Morphological analyser (morphological features classifiers) + SVM	MSA	97.6%	Corpus is part of the PATB (120,000training & 12,000 testing) tokens
Duh & Kirchhoff	2005	a minimally supervised approach- HMM	Egyptian Colloquial Arabic	69.83%	Tagset is Collapsed of the Bies of 17 tags ECA+ LCA+ LDC MSA Treebank corpus
Al Shamsi & Guessoum	2006	HMM	MSA	97%	The tagset used consists of 55 tags 9.15 MB training corpus of MSA
Tlili-Guiassa	2006	Rule-based and a Memory-based learning	MSA	86%.	APT tagset modified
Zribi et al.	2007	Combined approach Taggers combination approach	Vocalized MSA	-	Developed one fine grain tagset of A- micro-tags counting 223 tags for inflected forms and 65 tags for enclitics. B- hyper-tags 465 well-formed complex tags
Alqrainy et al.	2008	Rule based	MSA	91%	- -
El Hadj et al.	2009	Morphological analyser and HMM	Classical Arabic	96%	A built in tagset using traditional Arabic grammar in hierarchical classification The Corpus consists of 56,312 tokens Tagset used consists of 13 tags

AlGahtani et al.	2009	TBL	MSA	96.9%	ATB corpus of 770k – ATB (Bies)collapsed tagset
Habash et al.	2009	SVMs	MSA	96% +	The PATB Corpus A Tagset of 34 tags
Albared et al.	2010	Bigram HMM	Classical Arabic & MSA	95.8%	Tagset of 23 tags inspired by the ATB parts-of-speech guidelines Training corpus of 26,631 tokens
Köprü	2011	HMM	MSA	95.57%	A Coarse tagset consists of 17 tags
Alabbas & Ramsay	2012	Taggers combination technique	MSA	99.5%	The PATB Corpus The PATB fine grained tagset and collapsed tagset of 39 tags
Al-Sabbagh & Girju	2012	TBL	Twitter-based Egyptian Arabic	86.5%	A corpus of 423,691 tokens and 70,163 types.
Ali & Jarray	2013	Genetic approach	MSA	94.5%.	Reduced tagset of 22 tags without the punctuation tags The Corpus extracted from EASC and Watan Corpora
Hadni et al.	2013	Hybrid (HMM+Rule based)	MSA and CA	98% MSA 94.4% CA corpus	The Holy Quran Corpus and Kalimat Corpus A tagset of 3tags (Noun,Verb & Particle)

Muaidi	2014	Levenberg-Marquardt learning neural network	MSA	98.83 % on the training set 90.21% on the test set	Corpora of 24,810 that is collected and manually tagged ARBTAGS of 161 detailed tags and 28 general tags
Albogamy& Ramsay	2015	Pre-& Post-processing to existing Arabic taggers	Arabic tweets	96-97%	The Corpus consists of 390 tweets (5,454 words)
Hamdi et al.	2015	HMM	Tunisian dialect	89%	CATiB tagset The PATB corpus (Part3)
Btoush et al.	2016	Rule based approach	MSA		Tagset of 3tags (N,V,DET)
Abumalloh et al.	2018	Neural Network Modelling	MSA	96.96%.	Developed a three layer tagset which consists of 18 tags.
AlKhwiter & Al-Twairesh	2020	CRF and Bi-LSTM	Arabic tweets	96.5% Mixed dataset	A tagset of 44 tags

3.3 Existing Arabic parts-of-speech Tagsets

This section provides a review of the most known Arabic tagsets. These tagsets varies in length with different degrees of granularity. They are developed for different projects and purposes and are used in number of applications. They are (1) Khoja's Arabic tagset, (2) Penn Arabic Treebank tagset full, (3) Reduced Buckwalter Tagsets (3.1) Bies, (3.2) Kulick, (3.3) Erts (4) ARBTAGS, (5) CATiB parts-of-speech tagset. (6) SALMA Tagset.

3.3.1 Khoja's Arabic tagset (2001)

This tagset is one of the earliest tagsets developed for Arabic. It was developed by Khoja et al. (2001) to be implemented in her APT tagger (An Automatic Arabic Part-of-Speech Tagger). This tagset is a functional tagset based on the Arabic Traditional grammar theory instead of modern European EAGLES standards. Khoja argued that the EAGLES guidelines are not suitable for the Arabic language as they are developed for Indo- European language while Arabic belongs to the Semitic language family. Thus, using EAGLES standards will not cover some Arabic Morphosyntactic features such as the dual number.

This tagset consists of 177 tags which contain 103 types of nouns, 57 verbs, 9 particles, 7 residuals, and 1 punctuation. Khoja tagset includes the morphological features of gender, number, person, case, definiteness, and mood (Sawalha & Atwell, 2013). The tags are constructed by sequencing markers' tags together. For example, *kitabin'* book' is tagged as NCSgMGI, which stands for *Singular Masculine Genitive. Indefinite Common Noun*.

Khoja tagset is criticized on the basis of coverage. Though it is a fine grain tagset denoting morphological features, it lacks some classes and attributes, such as missing case marking of proper nouns and pronouns. Moreover, some morphological features are assigned faulty to some classes. For instance, some nouns are mistakenly given a person attribute though it is a verb-related attribute. e.g., the word *kitab* 'book' has no person attribute, but the verb *kataba* 'he wrote' has a second person singular feature. So, nouns are not to be treated as verbs.

Table 3.3 displays the tagset. In general, Khoja's tagset denotes morphological features and the syntactic classes, which is valid for morphological analyzers' tagsets rather than parts-of-speech taggers.

Table 3.3 ***The Khoja tagset***

• **N** *noun*
– **+C** *common*+ **Attribute:** *number-gender-case-definiteness*
– **+P** *proper*
– **+Pr** *pronoun*
∗**+P** *personal* + **Attribute:** *number-person-gender*
∗**+R** *relative*
· **+S** *specific* + **Attribute:** *number-gender*
· **+C** *common*
∗**+D** *demonstrative* + **Attribute:** *number-gender*
– **+Nu** *numerical*
∗**+Ca** *cardinal* + **Attribute:** *[Sg]-gender*
∗**+O** *ordinal* + **Attribute:** *[Sg]-gender*
∗**+Na** *numerical adjective* + **Attribute:** *[Sg]-gender*
– **+A** *adjective*+ **Attribute:** *number-gender-case-definiteness*
• **V** *verb*
– **+P** *perfective*+ **Attribute:** *number-person-gender*
– **+I** *imperfective*+ **Attribute:** *number-person-gender-mood*
– **+Iv** *imperative* + **Attribute:** *number-[2]-gender*
• **P** *particle*
– **+Pr** *preposition*, **+A** *adverbial*, **+C** *conjunction*, **+I** *interjection*, **+E** *exception*, **+N** *negative*, **+A** *answers*, **+X** *explanations*, **+S** *subordinates*
• **R** residual
– **+F** *foreign*, **+M** *mathematical*, **+N** *number*, **+D** *day of the week*, **+my** *month of the year*, **+A** *abbreviation*, **+O** *other*
• **PU** *punctuation*
• *Attributes*
– Gender: M *masculine*, F *feminine*, N *neuter*
– Number: Sg *singular*, Pl *plural*, Du *dual*
– Person: 1 *first*, 2 *second*, 3 *third*
– Case: N *nominative*, A *accusative*, G *genitive*
– Definiteness: D *definite*, I *indefinite*
– Mood: I *indicative*, S *subjunctive*, J *jussive*

Source:5. Habash, Nizar Y. 2010. Introduction to Arabic natural language processing. Synthesis Lectures on Human Language Technologies, 3(1), 85.

3.3.2 Penn Arabic Treebank (PATB) tagset (full) (2002)

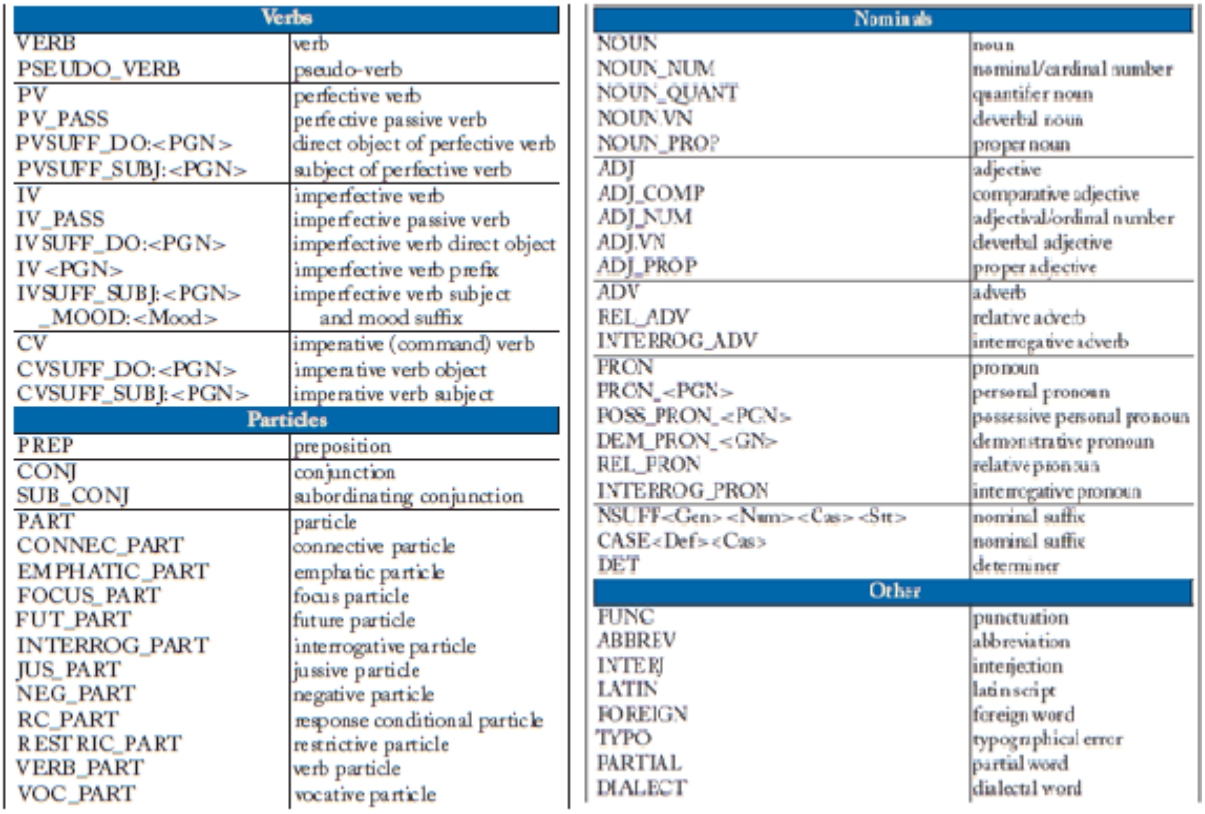
PATB was developed in 2002 by Tim Buckwalter. It is also called the Buckwalter tagset. It is used to annotate the Penn Arabic Treebank (PATB). First, the Buckwalter Arabic Morphological Analyzer (BAMA) was used to analyze the PATB morphologically. Then Arabic

linguists select each word's most appropriate parts-of-speech tag within the context (Sawalha & Atwell 2013).

This tagset is a form-based tagset rather than a function-based. So, it can be used for tokenized and untokenized text. The tokenized tags are used for the annotation of the Penn Arabic Treebank. The size of the tokenized tags set is around 500+ (Habash 2010). However, the size of the untokenized might reach over 2000 tag types (Diab 2007). The number of the morpheme tags in PATB is 135. The morphological features represented in this tagset are case, gender, number, definiteness, mood, person, voice, tense, and aspect. Figure 3.3 presents Buckwalter tagset components, consisting of mainly 70 or so sub-tag symbols (Habash 2010).

Buckwalter tagset is problematic mainly because of its huge size. Such size is not recommended for computational applications as it can affect the accuracy negatively. Many reduced forms of this tagset are proposed to be managed (Diab 2007). In addition, there are several issues with the tagset grammatical analysis, such as the lack of distinction between clitics, inflection suffixes, and attached pronouns (Qassem 2015). Moreover, the representation of the morphological attributes makes this tagset more suitable for morphological analyses than grammatical or syntactic analysis. Figure[12] 3.3 shows these components.

[12]7Figure 2.1 contains some parameters which are define here:
<PGN>*person-gender-number*, <GN>*gender-number*,
person: 1 *first*, 2 *second*, 3 *third*, *φ unspecified*
gender: M *masculine*, F *feminine*, *φ unspecified*
number: S *singular* D *dual* P *plural* 0 unspecified
<Mood>: I *indicative*, S *subjunctive*, J *jussive*, SJ *subjective/jussive*
<Gen>: _MASC *masculine*, _FEM *feminine*
<Num>: _SG *singular*, _DU *dual*, _PL *plural*
<Cas>: _NOM *nominative*, _ACC *accusative*, _GEN *genitive*, _ACCGEN *accusative/genitive*, *φ unspecified*
<Stt>: _POSS *construct/possessor*, *φ not construct*
<Def>: _DEF *definite*, _INDEF *indefinite*



Verbs	
VERB	verb
PSEUDO_VERB	pseudo-verb
PV	perfective verb
PV_PASS	perfective passive verb
PVSUFF_DO:<PGN>	direct object of perfective verb
PVSUFF_SUBJ:<PGN>	subject of perfective verb
IV	imperfective verb
IV_PASS	imperfective passive verb
IVSUFF_DO:<PGN>	imperfective verb direct object
IV<PGN>	imperfective verb prefix
IVSUFF_SUBJ:<PGN>_MOOD:<Mood>	imperfective verb subject and mood suffix
CV	imperative (command) verb
CVSUFF_DO:<PGN>	imperative verb object
CVSUFF_SUBJ:<PGN>	imperative verb subject
Particles	
PREP	preposition
CONJ	conjunction
SUB_CONJ	subordinating conjunction
PART	particle
CONNEC_PART	connective particle
EMPHATIC_PART	emphatic particle
FOCUS_PART	focus particle
FUT_PART	future particle
INTERROG_PART	interrogative particle
JUS_PART	jussive particle
NEG_PART	negative particle
RC_PART	response conditional particle
RESTRIC_PART	restrictive particle
VERB_PART	verb particle
VOC_PART	vocative particle

Nominals	
NOUN	noun
NOUN_NUM	nominal/cardinal number
NOUN_QUANT	quantifier noun
NOUN_VN	deverbal noun
NOUN_PROP	proper noun
ADJ	adjective
ADJ_COMP	comparative adjective
ADJ_NUM	adjectival/ordinal number
ADJ_VN	deverbal adjective
ADJ_PROP	proper adjective
ADV	adverb
REL_ADV	relative adverb
INTERROG_ADV	interrogative adverb
PRON	pronoun
PRON_<PGN>	personal pronoun
POSS_PRON_<PGN>	possessive personal pronoun
DEM_PRON_<GN>	demonstrative pronoun
REL_PRON	relative pronoun
INTERROG_PRON	interrogative pronoun
NSUFF<Gen><Num><Cas><Stt>	nominal suffix
CASE<Def><Cas>	nominal suffix
DET	determiner
Other	
PUNC	punctuation
ABBREV	abbreviation
INTERJ	interjection
LATIN	latin script
FOREIGN	foreign word
TYPO	typographical error
PARTIAL	partial word
DIALECT	dialectal word

Figure 3.3 *The Buckwalter tagset components*

Source: Habash, Nizar Y. (2010). Introduction to Arabic natural language processing. Synthesis Lectures on Human Language Technologies, 3(1), p. 81.

3.3.3 Reduced Buckwalter Tagsets: Bies, Kulick, and ERTS.

These tagsets are collapsed from the main Buckwalter tagset that is discussed above. The reason behind the development of these tagset is the criticism of the Full Buckwalter tagset, which is rich in computational problems and hard to manage computationally (Habash 2010). The following sub-section introduced the three reduced tagsets of the full PATB tagset.

3.3.3.1 Bies (2004)

Ann Bies and Dan Bikel developed the Bies tagset to improve the performance of Arabic parsing (Sawalha and Atwell 2013). it is a reduced form of the PATB tagset. It is also known as

the Reduced Tagset (RTS). RTS is inspired by the Penn English Treebank parts-of-speech tagset (Habash 2010). This tagset consists of 24 tags. It is a linguistically coarse tagset that researchers have widely used. It applies the following morphological features: case, mood, gender, person, and definiteness (Diab 2007). a list of the RTS tags is provided by Habash (2010). Table 3.4 provides the RTS tagset.

Table 3.4 ***The Reduced Tagset (RTS)***

NOMINALS	
Nouns	
NN	singular common noun or abbreviation
NNS	plural/dual common noun
NNP	singular proper noun
NNPS	plural/dual proper noun
Pronouns	
PRP	personal pronoun
PRP$	possessive personal pronoun
WP	relative pronoun
Other	
JJ	adjective
RB	adverb
WRB	relative adverb
CD	cardinal number
Fw	foreign word
PARTICLES	
CC	coordinating conjunction
DT	determiner/demonstrative pronoun
RP	particle
IN	preposition or subordinating conjunction
VERBS	
VBP	active imperfect verb
VBN	passive imperfect/perfect verb
VBD	active perfect verb
VB	imperative verb
OTHER	
UH	interjection
PUNC	punctuation
NUMERIC_COMMA	the letter ر **r** used as a comma
NO_FUNC	unanalysed word

3.3.3.2 Kulick (2006)

The Kulick tagset was named after its developer Seth Kulick. It was developed in 2006 to extend the RTS to benefit Arabic parsing. It consists of 43 tags. Habash (2010) listed the extensions made as shown in Table 3.5.

Table 3.5 ***The Kulick tagset extensions***

Punctuation Marks	
[,]	comma
[:]	colon
[.]	dot
[”]	quotation mark
-LRB-	left round bracket
-RRB-	right round bracket.
Nouns and Adjectives	
NOUN_QUANT	quantifier nouns
ADJ_COMP	comparative adjectives
ADJ_NUM	adjectival/ordinal numbers
DV	deverbals
Demonstratives and Definite article	
DEM	Demonstratives
DT	definite article
Definite article combination tags (examples)	
DT+NN	definite article and common noun
DT+ADJ_COMP	definite article and comparative adjective
DT+CD	definite article and cardinal number
DT+JJ	definite article and adjective

3.3.3.3 ERTS (2007)

In the Extended Reduced Tagset (ERTS), the RTS was extended, adding the explicit morphological markers such as number, gender, and definiteness on nominals only. It was developed by Mona Diab in 2007. The number of tags increased from 24 to 75 tags. Habash (2010) commented that the ERT is as accurate as RTS, but it benefits higher computational tasks providing explicit morphological information.

3.3.4 ARBTAGS (2006)

Alqrainy and Ayesh (2006) developed ARBTAGS following the footsteps of Shereen Khoja[13], In which they deviate from the EAGEL standards as it suits Indo-European Languages. Therefore, they built the tagset based on Traditional Arabic Grammar Theory as shown in the tagset hierarchy Figure 3.4. The tagset consists of 161 detailed tags divided into 101 nouns, 50 verbs, 9 particles, and 1 punctuation mark. In addition, the developers identified 28 general tags, which are displayed in Figure 3.5. The morphological features included are gender, number, case, mood, person and state[14]. Alqrainy (2008) implemented this tagset in his tagger called Arabic Morphosyntactic Tagger (AMT).

[13] For details on Khoja's work, check section 2.2.1

[14]To check the full tagset check AlqrainyS hihadeh; and Aladdin Ayesh. 2006. Developing a tagset for automated Parts-of-speech tagging in Arabic. WSEAS transactions on computers 5.no. 11, pp. 5-6.

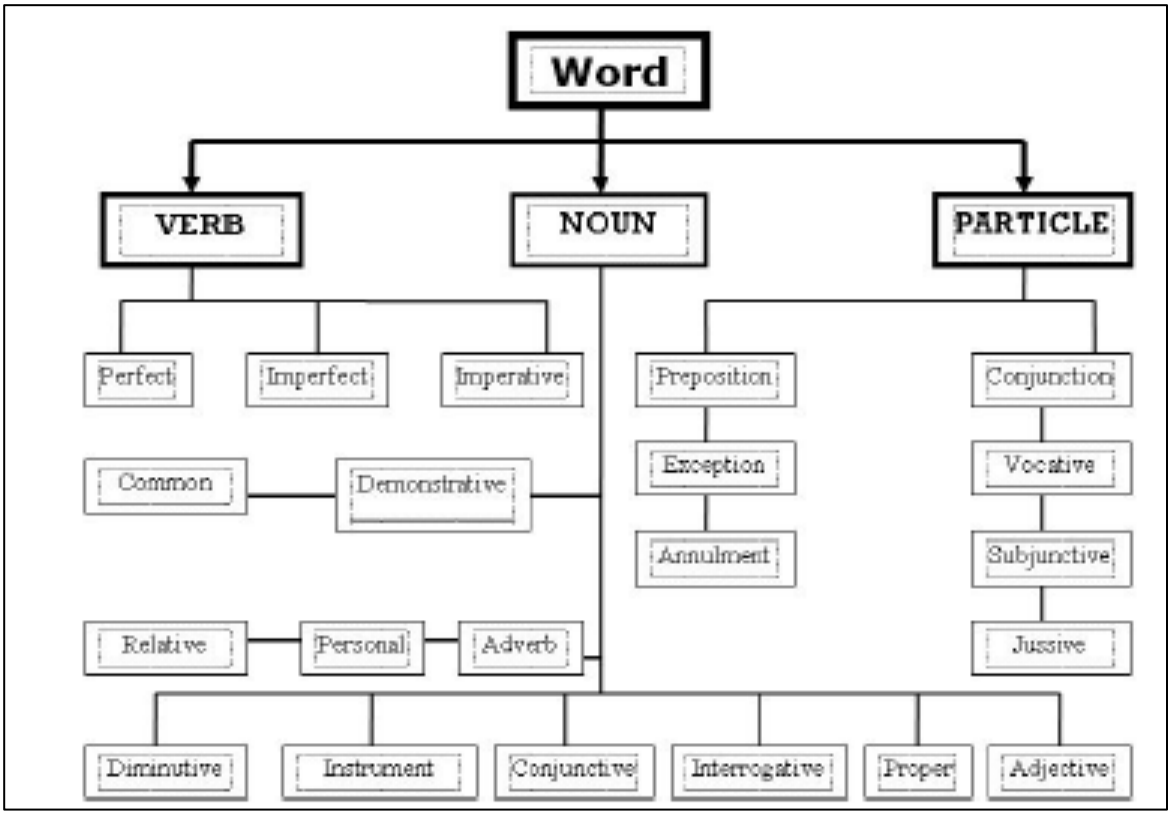


Figure 3.4 ***The ARBTAGS Tagset hierarchy***

Source: AlqrainyShihadeh; and Aladdin Ayesh. 2006. Developing a tagset for automated Parts-of-speech tagging in Arabic. WSEAS transactions on computers, 5(11): 4.

Tag	Description	Tag	Description
VePe	*Perfect verb*	NuCd	*Conditional noun*
VePi	*Imperfect verb*	NuDe	*Demonstrative noun*
VePm	*Imperative verb*	NuIn	*Interrogative noun*
NuPo	*Proper noun*	NuAd	*Adverb*
NuCn	*Common noun*	NuNn	*Numeral noun*
NuAj	*Adjective noun*	Fw	*Foreign noun*
NuIf	*Infinitive noun*	Pun	*Punctuation mark*
NuRe	*Relative noun*	PrPp	*Preposition*
NuDm	*Diminutive noun*	PrVo	*Vocative Particle*
NuIs	*Instrument noun*	PrCo	*Conjunction Particle*
NuPn	*Noun of Place*	PrEx	*Exception Particle*
NuTn	*Noun of Time*	PrAn	*Annulment Particle*
NuPs	*Pronoun*	PrSb	*Subjunctive Particle*
NuCv	*Conjunctive noun*	PrJs	*Jussive Particle*

Figure 3.5 ***General Tags of The ARBTAGS Tagset***

Source: Sawalha, Majdi; and Eric Atwell. 2013. A standard tagset expounding traditional morphological features for Arabic language parts-of-speech tagging." Word Structure 6, (1): 55.

3.3.5 CATiB part-Of-Speech Tagset (2009)

Nizar Habash and Ryan M. Roth developed the Columbia Arabic Treebank (CATiB) in 2009 for Columbia University (Habash & Roth 2009). The motivation behind developing this tagset is to minimize the time and effort spent on manual annotation through a small tagset that consists of only six tags. It is used for syntactic tagging and parsing. CATiB tags are listed in Table 3.6.

Table 3.6 *The CATiB tagset*

Tag	Description
VRB	all verbs including the class of *incomplete verbs*
VRB-PASS	passive-voice verbs
NOM	all nominals such as noun, adjective, adverb, active/passive participle, deverbal, noun pronoun (personal, relative, demonstrative, interrogative), numbers (including digits), and interjections
PROP	proper nouns
PRT	all particles
PNX	all punctuation marks

3.3.6 SALMA Tagset (2013)

SALMA tagset was developed by (Sawalha & Atwell 2013). It is a fine-grain tagset that follows the traditional Arabic grammar theory. It is described as a general-purpose tagset designed to encode morphological features of any word in detail. This tagset consists of 22 characters where each character represents a value or attitude, which refers to a morphological feature category. These 22 characters are arranged as the following:

Main Parts of Speech classes:

- Character number 1 refers to main parts of speech which are five, namely: noun, verb, particle, punctuation and residual.

Parts of Speech subclasses:

- Character number 2 represents subcategories of noun which are 34 subclasses.
- Character number 3 represents subclasses of verbs which are 3 subclasses.
- Character number 4 represents subclasses of particles which are 21 subclasses.

- Character number 5 represents subclasses of residuals.
- Character number 6 represents subclasses of punctuations.

Morphological features:

- Character number 7 represents gender.
- Character number 8 represents number.
- Character number 9 represents person.
- Character number 10 represents morphology.
- Character number 11 represents case & mood.
- Character number 12 represents case & mood markers.
- Character number 13 represents definiteness.
- Character number 14 represents voice
- Character number 15 represents emphasize.
- Character number 16 represents transitivity.
- Character number 17 represents humanness.
- Character number 18 represents variability & conjugation.

Morphological features related to Arabic text analysis:

- Character number 19 represents augmented and unaugmented.
- Character number 20 represents number of root letters
- Character number 21 represents verb internal structure.
- Character number 22 represents noun finals.

Sawalha and Atwell (2013) reported an upper limit to possible tag combinations to be "101,945,168 possible morphological feature combinations" (66). So, one hundred million

possible tag combinations are a huge unrealistic size. This tagset looks theoretical tagset more than a practical one. Moreover, some useless, redundant tags do not need to be included. It seems that the aim behind the SALMA tagset is to summarize all of the classifications of the Arabic language. Table 3.7 summarizes the investigated Arabic parts-of-speech tagsets.

To conclude, there is no standard parts-of-speech tagset for Arabic. Tagset are created based on the project in hand, the target variety, and linguistic theory adopted. Most of the available Arabic tagsets are built to represent MSA morpho-syntactically. It was pretty evident that including detailed morphological representation produced fine-grain tagsets of enormous size. Hence, these tagsets suit morphological analysis rather than parts-of-speech tagging. However, we found that the Bies/RTS is somewhat appropriate for performing parts-of-speech tagging in terms of granularity, size, and earlier application. Hence, we adapted the Bies/RTS tagset with certain modifications, explained and justified in Chapter Five.

Table 3.7 ***Summary of the Arabic Tagsets***

Tagset name	**author**	**year**	**size**	**Morphological features**
Khoja's Arabic tagset	Khoja et al.	2001	177 tags	Gender, Number, Case, Definiteness, Person, Mood
PATB tagset (full)	Buckwalter	2002	- Tokenized tagset size is over 500 - Untokenized tagset size is over 2000 tag types	case, gender, number, definiteness, mood, person, voice, tense, and aspect.

			- Morpheme tags size is 135	
Bies /RTS	Bies & Bikel	2004	24 tags	case, mood, gender, person, and definiteness
Kulick	Seth Kulick	2006	43 tags	case, mood, gender, person, and definiteness
ERTS	Mona Diab	2007	75 tags	+ Gender, Number, Definiteness on nominals
ARBTAGS	Alqrainy & Ayesh	2006	161 detailed tags and 28 general tags	gender, number, case, mood, person and state
CATiB	Habash & Roth	2009	6 tags	-
SALMA Tagset	Sawalha & Atwell	2013	22 characters over one hundred million possible tag combinations	gender, number, person, morphology, case & mood, case & mood markers, definiteness, voice, emphasize, transitivity, humanness and variability & conjugation

3.4 Review of Dialectal Arabic Corpora

The literature directed to Arabic dialects increases on each successive day over a long period after the bulk of significant works on the Arabic language was centered on MSA. However, research on Arabic dialects is still lagging far behind MSA in terms of data availability, coverage, or validity for machine use. This may be due to the paucity of data readily available for researchers as MSA is still predominant over dialectal Arabic informal

settings. However, with the advent of technology and the vast spread of social media networking sites, more individual-driven data becomes accessible and available.

A recent critical survey of the freely available Arabic Corpora was conducted by (Zaghouani 2017), where he listed about 66 free resources of Arabic Corpora. All these corpora exist in the form of 6 categories: i.e., 23 Raw Text Corpora (i.e., 11 Monolingual Corpora List; 4 Multilingual Corpora List; 2 Dialectal Corpora; and 6 Web-based Corpora List); 15 Annotated Corpora (i.e., 6 Named Entities Corpora List; 3 Errors Annotated Corpora List; and 6 Miscellaneous Annotated Corpora List); 16 Lexicon Corpora (i.e., 9 Lexical Databases List and 7 List of Words Lists); 1 Speech Corpora; 4 Handwriting Recognition Corpora and 7 Miscellaneous Corporatypes (e.g., Questions/Answers, comparable corpora, plagiarism detection, and summaries). As noted among this collection of texts, the focus can be summarized in terms of quantity, quality, coverage, and accessibility which are the criteria or the principal motives Arabic researchers opt for better resources. Out of this collection of texts, this survey mentioned only two dialectal corpora which exist in the form of raw text resources (i.e., Tunisian Dialect Corpus (Graja et al. 2010) and Arabic Multi Dialect Text Corpora (Almeman and Lee 2013). The Tunisian Dialect Corpus consists of 3,403 words that have been transcribed from spoken dialogues between staff and clients. At the same time, the Arabic Multi Dialect Text Corpora has a massive volume of about 2 million unique words gathered from 55K webpages obtained from main Arabic regional dialectal varieties (i.e., Gulf, Levantine, North Africa, Egypt).

Several other studies have been conducted on Arabic dialects. Most of them focus on preparing dialectal corpora for machine learning use and training and developing dialect-based NLP applications. These corpora either evolved as (1) raw texts dialectal corpora (Alshutayri

and Atwell 2018; SharafAddin and Al-Shehabi 2020); (2) annotated dialectal corpora (Zaghouani 2017; Almeman and Lee 2013; Al- Shargi et al. 2016; ZaghouaniandCharfi 2018; Khalifa et al. 2018; Al- Shargi et al. 2015); or (3) Parallel dialectal corpora (Bouamor et al. 2018; McEnery et al. 2006). Some studies focus on raw corpora (Alshutayri and Atwell 2018). Their work is a balanced multi-Arabic dialectal text corpus built using CMC and social media sources: Twitter, comments from online newspapers, and Facebook. Their corpus size is 13,876,504 word-tokens collected from five groups of Arabic dialects: Gulf, Iraqi, Egyptian, Levantine, and North African.

Several other studies were conducted on Arabic annotated corpora (category 2) to create standard reference resources that provide a stable base of linguistic analyses. These studies include (Al-Shargi et al. 2016; Jarrar et al. 2017; Khalifa et al. 2018; and Al Shargi et al. 2019) focused on morphological annotation. (Al-Shargi et al. 2016) presented new resources for two Arabic dialects: Moroccan and San'ani Yemeni Arabic. The corpus for each dialect was morphologically annotated using the DIWAN tool (Al-Shargi and Rambow 2015), which requires manual annotation. Their corpus size is 64K and 32.5K tokens for Morrocan and San'ani Yemeni Arabic. While (Jarrar et al. 2017) developed a corpus for Palestinian Arabic dialect called Curras. This corpus consists of 56,700 tokens and 16,416 types. Jarrar et al. annotated about 98.7 % tokens and (97.6 %) types that were valid. Each token was annotated morphologically with parts-of-speech (POS), stem, prefix, suffix, lemma, and gloss. They collected their corpus from Facebook, Twitter, Forums, Palestinian stories, Palestinian terms, and TV Shows. Khalifa et al. (2018) introduced another annotated large-scale resource for Emirati Arabic. It has a manual morphological annotation, tokenization, parts-of-speech, lemmatization, English glosses, and dialect identification. This corpus covers 200K words

chosen from eight Gumar corpus novels of Emirati Arabic. (Al-Shargi et al. 2019) presented a collection of morphologically annotated corpora for seven Arabic dialects: Taizi Yemeni, Sanaani Yemeni, Najdi, Jordanian, Syrian, Iraqi, and Moroccan Arabic. Their corpora collections cover 200,000 words provided with orthography, diacritized lemmas, tokenization, morphological units, and English glosses. The other type of dialectal corpora, on the other hand, used different annotations (Zaghouani and Charfi 2018). They presented a multi-dialectal corpus that covers 11 distinctive Arabic regional dialectal varieties spoken in 16 Arabic countries extracted from Twitter platforms, and they called it 'Arap-Tweet'. However, later on, they developed an improved version (version 2.0) with various improvements in terms of volume and quality of annotation (Charfi et al., 2019). The annotation adopted in these corpora was based on three criteria: Dialect, Age, and Gender.

The third corpora collections concentrated more on parallel dialectal corpora (Bouamor et al. 2018; Diab et al. 2014). (Bouamor et al. 2018) presented two resources: the MADAR Corpus (a parallel corpus) and MADAR Lexicon. In MADAR Corpus, they translated some selected sentences from the Basic Traveling Expression Corpus (BTEC) (Takezawaet al.2007) into Arabic multi-dialects covering about 25 cities. In contrast, MADAR Lexicon covers about 1,045 entries from the same cities. (Diab et al. 2014) on the other hand, presented a comprehensive 3-way large-scale parallel lexicon of English, MSA, and Egyptian Arabic with a deep linguistic annotation that includes parts-of-speech (POS), number, gender, rationality, and morphological root and pattern forms. This lexicon consists of about 73,000 Egyptian entries.

As our focus is on San'ani Yemeni Arabic, the only reported work on this dialect is done by (Al-Shargietal. 2016; Al-Shargietal. 2019). The first annotated corpus for the San'ani dialect was attempted by (Al-Shargietal. 2016), where a collection of 32.5K tokens was obtained from

both online and print materials. They covered as many genres as they could. This includes oral interviews, social texts, pearls of wisdom and tales, San'ani folktales, sermons, poems, humour, explanation, and politic text. They used the DIWAN tool, which assigns the following annotations for each word in the corpus: Diac, Lex, Bwhash, Gloss, Clitics, Other features (part of speech, gender, functional gender, formal number, and functional number.) The other study seems similar to (Al-Shargietal. 2016) conducted by the same authors and uses the same corpus size and tool (Al-Shargietal. 2019). However, this study includes two Yemeni dialects, San'ani and Taizi, along with other 5 Arabic dialects. Each word in the corpus was annotated with CODA, Lemma, Morph, Prefix, Stem, and Suffix to bridge a common ground with MSA and other Arabic dialects.

3.5 Summary

Chapter Three investigates the literature of parts-of-speech tagging through four main sections. The first section. 3.1 deals with the parts-of-speech tagging approaches and methods. It presents the classification of these methods discussing each method informatively.

On the other hand, the second section, 3.2, investigates the history of parts-of-speech tagging. The historical investigation is performed in non-Arabic languages as well as Arabic. The historical survey presents the prominent contributions in parts-of-speech tagging chronologically. The historical survey shows no parts-of-speech tagger for San'ani Arabic dialect. Thus, the present study attempts to address the literature gap by providing an automatic machine learning tagging tool based on an innovative deep learning model for parts-of-speech tagging of San'ani Arabic.

Section 3.3 surveys the Arabic parts-of-speech tagsets. The survey shows that there is no standard parts-of-speech tagset for Arabic.

Tagset are created based on the project in hand, the target variety, and linguistic theory adopted. Most of the available Arabic tagsets are built to represent MSA morpho-syntactically. However, the Bies/LDC tagset can be adapted to perform San'ani Arabic parts-of-speech tagging.

The available Arabic dialectal corpora survey conducted in section 3.4 clearly shows that there is not a reasonable size corpus of San'ani Arabic. The only San'ani Arabic corpus reported is (Al-Shargi et al. 2016), consisting of 33k. This corpus is small in size, and the major part of it is a transcription of spoken data rather than written text. Moreover, the corpus URL link is broken; hence it is ineffectual.

CHAPTER FOUR
DATA COLLECTION AND PRE-PROCESSING

4.1 Introduction

Work in the field of Arabic Natural language processing (NLP) is mainly directed to Modern Standard Arabic (MSA), which is the official written form in Arabic-speaking countries (Khalifa et al. 2016). Thus, most available resources are designed to benefit MSA ultimately, but they fail to serve dialectal Arabic. This issue is caused by the scarcity of dialectal data and linguistic divergence between MSA and dialectal Arabic (Darwish et al. 2021). Recently, Arabic dialects received growing attention as Arabic speakers started writing on social media platforms in their dialects. Therefore, NLP researchers and scholars target developing data and resources of Arabic dialects utilizing data from social media platforms and other online resources. However, some dialects received more attention than others. San'ani Arabic is one of the dialects that lack the availability of data resources.

Social media platforms have become an essential resource for acquiring Arabic dialect text as such text can be exploited in developing natural language processing tools and applications (Alshutayri & Atwell 2019; and Hegazi et al. 2021). However, social media text is characterized by having several messy, incomplete, and often frustrating data. These features make social media raw data useless unless being pre-processed. Pre-processing step facilitates text representation by making the input data more consistent and standardized. Nevertheless, pre-processing of social media Arabic text is still challenging for researchers and NLP tool developers (Hegazi et al. 2021). Moreover, the

nature of Arabic dialectal text makes the matter more complicated as Arabic dialects have no standard orthographies (Habash 2010; and Darwish et al. 2021).

This chapter describes the development and pre-processing of the social media-based corpus of San'ani Arabic. It consists of seven sections: 4.1 introduction, 4.2 corpus definition, 4.3 corpus development, 4.4 pre-processing, 4.5 corpus statistical analysis, 4.6 corpus genre, and 4.7 summary.

4.2 Corpus Definition

The term corpus is considered the centre of corpus studies. Several definitions of this term were proposed, some of which:

- "A collection of LINGUISTIC DATA, either written texts or a TRANSCRIPTION of recorded speech, which can be used as a starting point of linguistic description or as a means of verifying hypotheses about a language." (Crystal 2008, 117)
- " .. a corpus (pl. corpora) is a statistically sampled language database for the purpose of investigation, description, application and analysis relevant to all branches of linguistics." (Dash and Arulmozi 2018, 4)
- Cambridge English dictionary[15] define corpus as "a collection of written or spoken material stored on a computer and used to find out how language is used."

[15]dictionary.cambridge.org/dictionary/english, s.v. "corpus", accessed June 02,2021 https://dictionary.cambridge.org/dictionary/english/corpus

- Macmillan dictionary[16] provides the following definition: "a collection of written and spoken language stored on computer and used for language research and writing dictionaries."
- Merriam-Webster dictionary[17] describes corpus as "a collection or body of knowledge or evidence especially: a collection of recorded utterances used as a basis for the descriptive analysis of a language."

To sum up, a corpus can be defined as a sample of spoken or written data collected to represent a natural language/s for a specific purpose and based on pre-established criteria.

4.3 Corpus Development

This section describes the process of raw corpus collection from social media platforms. Generally speaking, dialectal Arabic written text is scarce compared to MSA, which is widely available as the medium of education, media, science, and news. Dialectal Arabic text is found in informal communication means such as blogs, social media platforms (Facebook, Twitter, Telegram, etc.). However, it practices a driven commentary on multiple domains covering the traditional folklore and literature (stories, plays, songs, and so on) (Jarrar et al. 2017).

Working on San'ani Arabic, a Yemeni Arabic Dialect spoken in northern Yemen as introduced in section 2.2.2.1. Chapter Two, is not an easy task. The main reason behind this is

[16]Macmillandictionary.com dictionary, s.v. "corpus," accessed June 02,2021https://www.macmillandictionary.com/dictionary/british/corpus
[17]Merriam-Webster.com Dictionary, s.v. "corpus," accessed June 02, 2021, https://www.merriam-webster.com/dictionary/corpus

resources limitation. In fact, the only sources available for San'ani Arabic are social media platforms. Moreover, social media data is prone to noise and orthographical inconsistencies. Therefore, data collection and pre-processing proved to be very difficult and time-consuming.

As shown in Figure 4.1, the process of our corpus development is divided into several successive stages and steps. The first stage is corpus selection, followed by corpus collection. Then an initial statistical calculation takes place. After collecting raw data, the pre-processing is applied using two techniques: data cleaning and normalization. After that data is processed, it is passed through the LancsBox[18] (2018) to reflect a final statistical analysis and to tokenize the data. In the following subsections a detailed description of different stages of corpus development is given.

[18] It is a software package developed at Lancaster University to analyze corpora and language data. It is freely available on: http://corpora.lancs.ac.uk/lancsbox/

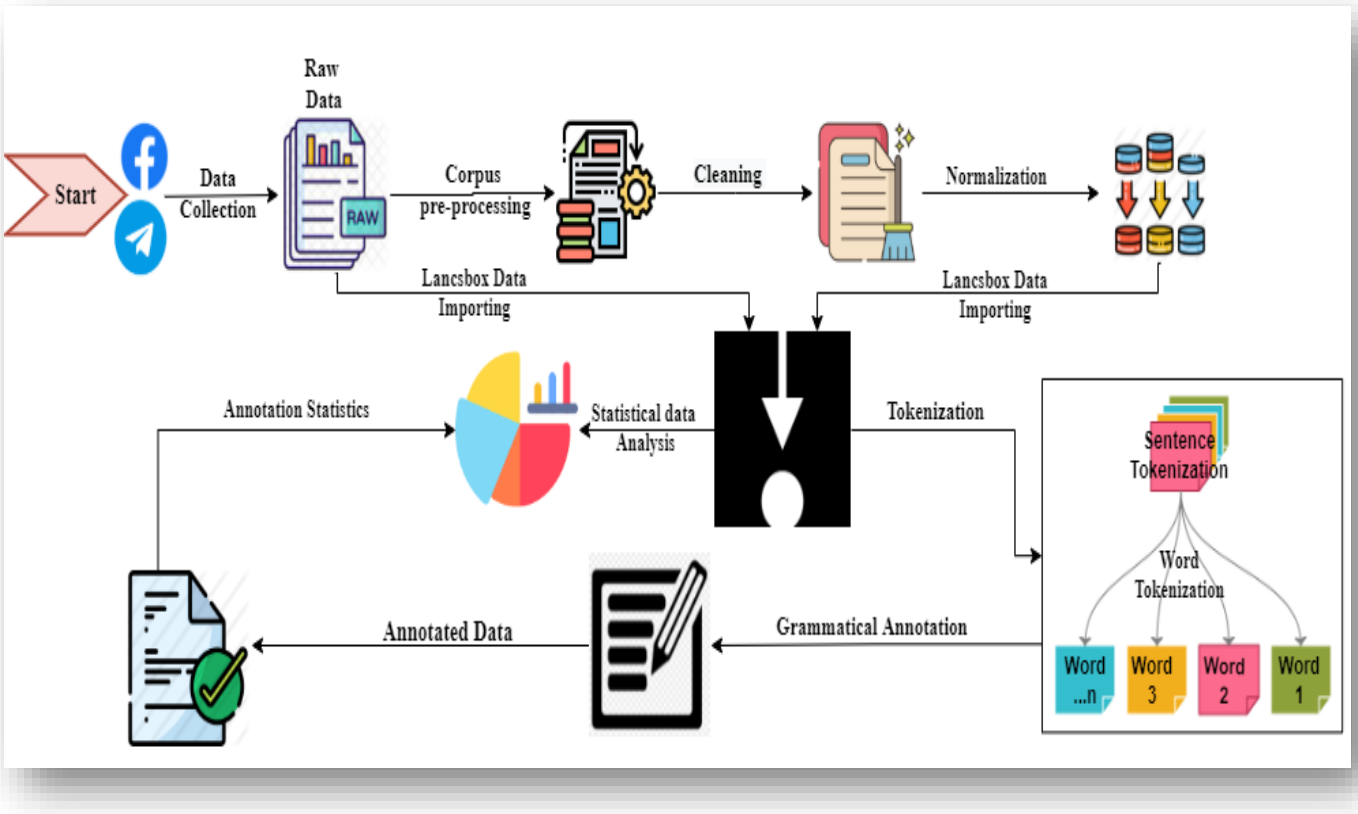


Figure 4.1 *Corpus developing process*

4.3.1 Data Selection

For data selection, certain criteria need to be taken into consideration. Hence, a conscious decision is made to adhere to Sinclair's common criteria (Sinclair 2004). These criteria are:

a. the mode of the text; whether the language originates in speech or writing, or perhaps nowadays in electronic mode;
b. the type of text; for example, if written, whether a book, a journal, a notice or a letter;
c. the domain of the text; for example, whether academic or popular;
d. the language or languages or language varieties of the corpus;
e. the location of the texts; for example (the English of) UK or Australia;
f. the date of the texts.

(Sinclair 2004, 8)

In the process of data selection for the present corpus, Sinclair's criteria apply as the following:

1. the mode of the text is electronic
2. the type of text; soap opera, i.e., drama serial that consists of fictional dialogues
3. the domain of the text; popular
4. the language or languages or language varieties of the corpus; San'ani Arabic dialect
5. the location of the texts; for example, social media text
6. the date of the texts. between the years 2017-2019

4.3.2 Data Collection

While developing the present corpus, attention was given to quality over quantity abiding by the following conditions:

- Data resources are reviewed carefully to validate data authenticity through native San'ani speakers.
- Data are collected manually to control the text closely.
- Mixed data, i.e., text that contains combinations of different languages or dialects, and
- Arabizi[19] are avoided.
- The raw data collected are kept intact.

[19] It is an encoded system that uses roman script instead of Arabic script to write Arabic text. It is also known as The Arabic Chat Alphabet.

The corpus data was collected from mainly two social media resources: Facebook and Telegram. To be more specific, the Facebook pages selected have linked channels on Telegram App, where they post on both platforms. The choice of Facebook is intentional based on its popularity in Yemen. According to the Global Stats website (2021) Facebook has been the most popular social media platform in Yemen since 2010. Figure 4.2 shows Facebook statistics of Yemeni users from 2017 till 2021.

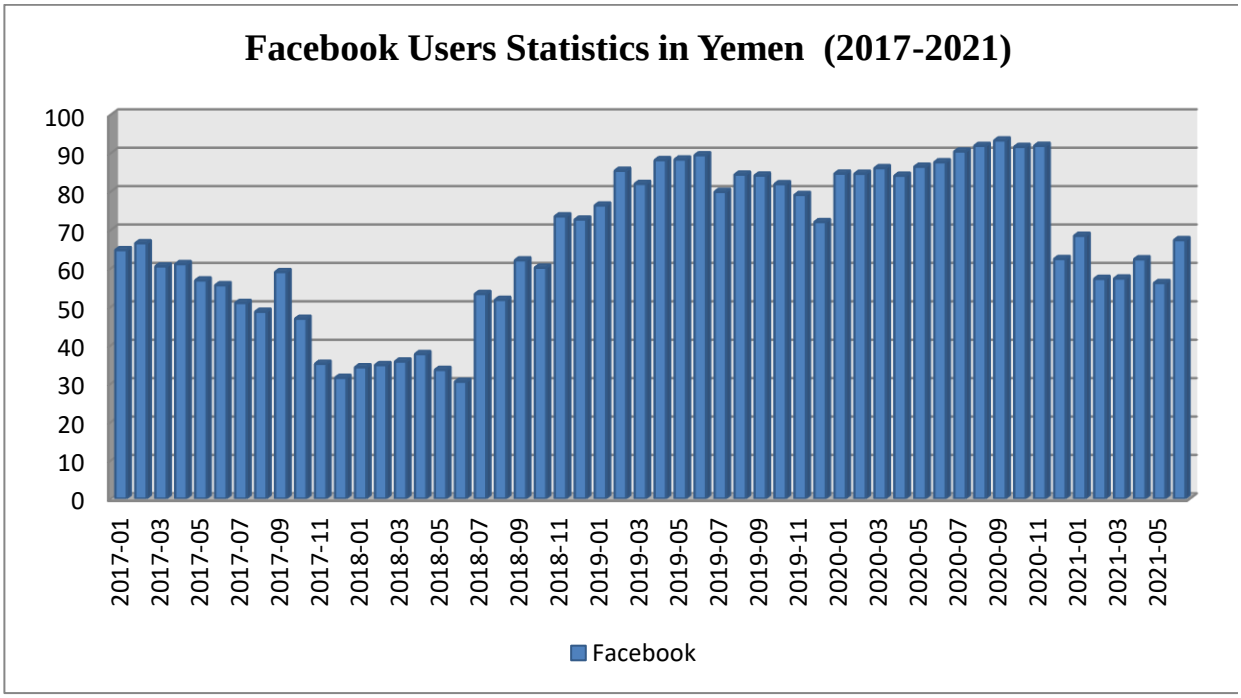


Figure 4.2 ***Facebook Users Statistics in Yemen (2017-2021)***

Source: Data is taken from Global Stats website (Statcounter)https://gs.statcounter.com/social-media-stats/all/yemen accessed on July/12/2021

Some of the data were collected manually from Facebook pages; e.g., صنعاني (قصص) /*qisˁasˁ sˁanʕa:ni:/* "San'ani stories" that post San'ani Arabic soap operas which are written in the form of fictional dialogues. These Facebook pages post on a daily basis and sometimes

weekly. Each post usually contains a part or two, similar to episode scripts. The writers tend to include their comments or announcements at the end of each part.

In addition, Telegram channels connected to the Facebook pages also provide posts in San'ani Arabic of the same kind. However, Telegram channels posts include heavily mixed dialects and MSA data; thus, manual data collection was done carefully to avoid such data. The Telegram Channels sometimes provide MS Word or PDF documents containing a collection of parts of some San'ani Stories. These documents were also collected and investigated.

The data collected was posted during the years 2017 and 2018. Since Facebook and Telegram are open-source platforms, no official permissions are needed for data collection. Mainly three stories were collected, written by two different writers[20]. First, the text was collected and saved in MS Word documents, where each part was saved in a separate word document and every story in a separate file. Then metadata of each story is saved in a separate MS Word document. Moreover, all the data is merged into a single document to perform the corpus statistics process.

4.4 Pre-processing

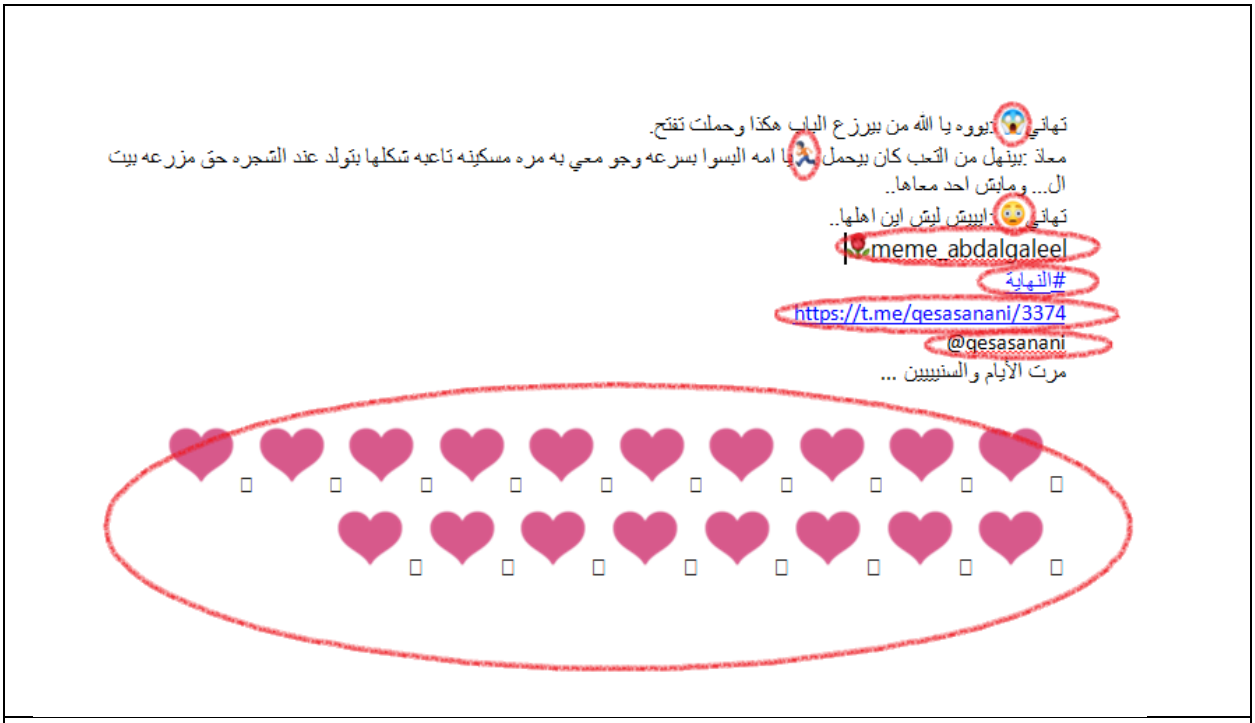
As established by many scholars, social media-based data is characterized by noisy data, such as non-standard spelling, emojis, emoticons, shortening or the omission of some letters, and lengthening (Crystal 2008; Habash 2010; and Farzindar & Inkpen 2015). Such noisy data can influence the accuracy of any further machine processing (Alshutayri and Atwell 2019). As our raw data is collected from social media platforms, it is in dire need of pre-processing. According to Zimmermann and Weibgerber (2004), pre-processing influences the accuracy of any machine

[20] Meme-Abdalgaleel and Shaimaa Ahmed

output. Upon investigation of our raw corpus, a lot of noise and ill-formed text is found, requiring refinement. Hence, in this stage of corpus building, three pre-processing steps took place; corpus cleaning, text normalization, and tokenization.

4.4.1 Data Cleaning

The first technique in pre-processing is data cleaning. It is an essential step for preparing the data for further analysis. Actually, it has a direct impact on processing and output accuracy. The data in hand is rich in noisy data such as emojis, emoticons, and non-Arabic text. Figure 4.3 displays an example of noise in our social media-based corpus collected from Facebook and Telegram.



"Tahani : oh Allah who is knocking the door like this and she hurried to open."

"Muaaz: panting out of fatiguehe was running mom change (your cloths) quickly and come with me there is a sick poor woman she looks like giving birth by the tree of the… farm house and no one is with her.."
"Tahani: what why where is her family.."
Meme_abdalgaleel
#the end
url http://t.me/ qesasanani/3374
@qesasanani
" Day and years passed "

Figure 4.3 ***Example of noise in the raw data***

The noisy data in the raw corpus were cleaned as follows:

- Emojis and Emoticons[21] are removed
- URLs and non-Arabic text are removed
- Images and shapes are removed

Other ill-formed data is to be pre-processed in the normalization step.

4.4.2 Text Normalization

Unlike MSA, Arabic dialects lack a conventional orthographic system. Dialectal Arabic speakers usually write their own tongue using Arabic script; however, they do not abide by any standard guidelines (Jarrar et al. 2017). In fact, a great deal of inconsistency is found in dialectal data to the point that writers not only contradict others' writing; but also, with their own writing. It is possible to find a word written in two or more different ways by the same writer and in the exact text. This is mainly caused by the fact that Arabic dialects used to be mainly spoken and not written. As a result, some dialectal data is written according to the writer's pronunciation with a great deal of variation.

[21]Emojis and Emoticons in our raw data are extra symbols; they were never meant to replace any word in data.

Such dialectal data variation poses several challenges for NLP tools and tasks (Habash et al. 2012b). In an attempt to face these challenges (Habash et al. 2012b) proposed a conventional orthography for dialectal Arabic (CODA) in general. However, currently, CODA guidelines are meant only for Egyptian Arabic. Moreover, other scholars (Eskander et al. 2013; Zribi et al. 2014; and Saadane & Habash 2015) also proposed guidelines and extensions for other dialects. However, no such convention or guidelines are available for San'ani Arabic. Hence, we applied a manual and shallow text normalization for our San'ani Arabic social media-based corpus, guided by Arabic language standards.

Ill-formed data analysis

After data cleaning, the corpus was investigated for noise to be resolved by normalization. Two main types are found; Faulty or non-standard spelling and Unspaced (connected) text. An explanation of these types is provided in this section. Table 4.1 presents ill-formed data types, examples, and solutions.

- Faulty or non-standard spelling

  Under this type, there are nine sub-types which are:

1. *hamza* (U+0621) related variations

   hamza /ء/ "glottal stop ʔ" is a common spelling issue in Arabic forms in general. In writing, it appears as subscript or a superscript with certain vowels or semi-vowels of the Arabic letters; i.e., /أ, إ, ئ, ؤ/"above or below the bare *ʔalif* (U+0627), above *alif maqsˁura*: (U+0649) or above the letter *wa:w* /w/ (U+0624)". Usually, *hamza* is left out by Arabic dialects writers and not included, which is the case in our data. According to Buckwalter (2007), the *hamza* (U+0621) and *maddah* (~) (U+ 0622) positioning is an acceptable

variation in orthography whenever it has a single meaning. It was the most common spelling issue in our corpus, so we adopted the bare *alif* as the standard form.

2. *ta:ʔ marbu:tˁa(t)* (U+0629) and *ha:ʔ* (U+0647)alternation

   ta:ʔ marbu:tˁa(t)/ة/ (U+0629) and *ha:ʔ / ه/* (U+0647) are mixed together at word end position. As a feature of San'ani Arabic, writers write the same word with both forms at word end position; i.e., with dots (*ta:ʔ marbu:tˁa /ة/* or without dots *ha:ʔ / ه/*. It is noticed that *ha:ʔ* (U+0647)is used more often than *ta:ʔ marbu:tˁa* (U+0629) Hence, we replaced */ة/with / ه/* at the word end position.

3. dˁa:d */ض/* (U+0636) and ðˁaʔ */ظ/* (U+0638) alternation

   Though dˁa:d is not part of the speech inventory of San'ani Arabic, it is still used in the script, resulting from using Arabic orthography standards. However, dˁa:d */ض/ is* mixed with ðˁaʔ */ظ/* in writing. In our data, sometimes words which contain dˁa:d */ض/* is written with ðˁaʔ */ظ/*

   and vice versa. So, in normalization, such confusion is resolved with the appropriate letter.

4. *ʔalıf maqsˁura*: */ى/* (U+0649), *ja:ʔ /ي/* (U+064A) alternation

   Some spelling error is related to the use of ʔalıf maqsˁura: */ى/* and ja*:ʔ /ي/* interchangeably at the word end position. Such misspellings are normalized using the correct letter.

5. Random misspellings

In addition, there are different spelling mistakes which are either typos[22] or incorrect or incomplete spelling. Normalization took care of such words correcting the misspelling.

6. Broken words

   Broken words are those words that include a space/space within a single word. Social media text is characterized by such noise. Normalization resolves the noise by removing any extra spaces.

7. Selective diacritics inclusion

   Originally, Standard Arabic script is written with diacritics which are included as subscript and superscript. These diacritics represent short vowels and gemination in Arabic. However, MSA and Dialectal Arabic writers ignore them. In our data, a small number of diacritics appear occasionally. These diacritics are stripped out from the text.

8. abbreviations

   It is another feature of social media text where writers write some words in short form using the initial letter or dropping other letters. It is similar to English (h r u?) instead of (How are you?). In this case, words are corrected, and the complete form of the word is given.

9. letter lengthening/elongation

   Letter elongation is also a result of social media text where some letters within a word are repeated several times. For instance, the word cute and what is written as (cuuuuuute) (whaaaaaat). Such noise is removed, and the correct spelling is given.

[22]typo here means a typographical error.

- Unspaced (connected) text

  Improper spacing of words may directly influence word tokenization. Hence, normalization plays an essential role in data pre-processing. There are two sub-types which are as follows:

1. alphanumerical words

   Alphanumerical words are numbers and letters connected, i.e., they are not properly spaced; Such as (33markets) instead of (33 markets). As a solution, proper spaces are included.

2. Connected words

   Like alphanumerical words, connected words are words without poor spacing, such as (goodjob) instead of (good job). So, they are correctly spaced.

Table 4.1 ***Ill formed types, solution and examples***

Ill formed Types		
Faulty or non-standard spelling		
Sub-type	Solution	Example
Hamza variations	Bare *alif* as the standard form.	الاكل→الأكل “the food”
ta:ʔmarbu:tˤa and *ha:ʔ* alternation	*ha:ʔ*	معقوله→معقولة “reasonable”
dˤa:d and ðˤaʔ alternation	corrected	موضوع→موظوع “Topic”
ʔalıf maqsˤura: and *ja:ʔ* alternation	corrected	الى→الي “to”
Random Misspellings	corrected	اللي→الي “Who, which“
Selective diacritics inclusion	removed	بسم→بِسم “In the name”
Broken words	Properly spaced	الحوش→الحوش “the yard”

abbreviations	corrected	ع→على “over, on”
letter elongation	corrected	بسسسسس→بس “enough , but”
Unspaced (connected) text		
alphanumerical words	Properly spaced	بارت122 → بارت 122 “part 122”
Connected words	Properly spaced	مشهي→مشهي “not she”

4.4.3 Tokenization

The tokenization process is crucial in data processing (Anandarajan et al. 2019). The text needs to be tokenized into sentences and words to further processing and analysis. Keeping in mind that the present data is meant to train a parts-of-speech tagger, two types of tokenization are required: word and sentence tokenization.

4.4.3.1 Word Tokenization

Word tokenization was performed using the LancsBox[23] (2018) (2018) tool. After normalization, the data was imported to the LancsBox tool to carry out word tokenization and statistical analysis. The word tokenization is done using white-space tokenization. Then, the output is checked manually for any correction. The number of tokens before pre-processing was 212,288, while after pre-processing, the number of tokens becomes 204,084. Table 4.4 presents the whole corpus size pre and post data pre-processing. Data statistics are presented and discussed in section 4.5.3.

[23] It is a software package developed at Lancaster University to analyze corpora and language data. It is freely available on: http://corpora.lancs.ac.uk/lancsbox/.

4.4.3.2 Sentence Tokenization

Sentence tokenization is an essential step in data processing, especially in Parts of Speech tagging, where words' context plays a significant role in identifying appropriate tags. Therefore, sentences need to be segmented carefully. Generally speaking, punctuation marks are the identifiers of sentence boundaries. In English, for instance, punctuation marks are used systemically, which is not the case in all Languages. Unfortunately, in Arabic, punctuation marks are not reliable tools for identifying sentence boundaries (Ditters 1991; Meiseles 1979; Stetkevych 2006; and Alkohlani 2015). Arabic writers use punctuation marks for decoration if not ignored (Ghazala 2004).

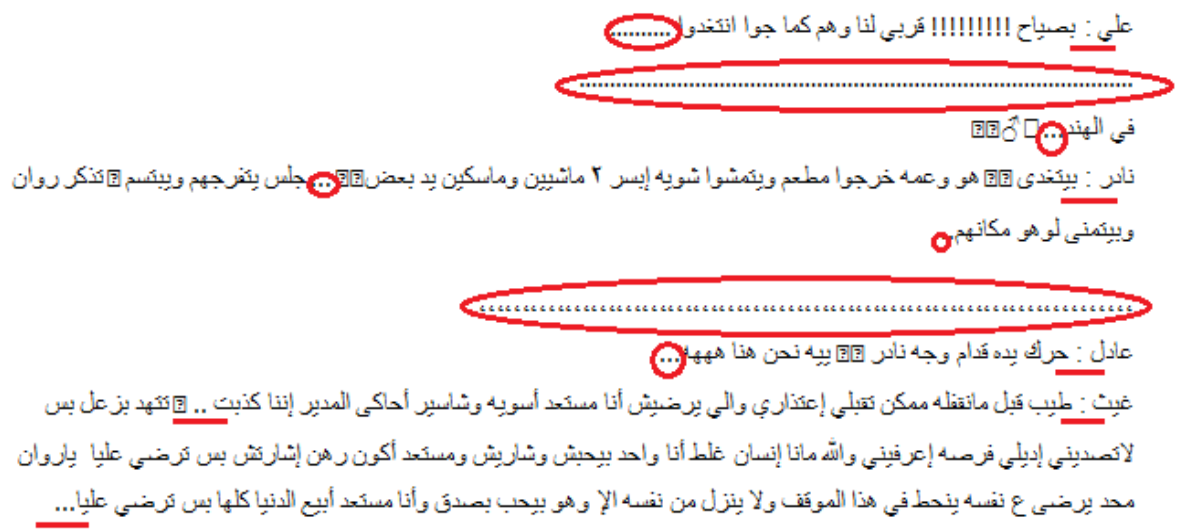
Moreover, Arabic text is characterized by lengthy sentences (Alkohlani 2015). Modern Arabic linguists considered Arabic sentences' unusual length an obstacle to identifying sentence boundaries (el-Shiyab 1990). Arabic Writers tend to make their sentences lengthy, either with coordination or subordination. According to Badr, Zbib, and Glass (2009), the average length of a sentence in the LDC Arabic news corpora is 25 words, considered longer than English. In English, the average sentence length is between 12 -17 words (Borja 2015).

In social media data, the situation is more complicated. Figure one shows that the text is characterized by writing inconsistencies, noise, and random use of punctuation marks. For instance, dots and commas appear as a string of dots or commas either at the end of lines or within the text. Moreover, other punctuation marks, such as semi-colons, question marks, and exclamation marks, are misplaced or totally ignored. Figure 4.4 is an example of punctuation marks’ random use in our data.

Alkohlani (2015) investigated Arabic sentences and their boundaries. Based on traditional Arabic grammarians' and Modern linguists' views of a sentence, she suggested a syntactic-

semantic criterion to identify sentence boundaries. This criterion states that a sentence must be syntactically independent and semantically informative.

In our project, sentence segmentation was applied manually, abiding by Alkohlani's criterion for sentence boundaries identification. As a result, our data was tokenized into 11,163 sentences. The average length of a sentence is between 12-25 words. It is fair to say that the sentence length does not affect the tokenization process. However, it might influence other NLP tasks.


علي : بصياح !!!!!!!! قربي لنا وهم كما جوا انتخدوا.....

.....

في الهند...

نادر : بيتغدى هو وعمه خرجوا مطعم وبيتمشوا شويه إبسر ٢ ماشيين وماسكين يد بعض... جلس يتفرجهم وبيتسم تذكر روان وبيتمنى لو هو مكانهم

.....

عادل : حرك يده قدام وجه نادر يه نحن هنا هههه...

غيث : طيب قبل مانقفله ممكن تقبلي إعتذاري والي يرضيش أنا مستعد أسويه وشاسير أحاكي المدير إننا كذبت .. تنهد بزعل بس لاتصديني إديلي فرصه إعرفيني والله مانا إنسان غلط أنا واحد بيحبش وشاريش ومستعد أكون رهن إشارتش بس ترضي عليا باروان محد يرضى ع نفسه ينحط في هذا الموقف ولا ينزل من نفسه الإ وهو بيحب بصدق وأنا مستعد أبيع الدنيا كلها بس ترضي عليا...

"Ali :yalling!!!!!!!!serve us and when they arrive (they'll) have lunch "

"in India …"

"Nadir: eating lunch with his uncle they got out of the restaurant and while hanging out together for a while he saw a couple walking hand in hand .. he was watching them smiling then he remembered Rawan whishing if he were in their place…"

"Aadil : waved his hand in front of Nadir's face yaiy we are here haaaa …"

"Ghayith: Ok before we drop it can you accept my apology and whatever pleases you I am ready to do it and I am going to tell the manager that I lied .. he sighed sadly but do not shut me out give me a chance know me wallah(swear expression) I am not a bad person I am someone who loves you and wants you I am ready to be at your disposal only to make you pleased with me Rawan nobody agrees to put himself in such position or descends himself unless he is genuinely in love I am ready to sell the world only to if you could be pleased with me …"

Figure 4.4 *An example of punctuation marks' random use in data*

4.5 Corpus statistical analysis

The corpus is analysed twice: pre- and post-cleaning using the LancsBox (2018) tool. For data analysis and tokenization, the LancsBox tool is used. Both analyses are conducted to reflect a clear picture of the raw corpus in all of its developing stages. The following three sub-sections discuss raw data analysis in detail. Section 4.5.1 presents pre- cleaning corpus statistics. Section 4.5.2 displays post- cleaning corpus statistics, while Section 4.5.2 describes the total corpus size.

4.5.1 Pre-Cleaning Corpus Statistics

As shown in Table 4.2 and Figure 4.5, the raw corpus size pre-cleaning is 212,288 tokens[24] and 26,244 types[25]. It consists of three stories titled: */fi: bajtʤadi/* "in my grandpa house", */dˁaɲa:ja: al-gadar/* "Destiny Victims" and */la: taxalajini: jati:mahmaratain/* "do not make me an orphan twice". The first story consists of 71,342 tokens, out of which 14,667 are types where the token to type ratio (TTR henceforth) is 5:1. It was posted in 2017 on Facebook[26]. The second was posted during 2016 and 2017 on Facebook. It consists of 129,580 tokens, out of which 21,483 are types. The TTR is 6:1. The third was posted on Facebook during the years 2017-2018. The calculation of tokens and types is 11,366 and 3,070, respectively, with a TTR of 3:1.

Notably, the types and TTR dropped down when the corpus statistics were calculated for all three stories together. Calculation shows that the sum of tokens is 212,288, out of which types

[24] Tokens here refer to the total number of individual words in a corpus.
[25] Types refer to the count of unique word forms in a corpus.
[26] https://m.facebook.com/537719673071970/

are only 26,244 with a TTR of 1 type to 8 tokens. The total calculation of types drops since there must be shared unique words in all the stories, which are counted only once instead of three times.

Table 4.2 *Pre-cleaning corpus calculation*

Title of the Story	Posting date	Platform	Tokens	Types	TTR
في بيت جدي */fi: bajtʤadi/* "in my grandpa house"	*2017*	Facebook and Telegram	71,342	14,667	5:1
ضحايا القدر */dˤaħa:ja: al-gadar/* "Destiny Victims"	*2016-2017*	FacebookandTelegram	129,580	21,483	6:1
لا تخليني يتيمة مرتين */la: taxalajini: jati:mahmaratain/* "do not make me an orphan twice"	*2017-2018*	Facebook	11,366	3,070	3:1
TOTAL			212,288	26,244	8:1

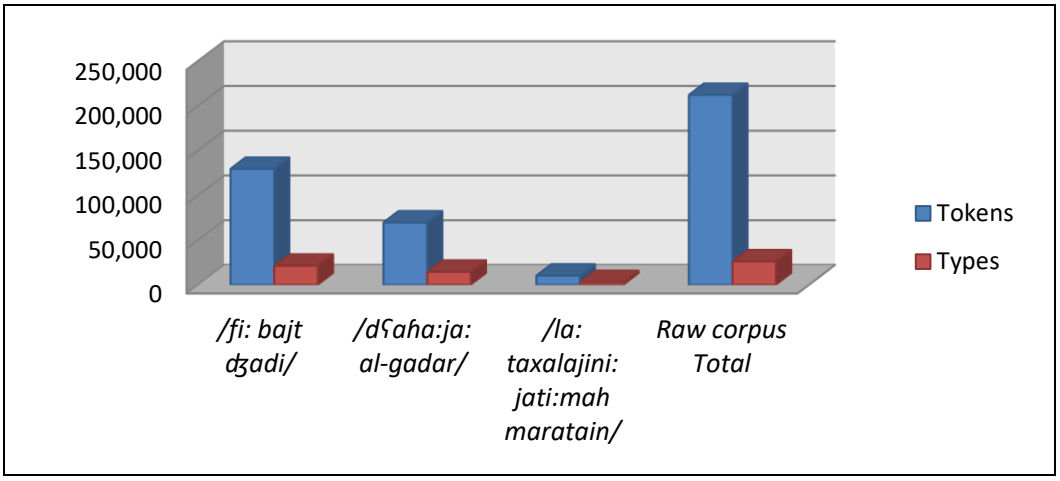


Figure 4.5 ***Pre-Cleaning Corpus visualization***

4.5.2 Post-Cleaning Corpus Statistics

After cleaning, the cleaned corpus was imported to the LancsBox(2018) tool to perform tokenization. Then Normalization was performed, and data was loaded to the LancsBox for one last time to extract data calculation. As shown in Table 4.3, each story size reduced post-cleaning in size. The first story number of tokens and types post-cleaning read as 96,299 tokens and 14,580 types. The second has 124,398 tokens and 20,439 types, while the third story contains 10,387 tokens and 3,039 types.

Compared with the pre-cleaning data, the difference in each story reads 2,043, 5,182, and 979 tokens, as a total of 8,204 tokens are eliminated from the raw corpus after pre-processing. Figure4.6 reflects the size of the final corpus versus noisy tokens eliminated from the corpus during pre-processing. Thus, the total size of the post-cleaning corpus is 204,084 tokens, out of which 24,712 are types.

Table 4.3 ***Post-cleaning corpus calculation***

Title of the Story	Tokens	Types	TTR
في بيت جدي */fi: bajtʤadi/* “in my grandpa house”	69,299	14,580	5:1
ضحايا القدر */dˤaħa:ja: al-gadar/* “Destiny Victims”	124,398	20,439	6:1
لا تخليني يتيمة مرتين */la: taxalajini: jati:mahmaratain/* “do not make me an orphan twice”	10,387	3,039	3:1
TOTAL	204,084	24,712	8:1

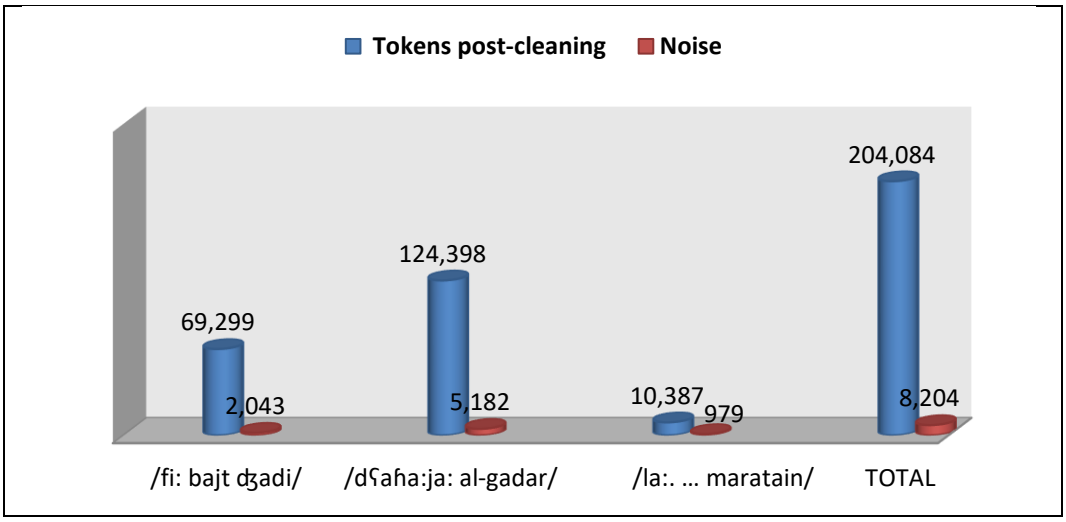


Figure 4.6 *Post-cleaning corpus size verses noise calculation*

4.5.3 Total Corpus Size

The total corpus size after pre-processing shrank by 4%. It decreased from 212,288 tokens to 204,084 tokens. Similarly, types count pre-cleaning versus post-cleaning reduced by 6%. As pre-cleaning types calculation was 26,244, but the post-cleaning reads as 24,712 types. TTR difference shows only in decimal where TTR in pre-cleaning is 8.25:1 while post-cleaning is 8.08:1. Table 4.4 and Figure 4.7 show total corpus statistics in all corpus development stages.

Table 4.4 ***Total corpus statistics***

Total Corpus	Tokens	Types
Pre-cleaning	212,288	26,244
Post-cleaning	204,084	24,712
TTR	8.25:1	8.08:1

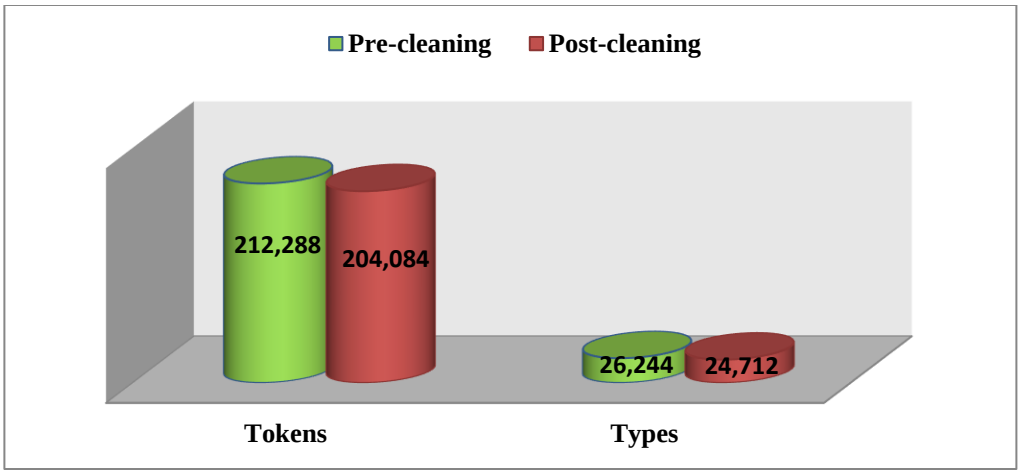


Figure 4.7 Total corpus statistics

4.6 Corpus Genre

Our corpus consists of fictional novels written in parts. Their main theme is drama; however, romance and tragedy are also present. They are written in San'ani Arabic, but sometimes writers use MSA or San'ani Arabic to include teasers, comments and ask for encouragement and participation from the audience at the end of certain parts. Also, writers usually include a moral lesson at the end of each novel and ask the audience about their opinions. In total, there are 414 parts. The first story */fi: bajtʤadi/* "in my grandpa house" consists of 154 parts. The second */dˁaɲa:ja: al-gadar/* "Destiny Victims" contains 230 parts. The third */la: taxalajini: jati:mahmaratain/* "do not make me an orphan twice" contains 39 parts.

4.7 Summary

This Chapter contains a description of the process of corpus development and pre-processing. It also presents a detailed statistical analysis of the corpus in hand. It consists of seven sections; the first two sections, i.e., 4.1 and 4.2, are Chapter introduction and corpus definition.

The third section, 4.3, is corpus development that describes the raw corpus selection and collection. Then section 4.4 introduces data pre-processing into three sub-sections: data cleaning, text normalization, and tokenization. These sections provide a clear picture of the intricacies of developing a social media-based corpus, especially when working on the non-standardized San'ani Arabic text. Additionally, we were able to extract specific normalization guidelines to deal with the existing noise and variations. As a result, a machine-readable social media-based corpus of San'ani Arabic was created and prepared for further processing.

Section 4.5 is corpus statistical analysis. Throughout this section, detailed corpus statistics were performed using the LancsBox tool. The statistics are conducted in the two stages, pre- and post-cleaning. Moreover, section 5.6 deals with the description of corpus genera. The final section is the chapter summary.

To conclude, this Chapter describes our method of corpus development and data pre-processing. It also reports corpus statistics at all stages of corpus development.

CHAPTER FIVE
TAGSET AND DATA ANNOTATION

This Chapter describes the adapted parts-of-speech tagset and the process of data annotation. It consists of four sections. The first section, i.e., 5.1, introduces the adapted tagset. The second section, 5.2, describes corpus annotation where text is enriched with parts-of-speech/grammatical tags. The third section, 5.3, is dedicated to annotation statistics. Finally, the fourth section, 5.4, summarizes the Chapter.

5.1 Tagset

This section deals with the adapted tagset within four sub-sections. The first introduces the concept "tagset". The second justifies the adapted tagset. Then the third describes our tagset, while the fourth compares it with the Bies/LDC Tagset.

5.1.1 What is a tagset?

A parts-of-speech tagset is a list of tags representing all the lexical classes of a language that is used to perform the parts-of-speech tagging. Khojah, Graside, and Knowels (2001) stated that a tagset is an essential component of any tagging tool and corpus annotation. The compilation or choice of a tagset depends on the linguistic analysis chosen and the degree of granularity needed. Thus, we can also define a parts-of-speech tagset as a set of unique labels used to annotate each token in a targeted text.

5.1.2 Justification for the adapted tagset

As reviewed in section 3.4, Chapter Three, there is no standard Arabic tagset used for parts-of-speech tagging. However, the RTS/ Bies tagset, also known as the LDC tagset, was found to be the most suitable for the current project as it holds the following advantages:

- Linguistically, this tagset covers all the main Arabic lexical classes.
- Computationally, it is a coarse tagset consisting of 24 tags, so both annotation speed and accuracy level are at benefit.
- It was widely used and tested in several tagging projects for Arabic and dialectal Arabic, e.g., the SVM tagger developed by (Diab, Hacioglu, and Jurafsky 2004), the Egyptian dialect parts-of-speech tagger done by (Duh and Kirchhoff 2005), the morphological analyzer, and SVM parts-of-speech tagger by (Nizar Habash & Owen Rambow 2005), and in the Analysis and Improvements of the Arabic Treebank parsing by (Kulick, Gabbard, and Marcus 2006).

Therefore, we adapted the LDC/Bies tagset with certain modifications to meet the structure of San'ani Arabic and the purpose of our work. Additionally, we ensure that our tagset only accounts for the syntactic features rather than the morphological ones. Hence, the modifications applied to the Bies/RTS tagset are meant to refine the tags and make them suitable for San'ani Arabic social media text annotation. This annotation aims to prepare a training corpus to train and build a parts-of-speech tagger for San'ani Arabic.

5.1.3 Description of the adapted tagset

The tagset used comprises twenty-Three tags, as shown in Table 5.1. The adapted tagset is coarse and utterly syntactic in the sense that it ignores all the inflectional features. These tags are divided into four categories which are:

1-Nominals	2-Verbs	3- Particles
	4-Others	

The first category, i.e., nominals, is further classified into three sub-categories:

1.1-Nouns	1.2-Pronouns	1.3-Other

The Nouns sub-category contains two tags:

NN	NNP

The NN tag indicates common nouns and Abbreviations such as: */sija:rih/* "car" and */ʔilax/* "etc.". The NNP tag refers to proper nouns such as: /*na:dijah*/ "Nadia (a name of a female)", */muʕa:ð/* "Muaath name of a male" */ʔaljaman/* "Yemen".

The Pronouns sub-category contains three tags:

PRP	WP	D_PRP

The PRP is used to tag personal pronouns and possessive pronouns. Such as */ʔiħna:/* "we", */kita:b-hin/* "their (F) book ". The WP denotes relative pronouns e.g., /ʔalij/" who". D_PRP is used to tag demonstrative pronouns such as: */ðajja:/* "this (M)" and */ha:ðɪ/* "this (F)".

The third sub-category of nominals, Other, consists of seven tags:

JJ	RB	WRB	CD
FCD	OD	FW	

The JJ tag is meant for adjectives, e.g., */ħa:lj/* "good" and */ʔaɣla/*"more expensive". The RB tag denotes adverbs such as */ha:kaða:/* "like this" and */bisa:ʕ/* "quickly". The WRB is used for relative adverbs as */mih/* "what". The CD tag refers to cardinal numbers, e.g., */xamsih/* "five". The FCD tag refers to foreign cardinal numbers such as */325/,* and */tu:/* "two". The OD tag is to tag ordinal numbers e.g., */ʔal-ʔawal/* "the first (M)" and */ʔal-ʔawalih/* "the first (F)". The FW tag represents the foreign words tag, such as*: /ʔi:skri:m/* "ice cream", and */suwbarma:rkit/* "supermarket".

The Verbs category is divided into two tags:

AUX_VB	VB

The AUX_VB tag is used to tag auxiliary verbs, known in Arabic terminology as deficient verbs such as */ka:n/* "to be". On the other hand, the VB tag is directed to main verbs such as */daxal/* "entered (3 SG M)."

The third main category is the Particles which is divided into six tags:

CC	SC	DT
RP	INTG_RP	IN

The CC tag signifies coordinating conjunctions such as */wa/* "and", and /*aw, walla:/* "or". The SC tags subordinating conjunctions e.g., */lama:/* "when" and/*law-ma:/* "when, while, until". The DT tag denotes determiners such as */ʔal/* "the", /*kul/* "all, every".The INTG_RP is meant to tag interrogative particles such as: /*wain*/ "where" and */kaif/* "how". The IN tag indicates prepositions, e.g., */fi:/* "in" and /ʕala:/ "on, on to, above". The RP tag covers all the other particles such as: /*mɪʃ, muʃ/* "not", and /gad/ "to indicate emphasis".

The last category, i.e., Other, contains Three tags:

UH	PUNC	SYM

The UH tag signifies interjections such as: */ju:h/* "oh" and /*aha:/* "Sound for assertion". The PUNC tag refers to punctuations marks as: full stop /. /, colon /:/ and semicolon/;/. The SYM tag refers to symbols such as: asterisk /*/ and percentile mark /%/.

Table 5.1 ***The adopted tagset***

Adapted Tagset		
Nominals		
Nouns		
1	NN	common noun or abbreviation
2	NNP	proper noun
Pronouns		
3	PRP	Personal & possessive pronoun
4	WP	relative pronoun
5	D_PRP	demonstrative pronoun
Others		
6	JJ	adjective
7	RB	adverb
8	WRP	relative adverb
9	CD	cardinal number
10	FCD	foreign cardinal number
11	OD	Ordinal number
12	FW	foreign word
Verbs		
13	AUX_VB	Auxiliary verb
14	VB	Main verbs
Particles		
15	CC	coordinating conjunction
16	SC	subordinating conjunction
17	DT	determiner
18	RP	particle
19	INTG_RP	Interrogative particle
20	IN	preposition
Others		
21	UH	interjection
22	PUNC	punctuation
23	SYM	symbol

5.1.4 Comparison between the adapted Tagset and the Bies/LDC Tagset

As reviewed in section 3.4.3.1, Chapter Three, the Bies tagset consists of 24 tags. The Bies tagset is a coarse tagset intended to improve the parsing performance of the PATB (Sawalhaand Atwell 2013). However, specific alternations are made to meet the structure of San'ani Arabic and the theme of this project which is parts-of-speech tagging. Moreover, we ensure that the tagset concentrates on word classes rather than inflectional features. Hence, this sub-section presents the

alternations made in the Bies tagset in detail. Table 5.2 shows both tagset, i.e., Bies and the adapted tagset. The modifications made are as the following:

a- Alternations in the Nominals category

- The number feature is ignored in the nouns' tags. The NN and NNS tags, which refer to singular common nouns and dual/plural common nouns, respectively, are combined into NN.
- Similarly, the NNP and NNPS tags that denote singular and dual/plural proper nouns are combined in one tag that is NNP, and the number feature is dropped.
- In the pronouns category, the PRP$ tags, which tags possessive personal pronouns, is joined with the personal pronouns tag PRP. Since possessive pronouns are bound morphemes that are attached to a stem and do not appear as free morphemes in San'ani Arabic.
- An additional tag in the pronouns sub-category is added to denote demonstrative pronouns, i.e., D_PRP.
- Within the other sub-category, two additional tags are added: FCD, i.e., foreign cardinal numbers, and OD, i.e., ordinal numbers.

b- Alternations in the Verb category

- The aspect and voice features are deleted from the main verb tag. So, VBP, i.e., active imperfect verb, VBN, i.e., passive imperfect/perfect verb, VBD, i.e., active perfect verb, and VB, i.e., imperative verb, are all joined into VB, which denotes the main verb.
- An additional tag, i.e., AUX_VB, is given to tag auxiliary verbs.

c- Alternations in the Particle category

- Two tags are added, namely SC and INTG_RP, to tag subordinating conjunctions and interrogative particles, respectively.

d- Alternations in the Other category

- The NUMERIC_COMMA tag, i.e., the letter ر r used as a comma, is replaced with the SYM tag to tag symbols. Moreover, the NO_FUNC is not used.

Table 5.2 ***Bies tagset and the adapted tagset content***

Bies Tagset		Adapted Tagset	
NOMINALS			
Nouns			
NN	singular common noun or abbreviation	NN	common noun or abbreviation
NNS	plural/dual common noun		
NNP	singular proper noun	NNP	proper noun
NNPS	plural/dual proper noun		
Pronouns			
PRP	personal pronoun	PRP	Personal & possessive pronoun
PRP$	possessive personal pronoun		
WP	relative pronoun	WP	relative pronoun
		D_PRP	demonstrative pronoun
Other			
JJ	adjective	JJ	adjective
RB	adverb	RB	adverb
WRB	relative adverb	WRB	relative adverb
CD	cardinal number	CD	cardinal number
		FCD	foreign cardinal number
		OD	Ordinal number
FW	foreign word	FW	foreign word
PARTICLES			
CC	coordinating conjunction	CC	coordinating conjunction
		SC	subordinating conjunction
DT	determiner/demonstrative pronoun	DT	determiner
RP	particle	RP	particle
		INTG_RP	Interrogative particle
IN	preposition or subordinating conjunction	IN	preposition
VERBS			
VBP	active imperfect verb	AUX_VB	Auxiliary verb
VBN	passive imperfect/perfect verb		
VBD	active perfect verb	VB	Main verbs
VB	imperative verb		
OTHER			
UH	interjection	UH	interjection
PUNC	punctuation	PUNC	punctuation
NUMERIC_COMMA	the letter ر r used as a comma	SYM	symbol
NO_FUNC	unanalysed word		

5.2 Corpus Annotation

Since our corpus is a social media-based corpus, several decisions are made prior to the grammatical annotation. First, we must consider the type of tagset used for the annotation, so we decided to use a coarse tagset explained in the earlier section 5.1.3. Second, our annotation adheres to Leech maxims of corpus annotation by Leech (1993, 275):

> (1) It should always be easy to dispense with annotations, and revert to the raw corpus. The raw corpus should be recoverable.
> (2) The annotations should, correspondingly, be extractable from the raw corpus, to be stored independently, or stored in an interlinear format.
> (3) The scheme of analysis presupposed by the annotations—the annotation scheme—should be based on principles or guidelines accessible to the end-user. (The annotation scheme consists of the set of annotative symbols used, their definitions, and the rules and guidelines for their application.)
> (4) It should also be made clear how, and by whom, the annotations were applied.
> (5) There can be no claim that the annotation scheme represents 'God's truth'. Rather, the annotated corpus is made available to a research community on a *caveat emptor* principle. It is offered as a matter of convenience only, on the assumption that many users will find it useful to use a corpus with annotations already built in, rather than to devise and apply their own annotation schemes from scratch (a task which could take them years to accomplish).
> (6) Therefore, to avoid misapplication, annotation schemes should preferably be based as far as possible on 'consensual', theory-neutral analyses of the data.
> (7) No one annotation scheme can claim authority as a standard, although *de facto* interchange 'standards' may arise, through widening availability of annotated corpora, and perhaps should be encouraged. (Leech 1993, 275)

Third, the orthographical variations are dealt with using the normalization rules described in section 4.4.2 Chapter Four. Fourth, the data has to be annotated manually and by native speakers of San’ani Arabic. Finally, our annotation is guided by the PATB annotation guidelines described by (Maamouri et al. 2008).

5.2.1 Annotation Process

The annotation performed is a grammatical/parts-of-speech annotation. It is mainly performed manually by a native speaker of San'ani Arabic over two years, i.e., 2018 and 2019. Moreover, data is annotated in a .xls extension document, consisting of two columns. The first column presents the token and the second contains the appropriate tag. After each sentence, a blank row is left as a sentence boundary marker. Figure 5.1 shows an example of the annotation format.

عادل	NNP
:	PUNC
أيوه	UH
..	PUNC
الدكتور	NN
:	PUNC
التشخيص	NN
الأولي	JJ
أعراض	NN
جلطه	NN
في	IN
القلب	NN
ان	RP
شاء	VB
الله	NNP
خير	NN
..	PUNC

Figure 5.1 An example of parts-of-speech tags annotation format

5.3 Annotation Statistics

All of the data, which counts 204,084 tokens, were annotated fully. The frequency and percentage of each tag are introduced in Table 5.3. Figure 5.2 presents the percentage of tags' main categories. The Nominal category, which contains 12 tags (NN, NNP, PRP, WP & D_PRP, JJ, RB,

WRB, CD, FCD, OD, & FW), shows the greatest frequency of 97,958 tokens which represents 48% of the data. Within the Nominal category, the three sub-categories, i.e., nouns (NN& NNP), pronouns (PRP, WP & D_PRP) and others (JJ, RB,WRB, CD,FCD, OD, & FW), calculate 79,343; 7,088; and 11,527 respectively. The second-highest category is Verb (AUX_VB & VB) with 48,288, i.e., 24%. Then, the Particle (CC, SC, DT, RP, INTG_RP & IN) category scores the third highest frequency, which calculates as 31,601, which is 15%. The final score is the Other (UH, PUNC & SYM) category, which counts 26,237, representing 13% of the data.

Table 5.3 *The annotation statistics*

Tags	Frequency	Percentage %
NN	56,479	27.7
VB	46,553	22.8
PUNC	23,406	11.5
NNP	22,864	11.2
IN	16,150	7.9
RP	9,334	4.6
JJ	6,137	3
PRP	4,618	2.3
RB	3,313	1.6
INTG_RP	2,880	1.41
UH	2,780	1.4
CC	1,569	0.8
D_PRP	1,774	0.9
SC	1,402	0.7
AUX_VB	1,735	0.8
WP	696	0.3
OD	512	0.2
FCD	604	0.3
FW	472	0.2
CD	277	0.1
DT	266	0.1
WRB	212	0.1
SYM	51	0.02

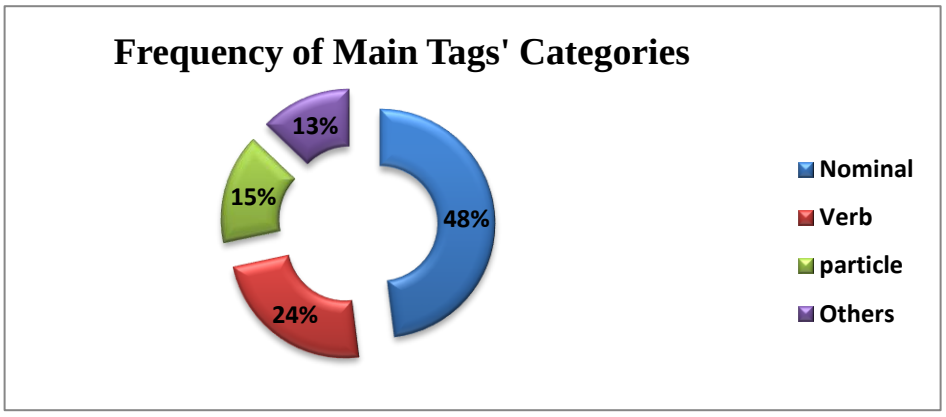


Figure 5.2 Frequency of main tags' categories

5.4 Summary

This chapter introduces the adapted tagset as well as the data annotation process and statistics. Our tagset is adapted from the Bies/LDC tagset, which is most suitable in coverage and size. It is a coarse tagset consisting of 23 tags. The modifications are made to fit the structure of San'ani Arabic. This tagset is used to enrich the data with parts-of-speech annotation and for the automatic parts-of-speech tagging.

Additionally, the corpus annotation is explained. We followed Leech maxims (1993) for data annotation. The annotation was done manually by a native speaker of San'ani Arabic. The annotation aims to prepare a training corpus for training the automatic tagger. After the annotation, a statistical analysis of the annotated corpus is conducted, where the frequency and percentage of the tags are displayed.

CHAPTER SIX
BIDIRECTIONAL-GATED RECURRENT UNITS-CONDITIONAL RANDOM FIELDS (BI-GRUs-CRF) MODEL DESCRIPTION

6.1 Introduction

This chapter describes the BI-GRUs-CRF Model for Parts-of-Speech Tagging for San'ani Arabic. In 2018, Che et al. (2018) proposed the addition of a CRF component to a bidirectional GUR- model to segment Chinese words. They combined the two methods to enhance the segmentation result. Their results were quite promising, showing high performance in segmenting Chinese words. We adopted this model to perform parts-of-speech tagging of San'ani Arabic motivated by the following model advantages:

- This is a sequence processing model which deals with databases of longer sequences accurately using lesser memory space than other models; hence, it is suitable for parts-of-speech tagging.
- The training of a BI-GRUs-CRF model is faster and easier than other neural network models (Jozefowicz, Zaremba, and Sutskever 2015).
- The BI-GRUs Layers are designed to deal with the previous and following information of the input data, making it more efficient.
- The CRF layer enhances the output's prediction using sentence-level features.
- BI-GRUs-CRF Model outperforms the GRU, BI-GRUs, and CRF Models (Che et al. 2018).

This chapter introduces the structure of the model, presenting each component in detail. It consists of four main sections: Section 6.1 introduction, Section 6.2 The Bidirectional-Gated Recurrent Units- Conditional Random Fields (BI-GRUs-CRF) Model for parts-of-speech tagging, Section 6.3The Graphical user interface (GUI) of the San'ani Arabic parts-of-speech Tagger, and section 6.4 A Summary.

6.2 Bidirectional-Gated Recurrent Units- Conditional Random Fields (BI-GRUs-CRF) Model for Parts of Speech Tagging

The Bidirectional-Gated Recurrent Units-Conditional Random Fields (BI-GRUs-CRF) Model utilizes two methods to parts-of-speech tagging, i.e., deep learning and stochastic. The deep learning part is the BI-GRUs network, and the stochastic one is the CRF. The CRF is integrated into the model as a layer that follows the output of the BI-GRUs network. The function of the first part, i.e., the BI-GRUs network, is to acquire the past and future information of the input text in both directions, i.e., forward and backward, and the CRF layer is to predict the appropriate tag for each word achieving the optimal tagging sequence.

In order to explain the model, we will introduce the following basic concepts: RNN, BI-RNN, GRU, BI-GRUs, and CRF. Then the BI-GRUs-CRF model will be described.

6.2.1 Recurrent Neural Network (RNN)

A recurrent Neural Network (RNN) is a type of neural networks that is designed to effectively deal with sequential data rather than spatial data, which is better dealt with using Convolutional Neural Networks (CNN) (Zhang 2021). Unlike RNN, in CNN, the input and the output are totally independent. So, the significant advantage of RNN is that it uses the past information of the previous step and feeds it as the input of the present step. This advantage

benefits several NLP tasks that require sequence labelling, such as parts-of-speech tagging and Named Entity Recognition (NER). In the case of parts-of-speech tagging, the context of a word, i.e., neighboring words, is crucial in predicting the suitable parts-of-speech tag. Hence, RNN stores the past information in a hidden layer that is used to remember the needed sequential information in the following input.

The major difference between CNN and RNN is how the feed is proceeded. In CNN, the input is fed at one go, while in RNN, the input is fed one by one and in a sequence. Then the RNN incorporates the output with the following input. Figure 6.1 shows the basic architecture of RNN.

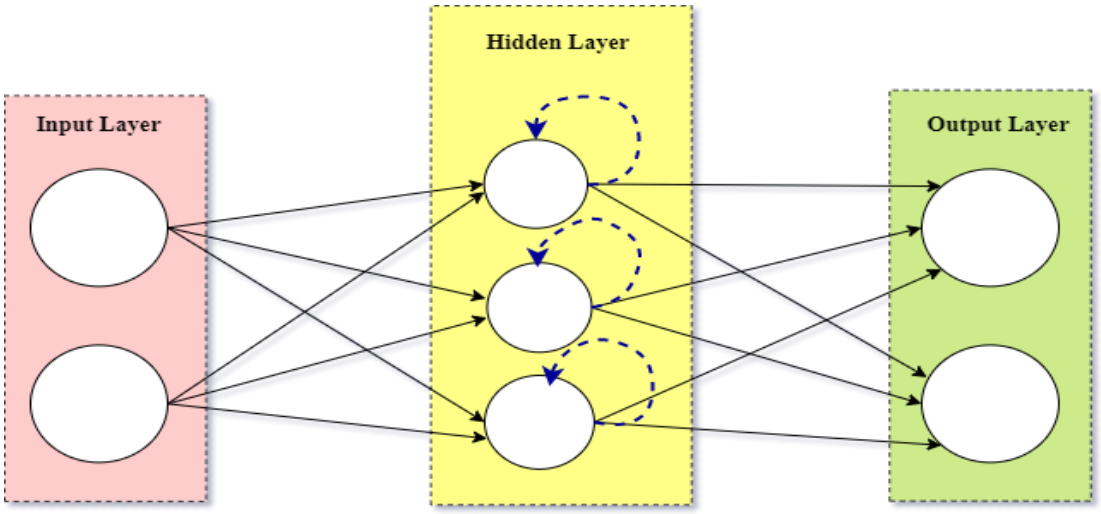


Figure 6.1 *The architecture of the Recurrent Neural Network*

As shown in Figure 6.1, RNN has a memory that allows it to add the past information to the current input, and this is the meaning of "recurrent". So it uses the same parameters in processing

each input or hidden state to reach the output. Such quality makes the RNN less complex in comparison with other networks. To calculate current state formula (1) is used:

$$\boldsymbol{a}^{<t>} = \boldsymbol{f}\,(\boldsymbol{a}^{<t-1>}, \boldsymbol{x}^{<t>}) \tag{1}$$

where:

$a^{<t>}$refers to the current state

$a^{<t-1>}$refers to the previous state

$x^{<t>}$refers to input state

For output calculation formula (2) is used:

$$\boldsymbol{y}^{<t>} = \boldsymbol{W}_{\boldsymbol{hy}}\boldsymbol{a}^{<t>} \tag{2}$$

Where:

$y^{<t>}$ refers to the output

W_{hy} refers to the weight at output layer

There are several forms of RNN, which are basically defined based on the input-output correspondence. These types are One-to-One, One-to-Many, Many-to-One, and Many-to-Many. In our case, which is Parts-of-speech tagging, we use Many-to-Many RNN. The Many-to-Many type is further divided into two sub-types. The first type is where the number of inputs equals the number of outputs $T_x = T_y$while the second is the opposite, i.e., the number of inputs does not equal the number of outputs$T_x != T_y$. In Parts-of-speech tagging the first sub-type is used where $T_x = T_y$. The architecture of Many-to-Many RNN is shown in Figure 6.2

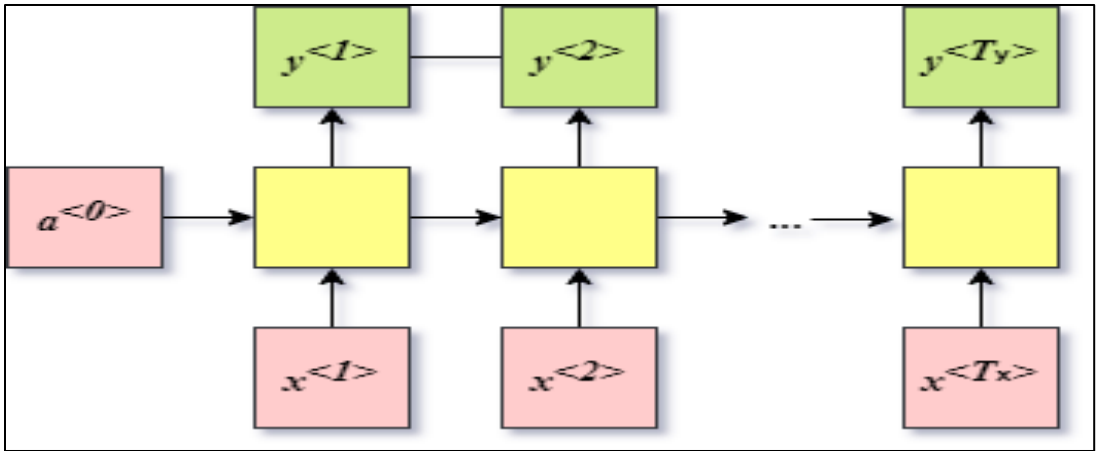


Figure 6.2 The structure of Many-to Many RNN

In Figure 6.2, $x^{<1>}, x^{<2>}$ *and* $x^{<Tx>}$ are the inputs. While $y^{<1>}, y^{<2>}$ *and* $y^{<Ty>}$ are the outputs. $a^{<0>}$ symbolizes the activation. The Figure clearly shows that the number of inputs T_x equals the number of the outputs T_y.

To calculate both target and activation, a deeper look into a cell at a time step t is provided in Figure 6.3

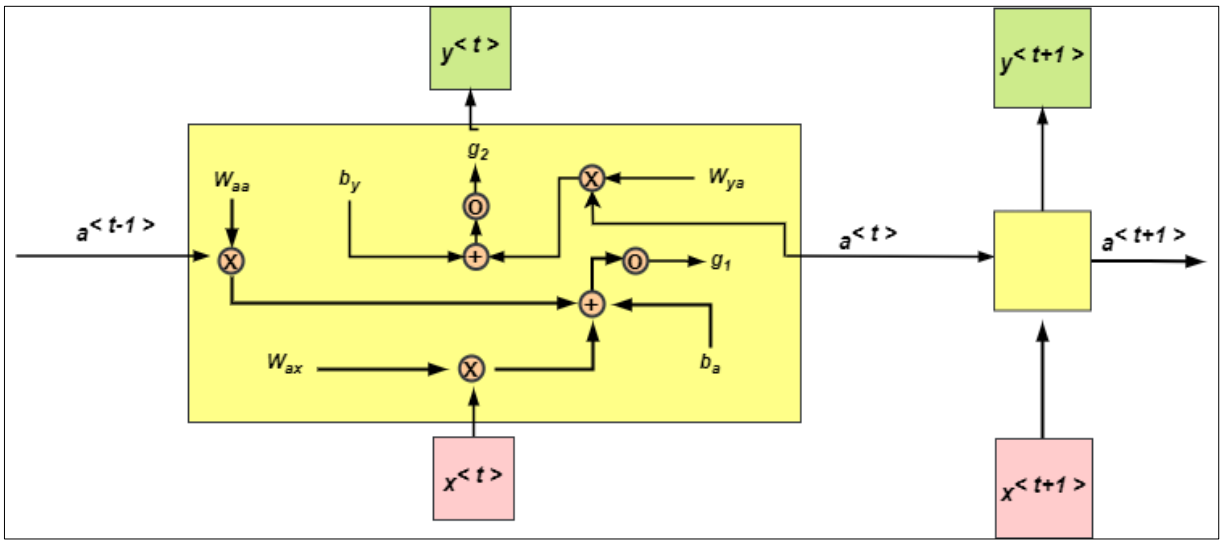


Figure 6.3 ***The RNN cell at time step t***

Source: adopted from https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks

Formula (3) expresses the calculation of the activation $a^{<t>}$ at each time step $t$and Formula (4) calculates the target $y^{<t>}$for each time step t:

$$\boldsymbol{a^{<t>} = g_1(x^{<t>}W_{ax} + a^{<t-1>}W_{aa} + b_a)} \quad (3)$$

$$\boldsymbol{y^{<t>} = g_2(a^{<t>}W_{ya} + b_y)} \quad (4)$$

Where:

$y^{<t>}$refers to the tag of word tag.

$a^{<t>}$refers to the state at time step t.

W_{ya}, W_{ax}, W_{aa}: are coefficients that are shared temporally

g_1 , g_2 : Activation functions they differ from architecture to another. It could be a tanh, a sigmoid or a relu. Formula (5), (6) and (7) represent tanh, sigmoid and relu respectively.

Tanh

$$\boldsymbol{g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}} \quad (5)$$

Sigmoid

$$\boldsymbol{g(z) = \frac{1}{1 + e^{-z}}} \quad (6)$$

Relu

$$\boldsymbol{g(z) = max(0, z)} \quad (7)$$

As explained earlier, the intelligent structure of RNN provides several advantages. Some of these advantages can be summarized as: the ability to process inputs at any length, making use of the past information and creating dependency between inputs in different stages through the hidden layer recurrence. Moreover, since the parameters are the same for each input, the model size does not grow with the increase of the inputs. Another advantage is that weights are saved to be shared across time. However, the big disadvantage of RNN is vanishing or exploding gradients[27]. This disadvantage is dealt with using other variants of the RNN involving certain techniques. In the following sub-section, we will briefly introduce two variants of RNN that are used.

6.2.1.1 Bidirectional Recurrent Neural Networks (BI-RNN)

It is a variant architecture of RNN which was developed by Schuster andPaliwalin1997 (Schuster&Paliwal 1997). It contains two hidden layers, backward and forward, that are connected to the same output. This bidirectionality allows the output to gain information from the past as well as the future simultaneously. Hence the input is not fixed as in the RNN. The major upgrade of BI-RNN is that it allows the current state to make use of future information, contrary to the RNNS where future information is not reachable from the current state (Salehinejad et al 2017). Figure 6.4 shows the structure of BI-RNN.

[27] Vanishing and exploding gradients are problems that may be encountered in the training of Artificial Neural Networks including. Vanishing gradients are the case when the number of derivatives in a network is small then the gradients will decrease exponentially till they ultimately vanish. Exploding gradients, on the other hand, are the case when the derivatives multiply and grow larger in number then the gradients increase more and more till they explode. In both cases the network will not function properly. For more information refer " Zhang, Aston, Zachary C. Lipton, Mu Li, and Alexander J. Smola. "Dive into deep learning." arXiv preprint arXiv:2106.11342 (2021)"

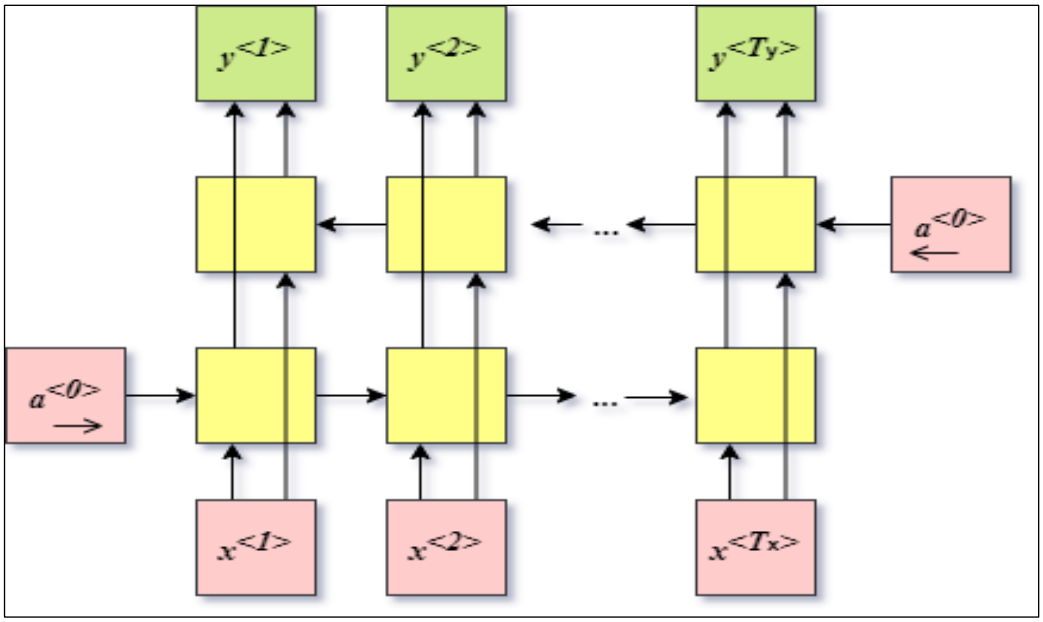


Figure 6.4 The Structure of BI-RNN

Source: Amidi, Afshine and Amidi, Shervine "Recurrent Neural Networks cheatsheet"*Stanford University*, Accessed on 08-09-2021https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks

6.2.1.2 Gated Recurrent Units (GRUs)

GRU is a variant of RNN that employs a gating mechanism to regulate the circulation of information within the cells in neural networks. It was introduced by Cho et al in 2014 to target capturing dependencies of large sequential data without losing information from previous steps of data processing (Cho et al 2014). In fact, GRU offers a solution to the vanishing/exploding gradients problem of RNN through the use of its gating units. As shown in Figure 6.5, in structure, GRU is similar to Long Short-Term Memory (LSTM) as both target information managing. However, unlike LSTM, GRU has only two gates which operate differently. These two gates are: a reset gate and an update gate. Figure 6.6 presents the inner mechanics of the GRU cell.

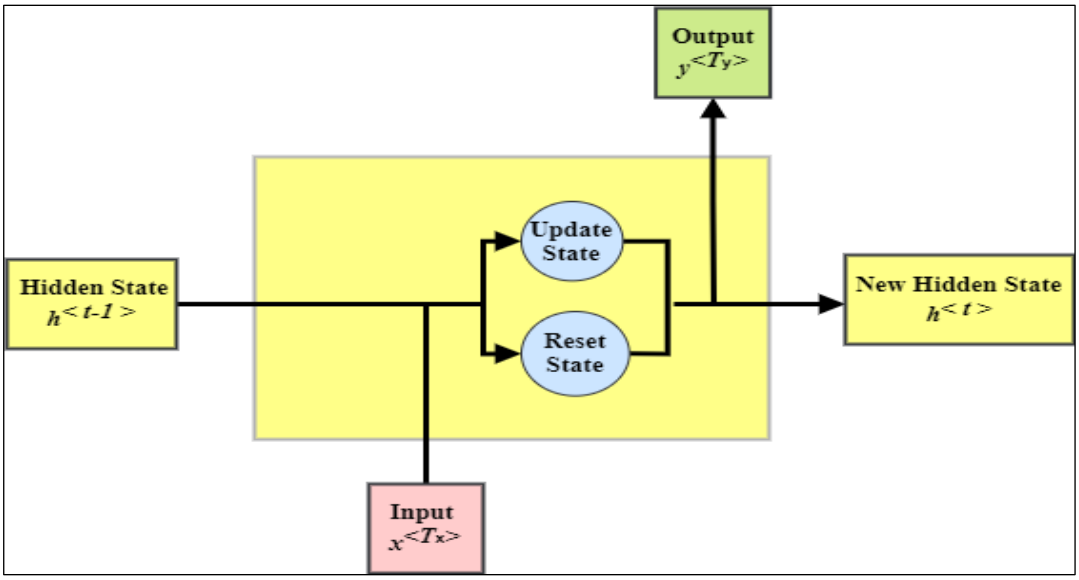


Figure 6.5 General Structure of GRUs

Source: Loye, Gabriel "Gated, Recurrent Unit (GRU) With PyTorch" ,FLOYDHUB, Jul 22, 2019
https://blog.floydhub.com/gru-with-pytorch/

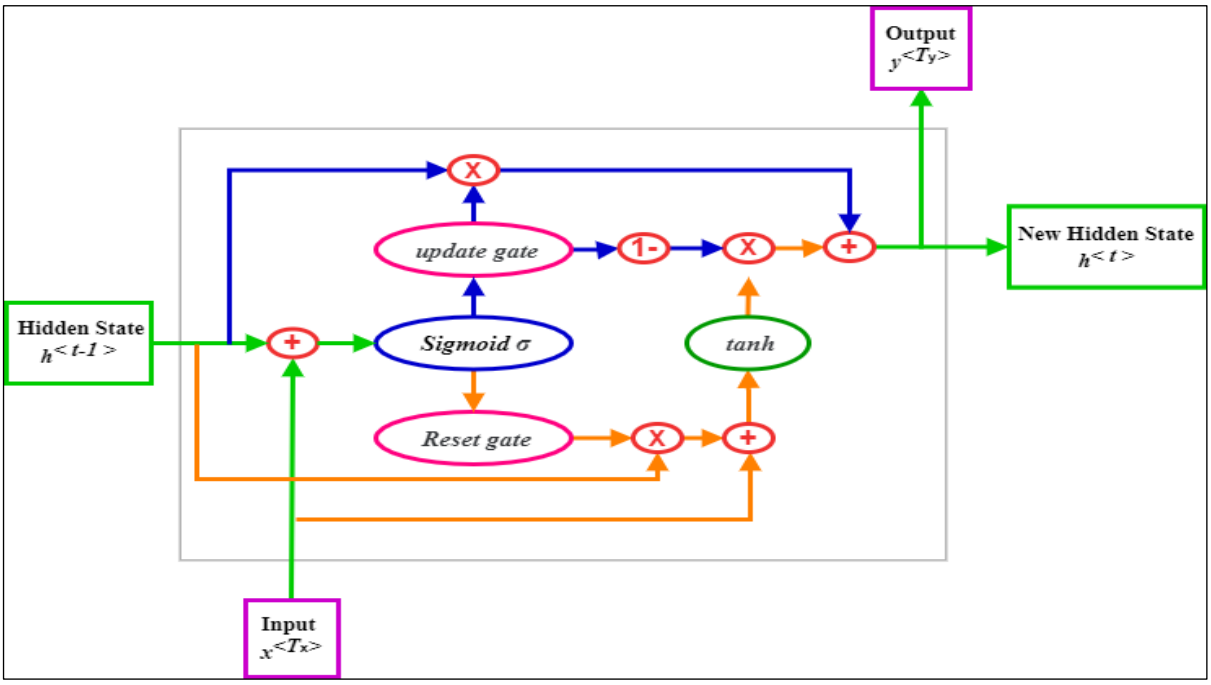


Figure 6.6 The inner mechanics of the GRU

Source: Loye, Gabriel "Gated, Recurrent Unit (GRU) WithPyTorch" ,FLOYDHUB, Jul 22, 2019
https://blog.floydhub.com/gru-with-pytorch/

6.2.1.2.1 Reset Gate

The reset gate deals with the short-term dependencies (short term memory). More specifically, it manages how much of the previous memory is cooperated with the new input as it forgets any non-useful past information. Figure 6.7 shows the stream of the reset gate. The calculation is done by multiplying the previous hidden state $\boldsymbol{h}^{<t-1>}$ and the current input $\boldsymbol{x}^{<t>}$ with their weight parameters $\boldsymbol{W}^{<xr>}$ and $\boldsymbol{W}^{<hr>}$ and sumthe results of the multiplication. Then the sum is passed through a sigmoid function $\boldsymbol{\sigma}$ to transform the value to fall between the intervals (0,1) as shown in equation (8).

$$\boldsymbol{Reset\ gate} = \boldsymbol{\sigma}(\boldsymbol{x}^{<t>}\boldsymbol{W}^{<xr>} + \boldsymbol{h}^{<t-1>}\boldsymbol{W}^{<hr>}) \quad (8)$$

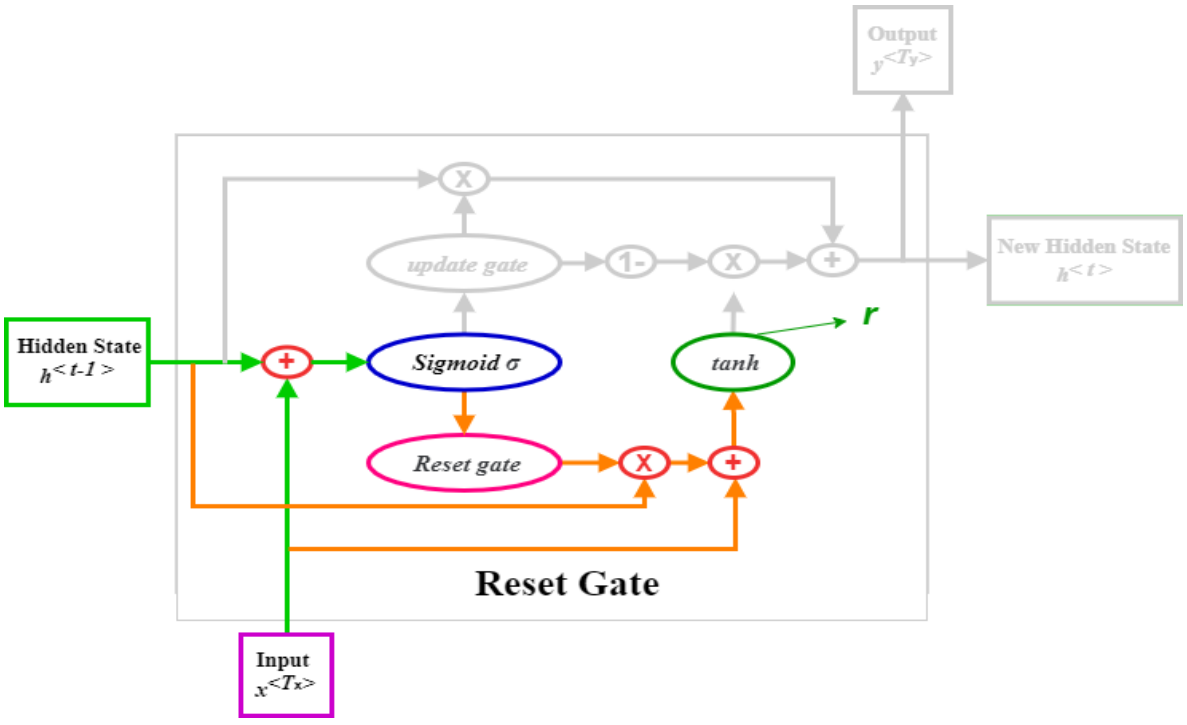


Figure 6.7 The reset gate stream

To get the output of the reset gate $\boldsymbol{r}$, equation (9) is used. Initially the previous hidden state $\boldsymbol{h}^{<t-1>}$is multiplied in a trainable weight $\boldsymbol{W}^{<h1>}$. Then the result undergoes an entrywise product multiplication (Schur product) with the reset gate. This part of the equation is responsible for identifying the amount of information retained from the previous time step so that it is used with the new inputs. The rest of the equation shows the current input$\boldsymbol{x}^{<t>}$ gets multiplied with a trainable weight$\boldsymbol{W}^{\boldsymbol{x1}}$ and then summed with the previous part to undergo a non-linear activation $\boldsymbol{tanh}$ function to get the final result of $\boldsymbol{r}$.

$$\boldsymbol{r} = \boldsymbol{tanh}(\boldsymbol{reset\ gate} \odot (\boldsymbol{W}^{<h1>}.\boldsymbol{h}^{<t-1>}) + \boldsymbol{W}^{\boldsymbol{x1}}.\boldsymbol{x}^{<t>}) \quad (9)$$

6.2.1.2.2 Update Gate

The update gate, on the other hand, is responsible for the long term dependencies (long term memory), i.e., it decides how much of the previous memory is useful for the future to be retained. Moreover, update gate controls the addition of new information to be saved. As shown in Figure 6.8 the stream of update gate is initiated from the current input $\boldsymbol{x}^{<t>}$and the previous hidden state$\boldsymbol{h}^{<t-1>}$**.**

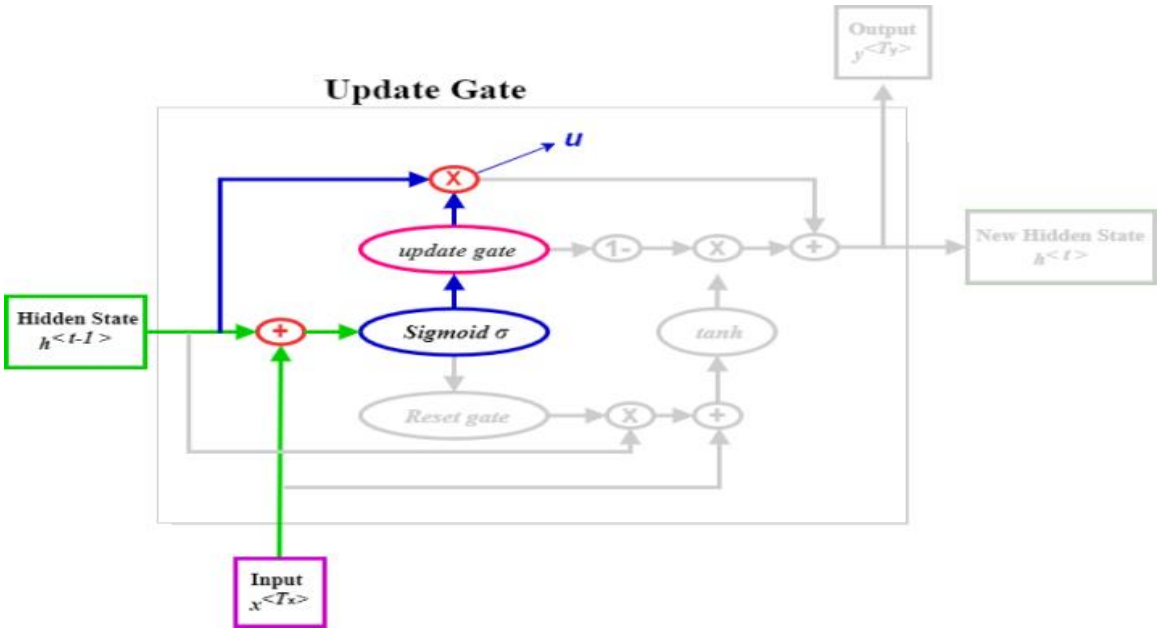


Figure 6.8 The update gate stream

For the calculation of the update gate, equation 10 is used. It is similar to equation 8 of the reset gate calculation the only difference is the unique weights used.

$$\boldsymbol{Update\ gate} = \boldsymbol{\sigma}(\boldsymbol{x}^{<t>}\boldsymbol{W}^{<xu>} + \boldsymbol{h}^{<t-1>}\boldsymbol{W}^{<hu>}) \quad (10)$$

The final output of the update gate$\boldsymbol{u}$ is calculated by equation 11. In this equation the update vector undergoes a an entry wise product multiplication with the previous hidden state $\boldsymbol{h}^{<t-1>}$to get $\boldsymbol{u}$which is essential value in the calculation of the final cell output as described in the next section 6.2.2.3.

$$\boldsymbol{u} = \boldsymbol{update\ gate} \odot \boldsymbol{h}^{<t-1>} \quad (11)$$

6.2.1.2.3 Final Output

The final output $\boldsymbol{y^{<T_y>}}$ and the new hidden state $\boldsymbol{h^{<t>}}$of the cell are calculated using the factors shown in Figure 6.9. The computation process is done by using an element-wise inverse of the update gate($\boldsymbol{1 - update\ gate}$)that undergoes an entrywise product multiplication with the rest gate output $\boldsymbol{r}$. This part of the operation is the one responsible for figuring out the portion of the new information to be saved in the hidden state $\boldsymbol{h^{<t>}}$. The last part of the computation is the sum of the earlier part with the output from the update gate . Equation (12) summarize the calculation of new hidden state $\boldsymbol{h^{<t>}}$.

$$\boldsymbol{h^{<t>} = r \odot (1 - update\ gate) + u} \tag{12}$$

This output $\boldsymbol{h^{<t>}}$ can be used as the final output for the time step $\boldsymbol{y^{<T_y>}}$ by passing it through a linear activation function.

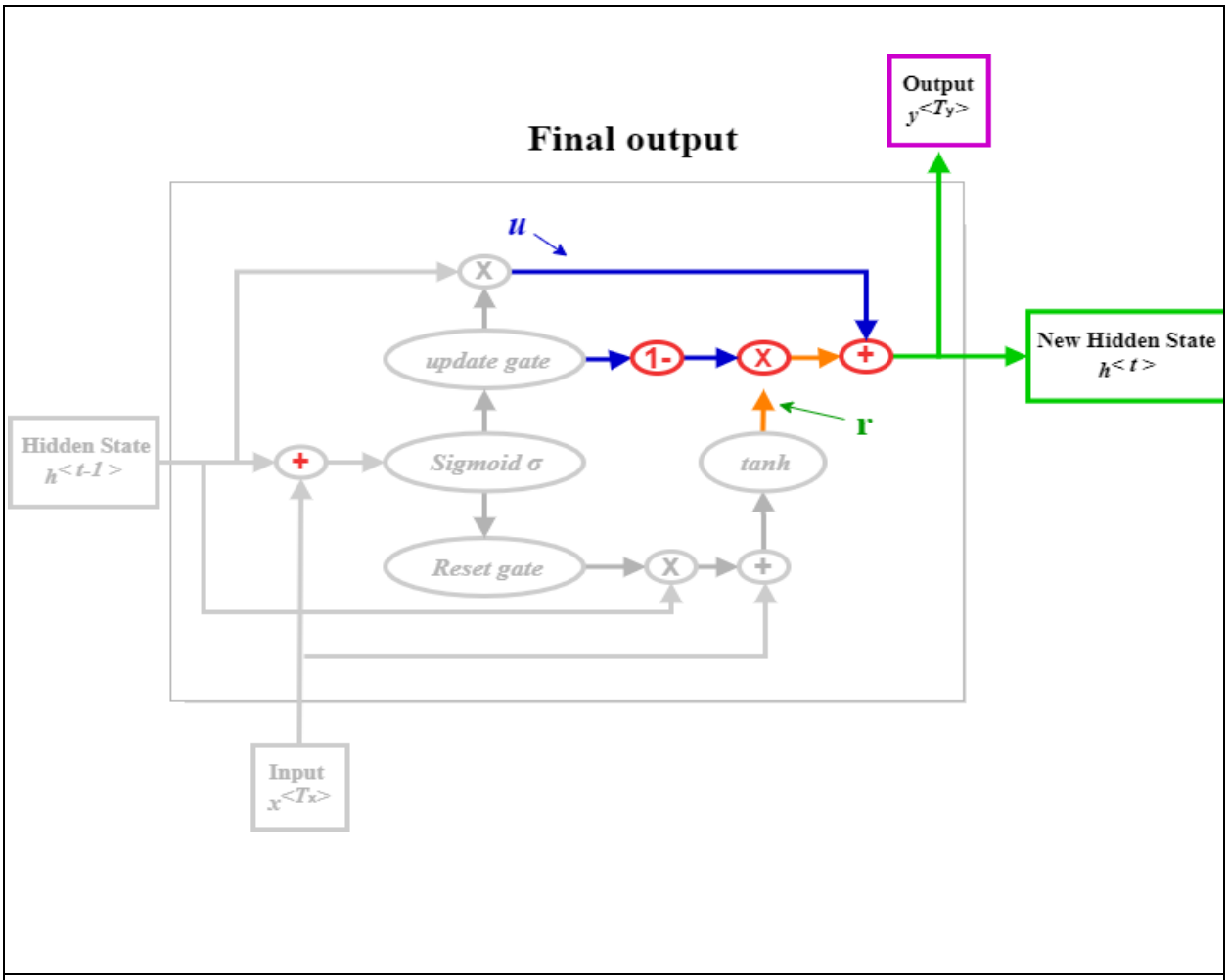


Figure 6.9 Final output calculation

6.2.1.2.4 Advantages of GRUs

As explained earlier, in section 6.2.1.2, GRU has a simple and efficient design offering number of advantages in favour of sequential data analysis and processing. One of these advantages that it can overcome the vanishing/exploding gradient problem of the RNN. It also improves the memory capacity as it controls the storage of information at both ends. It is also faster to compute and easier to train. Finally, it is suitable for various NLP tasks such as machine translation, sentiment analysis, and named entity recognition.

6.2.1.3 Bidirectional Gated Recurrent Units (BI-GRUs)

Bidirectional Gated Recurrent Unit (BI-GRU) is a sequence processing model that combines both BI-RNN and GRU together making use of the best of both worlds. As explained in section 6.1.2.1, BI-RNN has a forward and a backward hidden layers that are able to simultaneously utilize both past and future information. This bidirectionality of the RNN is joined with the GRU by replacing the RNN forward and backward hidden layers nodes with GRU cells as shown in Figure 6.10 which visualize the architecture of the BI-GRUs network.

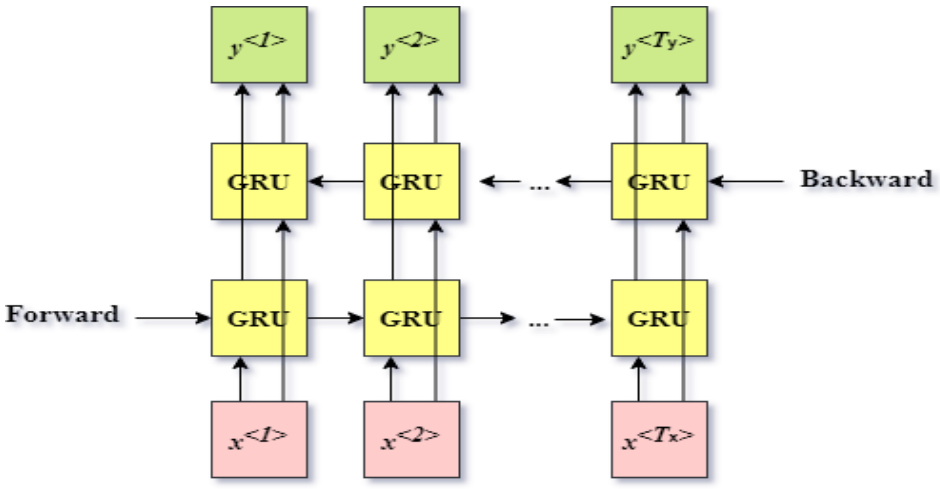


Figure 6.10 The architecture of the BI-GRUs network

In this model, there are two hidden layers of GRU that process input in both directions simultaneously as the sequential data is feed into the forward as well as the backward GRU layers at the same time (Che et al 2018). This allows the model to learn from previous as well as later data while dealing with the current data. Hence, current data is influenced by both past and future information (Liu et al 2021).

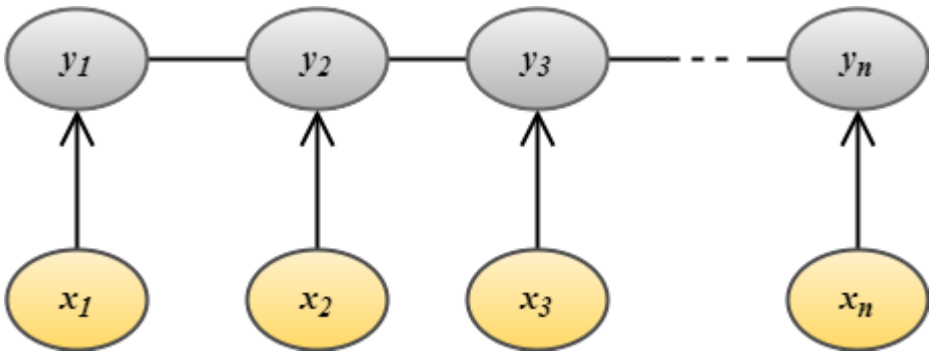
6.2.2 Conditional Random Fields classifier (CRF)

CRF is a statistical discriminative classifier that is used for sequence Modelling and prediction. It is used to recognize patterns within text and label them. In other words, it makes use of contextual information to identify words in the target text at the same time; it extracts features and learns patterns form input sequence to produce accurate predictions. As a discriminative classifier, CRF attempts to represent discriminative probability distribution as shown in equation (13)

$$\boldsymbol{P(y|x)} \tag{13}$$

Where $\boldsymbol{P}$is the probability, $\boldsymbol{x}$ is the input sequence, and$\boldsymbol{y}$ is the target sequence (output vector)

For Language processing, a linear chain CRF is commonly used as opposed to a general CRF [28] (Jurafsky and Martin 2000) Linear chain CRF applies sequential dependencies in the output as shown in Figure 6.11.



[28] Linear Chain CRF and General CRF are variants of CRF. For more information on these concepts refer: Charles Sutton and Andrew McCallum (2012), "An Introduction to Conditional Random Fields", Foundations and Trends® in Machine Learning: Vol. 4: No. 4, pp 267-373. http://dx.doi.org/10.1561/2200000013.

Figure 6.11 ***The linear chain CRF representation***

In CRF, features are extracted from data and modelled using the feature functions. These functions are the key to prediction$y$.They represent certain lexical attributes as well as contextual environment of the sequential variable x. Equation (14) represents CRF formula.

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} exp\left\{ \sum_{k=1}^{K} \theta_k f_k(Yt, Yt-1, x_t) \right\} \quad (14)$$

Equation 14 contains two components: normalization, weight and features. The first component, normalization, is represented by $Z(x)$. This part is responsible for converting the result to a probability that occurs between the intervals [0,1]. It is an input-dependent normalization constant which is expressed by equation (15)

$$Z(x) = \sum_{y} \prod_{t=1}^{T} exp\left\{ \sum_{k=1}^{K} \theta_k f_k(Yt, Yt-1, x_t) \right\} \quad (15)$$

The second component is weights $\theta_k f_k$ and the corresponding features $(Yt, Yt-1, x_t)$. The weights are usually estimated to match the features which are pre-defined. The Maximum Likelihood Estimation is used to produce weights. Having sequential data as input, the feature function can be defined using equation 16

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{i}, \boldsymbol{y}^{<i-1>}, \boldsymbol{y}^{<i>}) \quad (16)$$

In equation 16 f represents the feature function where x is a set of input vectors, i is the position of data under investigation, $y^{<i-1>}$ is the lable of the data at the position $i-1$ in x and $y^{<i>}$ is the lable of data in position i in x. For example, $f(x, i, y^{<i-1>}, y^{<i>})$ =1 if $y^{<i-1>}$ is a VB (main verb) and $y^{<i>}$ is a NNP (proper noun) otherwise 0. As seen in the example, the context of the data plays fundamental role in defining feature functions which equal 0 or 1.

6.2.3 Bidirectional Gated Recurrent Units Conditional Random Fields (BI-GRUs-CRF) Network

The BI-GRUs-CRF model is built using a combination of the BI-GRUs and the CRF classifier which are introduced earlier in this Chapter, section 6.2.1.3, and 6.2.2 respectively. The CRF layer is included as a hidden layer that takes the output from the BI-GRUs layer as its input. This model can produce the optimal tagging sequence as BI-GRUs layer extracts the past and future contextual information and it feed as the features into the CRF layer which predicts the optimal tagging sequence. Figure 6.12 shows the architecture of the BI-GRUs-CRF model.

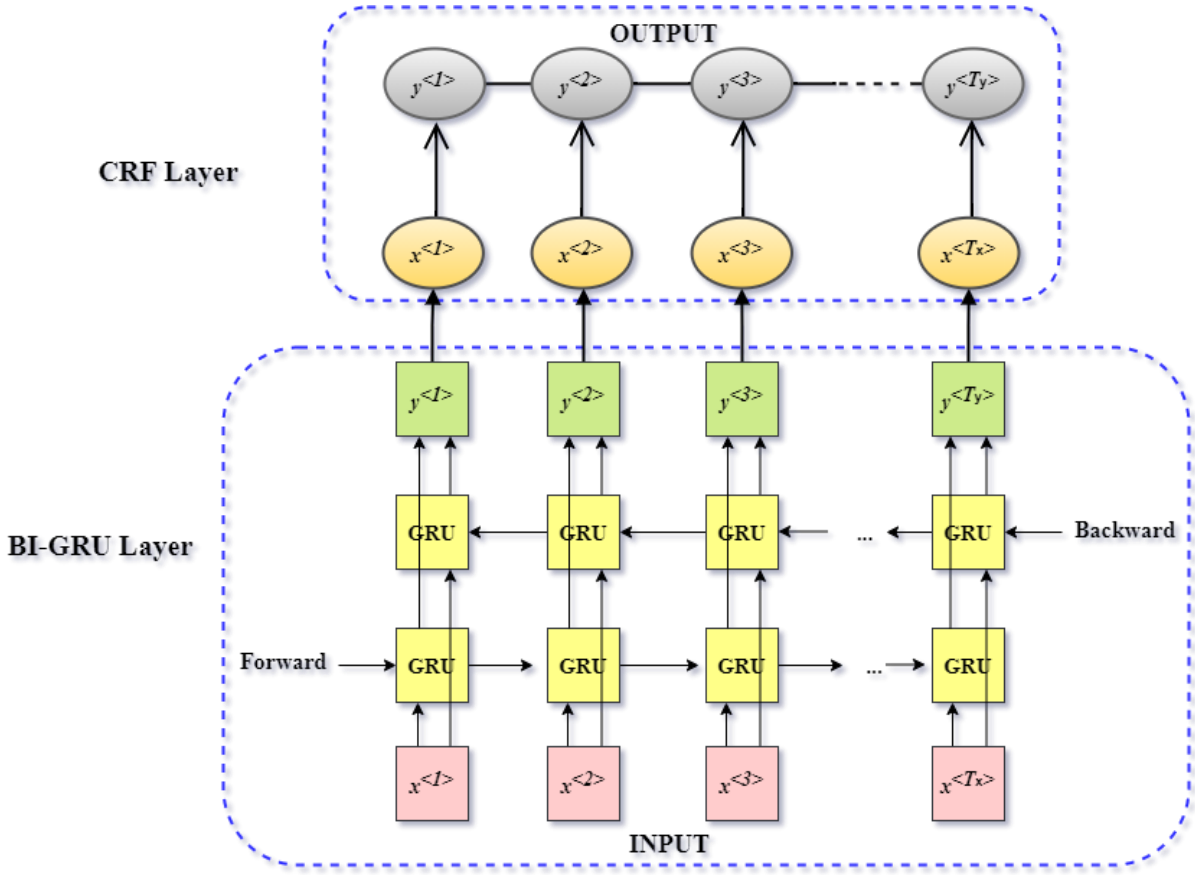


Figure 6.12 The architecture of BI-GRUs-CRF Model

6.2.4 The pipeline of the BI-GRUs-CRF Tagger

After defining the models listed in section 6.2 above, the pipeline of our project, which is visualized in Figure 6.16, could be summarized in five major steps:

- Data loading
- Word Embedding
- BI-GRUs-CRF building
- Model Training

- Predictions Making

a. Data loading:

We mean by data loading reading, copying and preparation of data sequences to be loaded as a machine-readable input. To do so, we worked on three main sub-functions described as follows:

1. First, we ensure that data is clean by defining a function for data cleaning.
2. Second, we read the .xlsx file that contains two columns, one for words and the second for word tag, considering those sentences are separated by a blank cell.
3. The function **load ()** returns two lists, one for tokenized sentences and the other for equivalent sentence tags. In total, we have 23 distinct tags and 11,163 sentences.

b. Word Embedding:

Our input is a text that has to be converted into a numerical type and then fed into our machine-learning model. In this context, we load the vector representation of words in Arabic. In other words, we want to collect the maximum number of features that cover all the characteristics of given words. In fact, the "**FastText**"[29] model was trained on Wikipedia texts to find the most accurate representation that minimizes the distance between words that are semantically similar and share the same characteristics. We load those vectors using **"word2vec.KeyedVectors.load_word2vec_format"** function under genism library. Therefore, each word in our corpus is represented by a vector of size 300.

[29] "FastText" model is an extension of the "word2vector" model, which targets word embedding. It has an advantage over the word2vector that allows it to deal with OOV tokens and embed them using n-grams instead of direct vectors.

c. BI-GRUs-CRF building

After loading the data, preparing the textual sequences (words and their corresponding tags) and their numerical representations, we move to the Deep learning model implementation. The input contains a list of sentences of different sizes; however, a deep learning model requires samples of a single tensor input that are masked to be shorter than the longest item. The masking is done by padding the data by adding zeros. After masking the input length to get a consistent length, the machine is informed of the padded data to be neglected. Figure 6.13 shows the model building code.

```
def build_model():
    crf_layer = CRF(23)
    input_layer = Input(shape=(None, 300,))
    mask_layer = Masking(mask_value=0., input_shape=(140, 300))(input_layer)
    bi_gru = Bidirectional(GRU(10, return_sequences=True))(mask_layer)
    bi_gru = TimeDistributed(Dense(10, activation="relu"))(bi_gru)
    output_layer = crf_layer(bi_gru)
    return Model(input_layer, output_layer), crf_layer
```

Figure 6.13 Model building and masking code

- CRF (23): refers to the CRF layer we are going to add at the end of the BI-GRU model, 23 represents the number of distinct tags in our tagset.
- Input(shape=(None,300,)): means that the input of our model will be a vector of size 300.
- Masking (mask_value=0., input_shape=(140, 300))(input_layer): 140 represents the maximum sentence size.
- Bidirectional (GRU(10, return_sequences=True))(mask_layer): 10 refers to the size of hidden units.

- TimeDistributed(Dense(10, activation="relu"))(bi_gru): We applied the relu activation function to the output of the BI-GRUs model. It consists of setting negative values to zero and keeping the positive ones to their values to avoide the vanishing gradiant problem.
- crf_layer(bi_gru): We add the CRF layer at the end of the architecture as explained above.

d. Model training:

The BI-GRUs-CRF model was trained using our social media-based corpus of San'ani Arabic described in Chapter Four and the adapted tagset described in Chapter Five, Section 5.1.3. We chose to train the model using the "**rmsprop**" optimizer since it automatically updates the learning rate. The learning rate is a scalar representing the speed of convergence to the optimal solution. Figure 6.14 presents model training code.

```
train_model, crf_layer = build_model()
train_model.compile(optimizer="rmsprop", loss=crf_layer.loss_function, metrics=[crf_layer.accuracy])
train_model.summary()

# train model
history = train_model.fit(np.array(train_x), np.array(train_y),
          epochs=20, verbose=1, validation_data=(np.array(test_x), np.array(test_y)))
```

Figure 6.14 ***Model Training code***

The CRF_layer.loss function, from the training code, can be defined as the score of the real path and the score of all the possible baths. Equation 17 calculates the ***Loss Function,*** where P refers to "the probability of". Mathematically, the score of the real bath has to be the highest; as it keeps maximizing during the training of the BI-GRUs-CRF model. It is due to the constant updating of the model parameters values. Here path refers to the ordered tags of words.

$$\boldsymbol{Loss\ Function} = \frac{\boldsymbol{P_{RealPath}}}{\boldsymbol{P_1 + P_2 + \ldots + P_n}} \quad (17)$$

The model was trained using 20 epochs, after which no notable increment was seen in training.

e. Predictions making:

After training the model using 20 epochs; as we can notice after each epoch the loss decreases and the accuracy increases, we get the final and optimal weights matrix. We save those results/weights to make predictions on other unseen sentences. Those weights are saved using the code shown in Figure 6.15.

```
train_model.save_weights('weights.hd5f')
```

Figure 6.15 ***The code for saving weights***

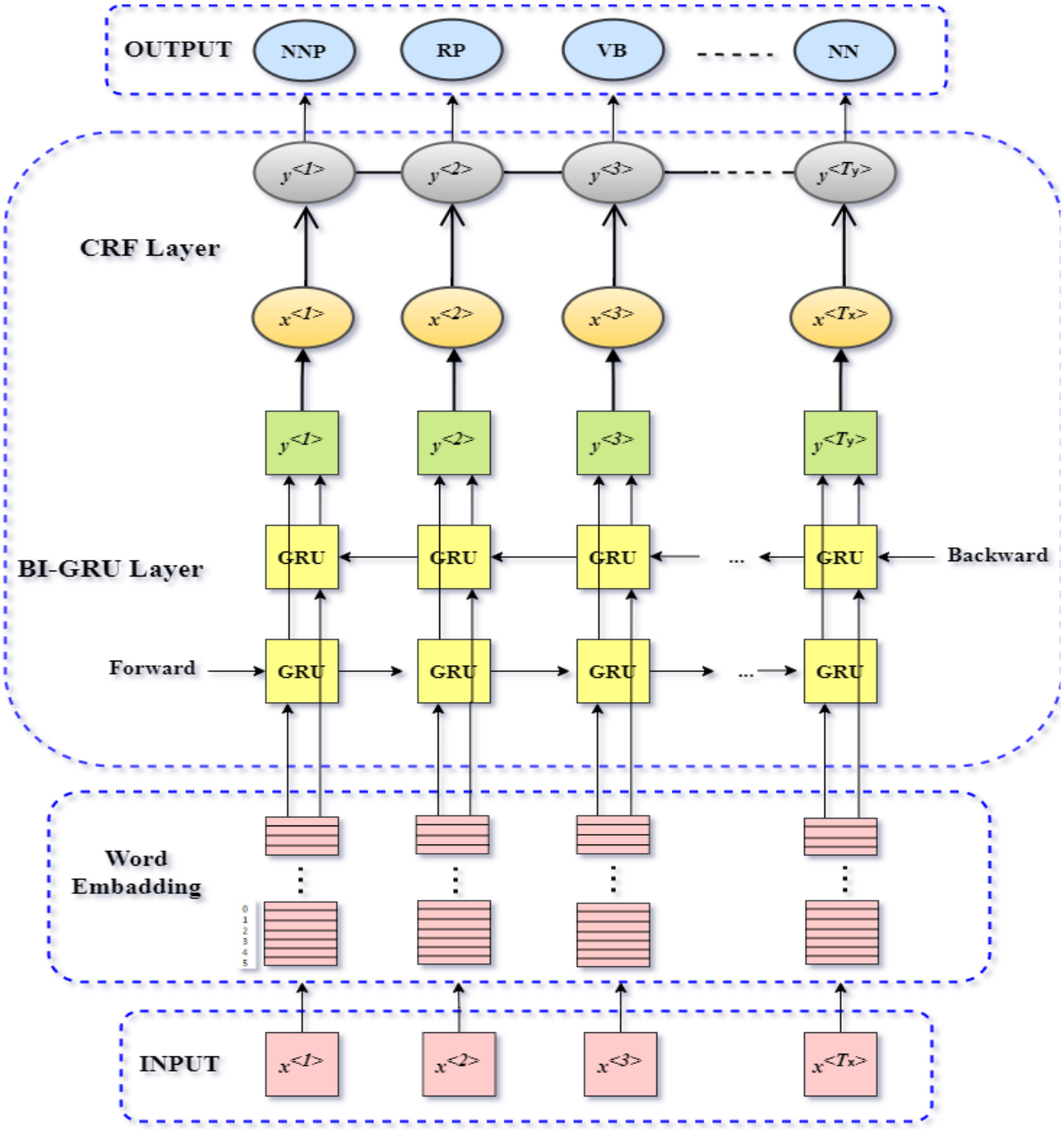


Figure 6.16 The pipeline of the BI-GRUs-CRF Tagger

6.3 The Graphical user interface (GUI) of the San'ani Arabic parts-of-speech Tagger

The GUI of our tagger was developed using the Django web framework. As shown in Figure 6.17, there are four GUI widgets on the GUI. The first is the "**Load vec file**" used to load

the vectors before text insertion. The **Raw input** widget is the second. It is an entry box where the end-users are supposed to enter the data. The **Submit** widget is the third which is below the entry box of the Raw input. After insetting the input in the entry box, it is to be pressed to submit the data and generate the output. Then the output appears in a table of two columns; the first column contains the tokenized words under the heading **Word,** and the second column shows the tags under the **Prediction.** Each token and its corresponding tag appear in a separate row. The fourth widget is the **Download** which gives the option of downloading the output in a .pdf format document. Appendix A presents the tagger's code and Appendix B shows sample of the output.

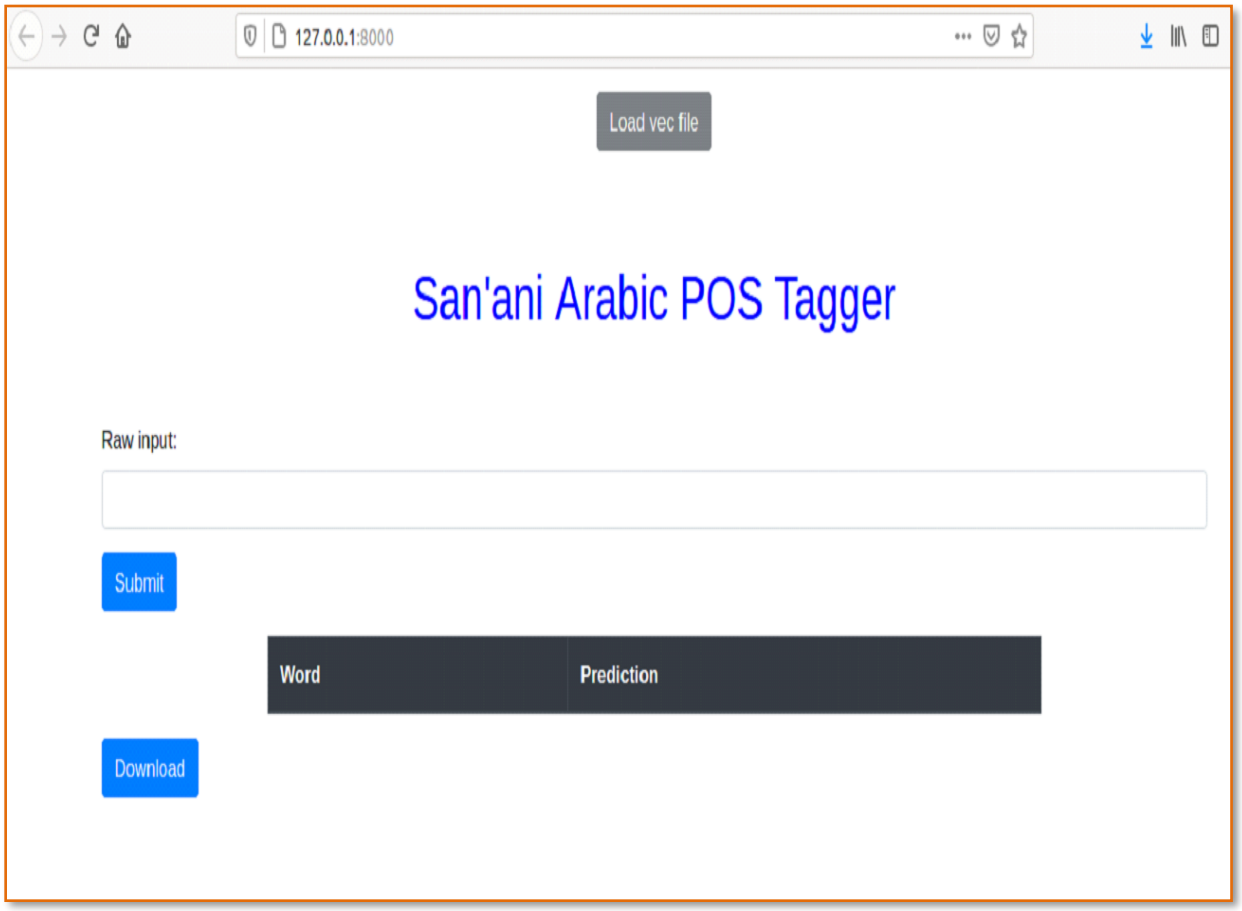


Figure 6.17 The GUI of the San'ani Arabic POS Tagger

6.4 Summary

This chapter describes the BI-GRUs-CRF tagging model of San'ani Arabic and its graphical user interface. It begins with an introduction justifying the selection of the BI-GRUs-CRF model. Then background manifestations of the RNN and three RNN variants, namely, the BI-RNN, the GRUs, and the BI-GRUs, are included. Also, the CRF classifier is displayed in isolation as it is an essential component of our model. After that, our model is described based on the earlier components' presentation.

Additionally, the project pipeline is demonstrated in detail. The pipeline consists of five main stages: Data loading, Word Embedding, The BI-GRUs-CRF model building, Model Training, and Predictions Making. Finally, we describe the GUI of the tagging system.

CHAPTER SEVEN
TAGGER EVALUATION (RESULTS AND DISCUSSION)

7.1 Introduction

Many evaluation methods are available for testing parts-of speech taggers, such as the accuracy measures, the average tagging perplexity, the f-measure, and the average ambiguity (Paroubek 2007). However, we chose to use the accuracy measure, which accounts for the success rate. The significant advantages of this measure are that it reflects how the tagger handles the data and allows a detailed error analysis. But since our San'ani Arabic Tagger is a novel tool, no gold standard test reference is available for testing. So we composed our own test sets abiding by (Paroubek 2007) conditions which states that the test reference must be segmented with the same conventions used by the tagger, and the annotation tagset must be the same which was the case of our test data.

The following sections of this chapter describe the evaluation of the BI-GRUs-CRF tagger. There are four main sections: the results, errors analysis, discussion, and chapter summary.

7.2 Results

To evaluate our tagger, we prepared two test sets. These test sets are the same source as the training corpus, i.e., social media platforms,[30]; however, they are new unseen data to the tagger. They are:

- 5,641 tokens were collected from a novel posted on Facebook and Telegram[31] platforms titled *ʕawdat ʔalma:dˤi:* "The return of the past," which is written by Meme Abdialgaleel. It will be referred to as the first test set henceforth.
- 5,635 tokens were collected from a romantic novel posted on Facebook and Telegram platforms titled *ʕaru:s sanʕa:* "The bride of Sana'a," written by Abeer Al Kebsi. It will be referred to as the second test set henceforth.

We have chosen the testing sets based on the availability of the data, source of data, domain which is the same as the training data, and variety of the data, which is mainly San'ani Arabic. Our data source is Facebook and Telegram, as these platforms are the most popular in Yemen, as explained in Chapter Four, Section 4.3.2. Additionally, since dialectal Arabic does not have a conventional writing system, the writers' writing style could influence the text writing[32].

[30] For more information on the source of data, refer to Chapter Four, Section 4.3.2
[31] The Facebook page link is https://www.facebook.com/RewayatMeme/ and the Telegram Channel is https://telegram.me/qesasSanani
[32] See Chapter Four, Section 4.4.2

To calculate the results of the first and second test sets, formulas (1) and (2) are used. Then equation (3) is used to figure the overall result of the tagger.

$$\textbf{General result of each test set} = \frac{\textbf{Count of Correctly taged words}}{\textbf{Count of all the words}} \times 100 \quad \textbf{(1)}$$

$$\textbf{Correct tags'rate} = \frac{\textbf{Count of Correctly taged words of a single tag}}{\textbf{Count of correctly tagged words}} \times 100 \quad \textbf{(2)}$$

$$\textbf{Overall result} = \frac{\textbf{Sum of the percentage score of 1st and 2nd test sets}}{2} \quad \textbf{(3)}$$

The following subsections present a detailed result of each test set. Additionally, the test statistics are provided for each test set in detail.

7.2.1 The first test set

This test set is very similar to the training data writing style as it was written by one of the authors of the training data. Also, the test set is dominantly written in San'ani Arabic dialect; i.e., MSA is kept at a minimum.

The BI-GRUs-CRF tagger correctly tagged 88.1% of this data. Figure 7.1 visualizes the tagging accuracy of the first test set. The result of this test set is higher than the second one. Out of the accuracy rate, i.e., 88.1%, VB (verb) tag scored 24% as the highest, followed by NN

(noun) tag, which scored 21%. The NNP (proper noun) tag represents 5%, indicating that the data domain is the same as the training data. Figure7.2 presents the correct tags' rate in the first test set.

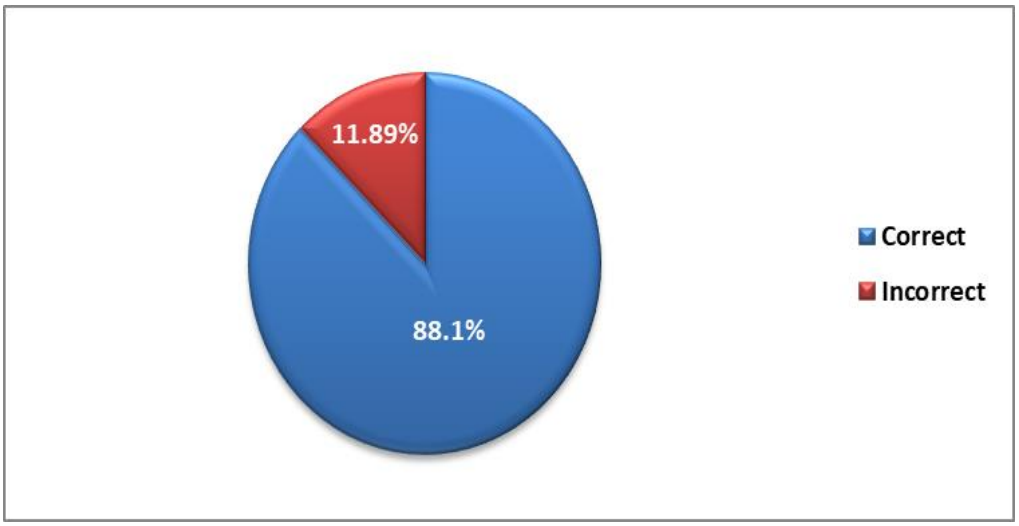


Figure 7.1 The general result of the first test set

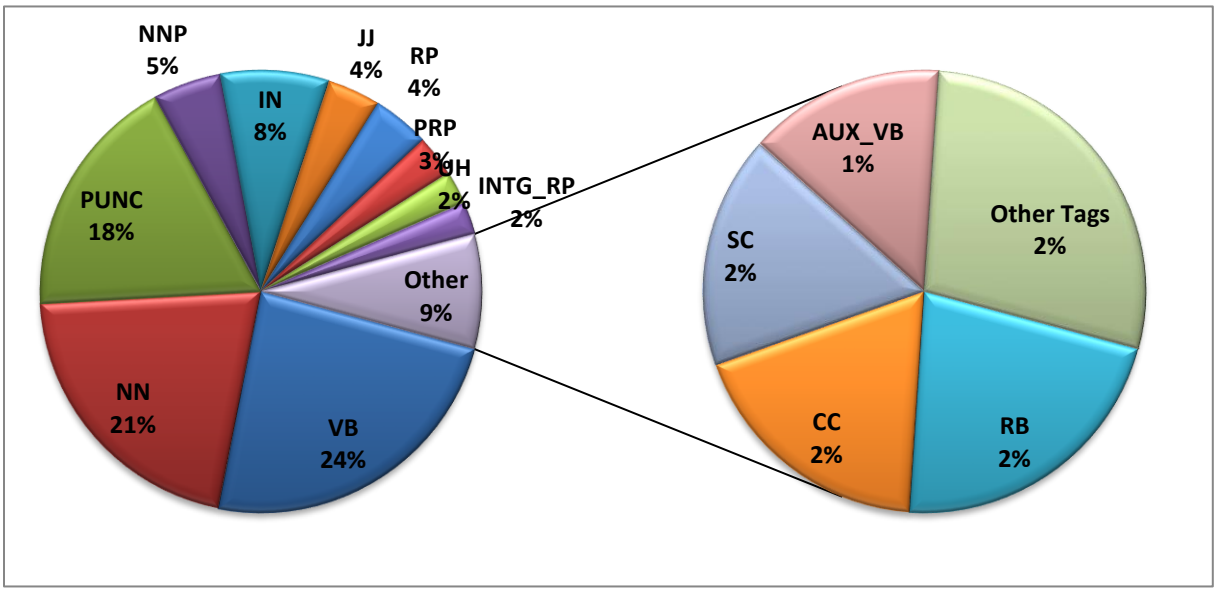


Figure 7.1 The Correct tags' rates in the first test set

A deeper look at the detailed statistics of the first test set, proofs that the first test set is the same domain as the training set as 94% of the NN (noun) tag and 88% of the VB (verb) tag are tagged accurately. Table 7.1 contains the detailed result statistics of each tag within the first test set.

Table 7.1 *The detailed results of the first test set*

Tags	Frequency	Correct Count	Incorrect Count	Correct Percentage %	Incorrect Percentage %
VB	1347	1186	161	88.04	11.95
NN	1109	1043	66	94.04	5.95
PUNC	890	890	0	100	0
NNP	535	248	287	46.35	53.64
IN	396	392	4	98.98	1.01

JJ	234	192	42	82.05	17.94
RP	230	209	21	90.86	9.13
PRP	149	149	0	100	0
UH	139	119	20	85.61	14.38
INTG_RP	120	117	3	97.5	2.5
RB	118	91	27	77.11	22.88
CC	96	77	19	80.20	19.79
SC	82	80	2	97.56	2.43
AUX_VB	67	60	7	89.55	10.44
D_PRP	36	36	0	100	0
OD	25	23	2	92	8
WP	18	18	0	100	0
FCD	17	17	0	100	0
WRB	12	8	4	66.66	33.33
FW	9	3	6	33.33	66.66
DT	7	7	0	100	0
CD	3	3	0	100	0
SYM	2	2	0	100	0
TOTAL	**5641**	**4970**	**671**	**88.1049**	**11.8951**

7.2.2 The second test set

Though the second test set is the same domain as the training set, it contains mixed data, namely MSA text. In fact, 10%; i,e,. 554 tokens of this test data are written in MSA. Moreover, the count of foreign words (FW) is quite higher than the first test set.

After running the second test set through our BI-GRUs-CRF tagger, the accuracy rate dropped to 83.56%, which is 4% lower than the first test set. Figure 7.1 shows the general result of the second test set in a pie chart.

The detailed statistics of the accuracy rate show that the representation of the correct tags is somewhat similar to the first test representation. As shown in Figure 7.4, the highest correct tag is the VB tag which represents 24%. Then NN tag is the second highest with 23.7%, followed by the IN (preposition) tag with 10%. The NNP rate represents 3.9% of the correct percentage wheel.

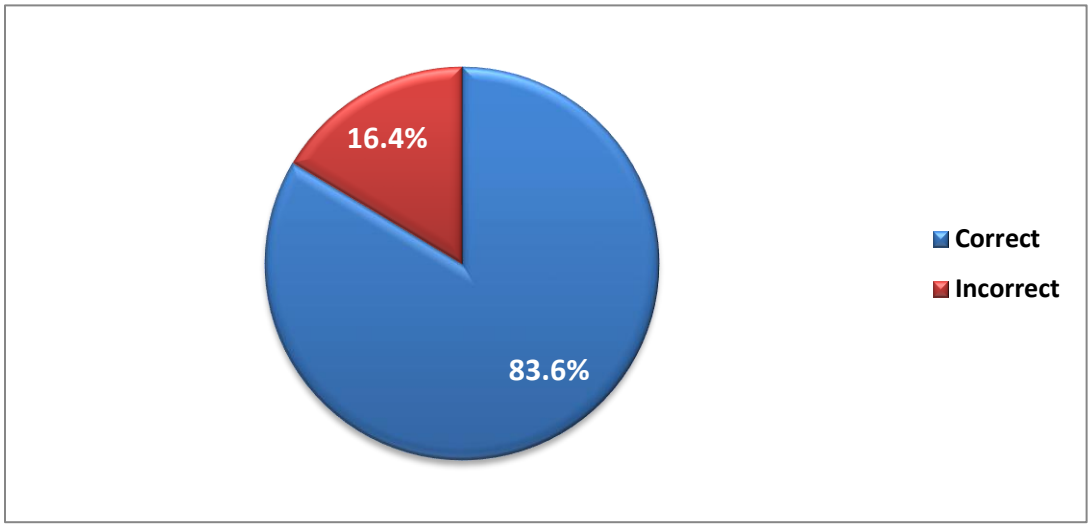


Figure 7.2 The general result of the second test set

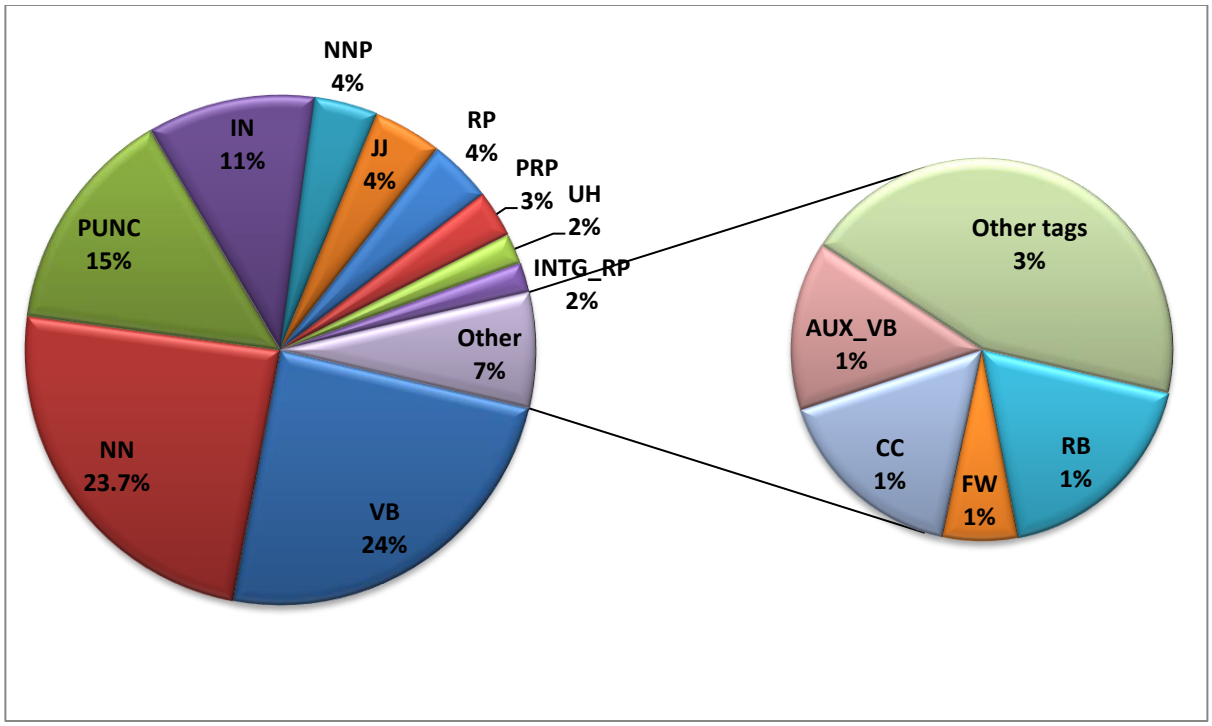


Figure 7.3 The Correct tags' rates in the second test set

A closer look to the detailed result of the second test set, which is presented in Table 7.2, reveals new types of errors in certain function word tags; i.e., CD, WP and D_PRP which are not present in the results of the first test set. These errors can be traced back to MSA rather than San'ani Arabic.

Table 7.2 *Detailed result of the second test set*

Tags	Frequency	Correct Count	Incorrect Count	Correct Percentage %	Incorrect Percentage %
VB	1,390	1,145	245	82.37	17.62
NN	1,348	1,137	211	84.34	15.65
PUNC	677	677	0	100	0
IN	528	500	28	94.69	5.30
NNP	455	189	266	41.53	58.46
JJ	241	205	36	85.06	14.93
RP	202	190	12	94.05	5.94
PRP	140	140	0	100	0
UH	104	90	14	86.53	13.46
INTG_RP	101	89	12	88.11	11.88
RB	79	63	16	79.74	20.25
FW	78	22	56	28.20	71.79
CC	69	57	12	82.60	17.39
AUX_VB	54	49	5	90.74	9.25
SC	51	48	3	94.11	5.88
D_PRP	37	35	2	94.59	5.40
WP	31	30	1	96.77	3.22
FCD	17	17	0	100	0
CD	10	8	2	80	20
WRB	11	9	2	81.81	18.18
OD	7	7	0	100	0
DT	5	5	0	100	0
TOTAL	**5635**	**4712**	**923**	**83.62**	**16.37**

7.2.3 The Overall result

Taking into consideration the results of both test sets, i.e., first and second, the overall accuracy of the tagger is 85.86%. Figure 7.5 visualize the accuracy of the first and second test sets

compared to the overall accuracy. The overall rate is calculated using formula (3) introduced in Section 7.1.

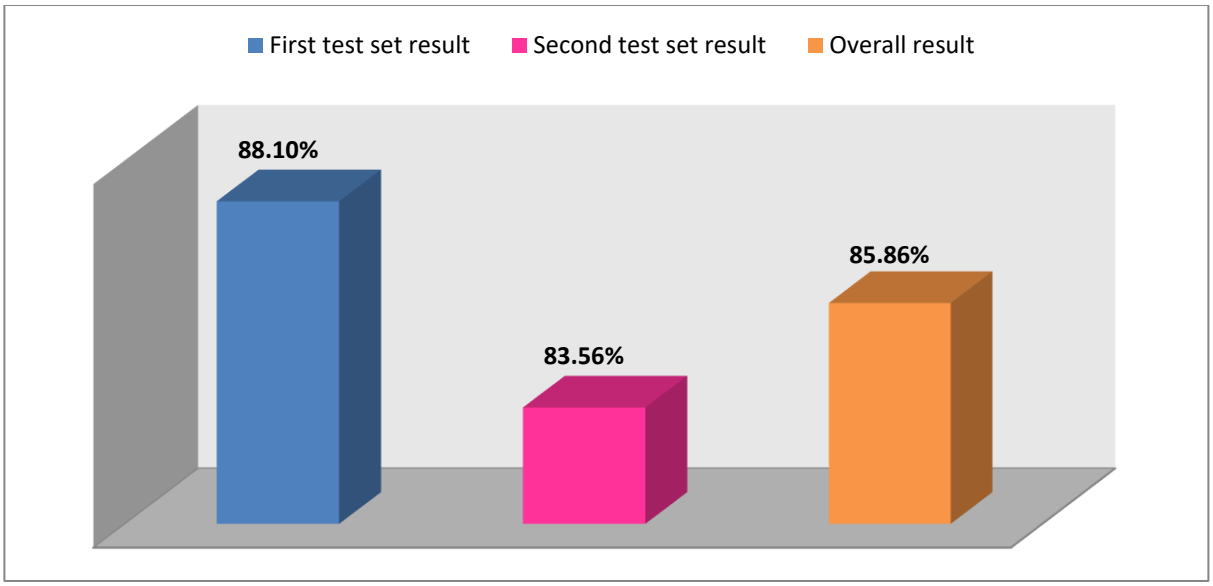


Figure 7.4 *The overall accuracy*

7.3 Errors analysis

The errors within the first and second test sets are listed in Figures 7.6 and 7.8, respectively. In both Figures, the tag to the left of the arrow represents the actual (correct) tag while the one to the right of the arrow is the assigned (incorrect) tag. For instance, the error type NN→NNP means that NN is the correct tag while NNP is the incorrect tag. Additionally, in the exact figures, the number in front of each bar represents the frequency of the corresponding error types.

On analysing the first test set error types, we notice that 59% of the errors in the first test set are tagged as NN (common noun), 21% as VB (verb), 5.96% as RP (particle), and 2.38% as NNP (proper noun). The rest of the error types' percentages of the first test set are shown in Figure 7.7. Similarly, the statistics of the second test set error types represented in Figure 7.9 shows that 56.33% of the errors are tagged as NN, 27% as VB, 4% as RP (particle), and 3.79% as NNP.

Considering both test sets, the most frequent error type is NN which scores more than 55%. The VB is the second-highest, followed by RP and NNP, respectively. However, the rate of VB error in the second test set is higher than in the first set, which might be explained by the mixed variety of the second test set. The following section discusses error types providing conclusions regarding causes of errors.

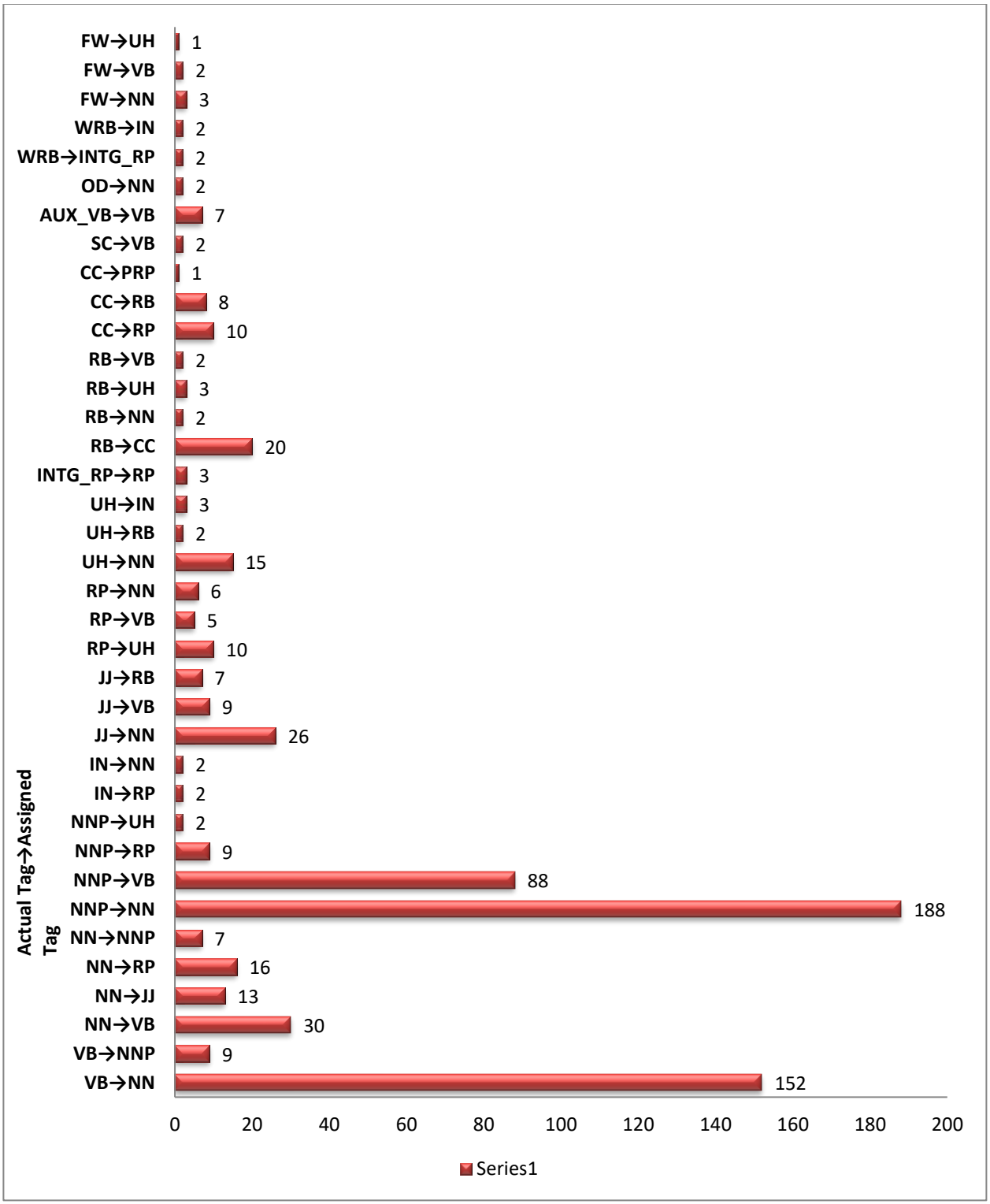


Figure 7.5 The error types in the first test set

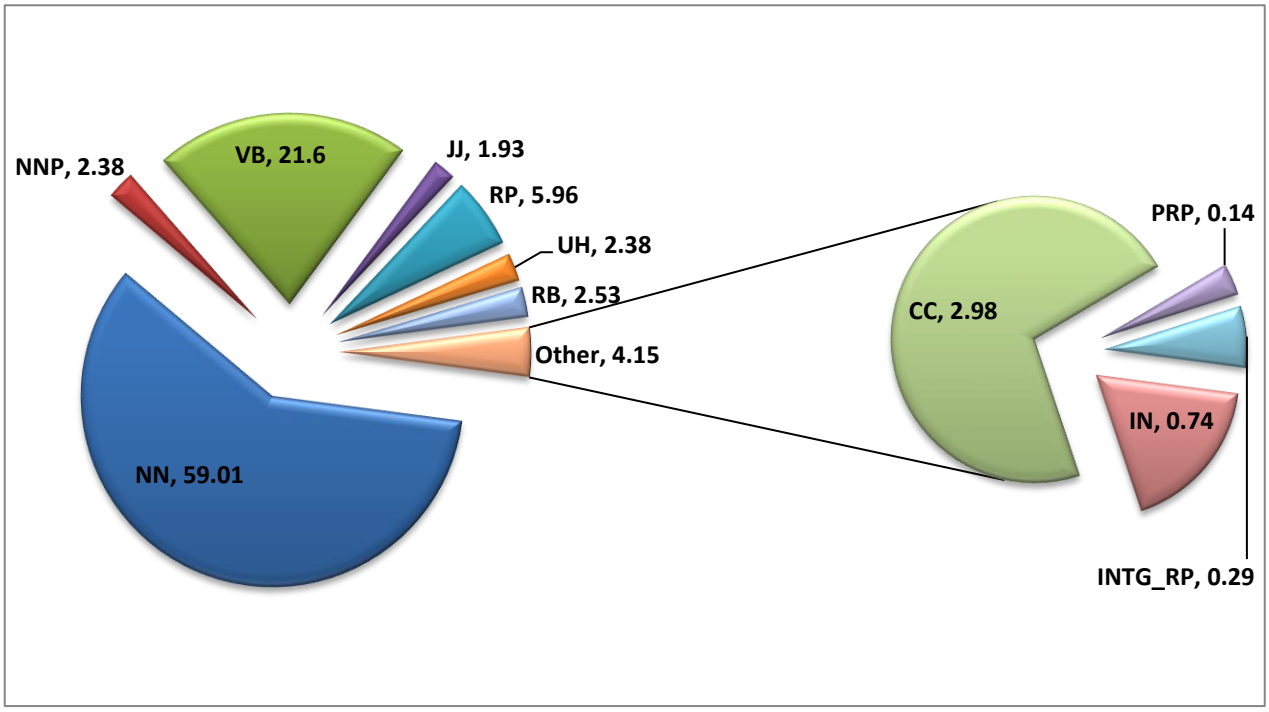


Figure 7.6 The percentage of the error types in the first test set

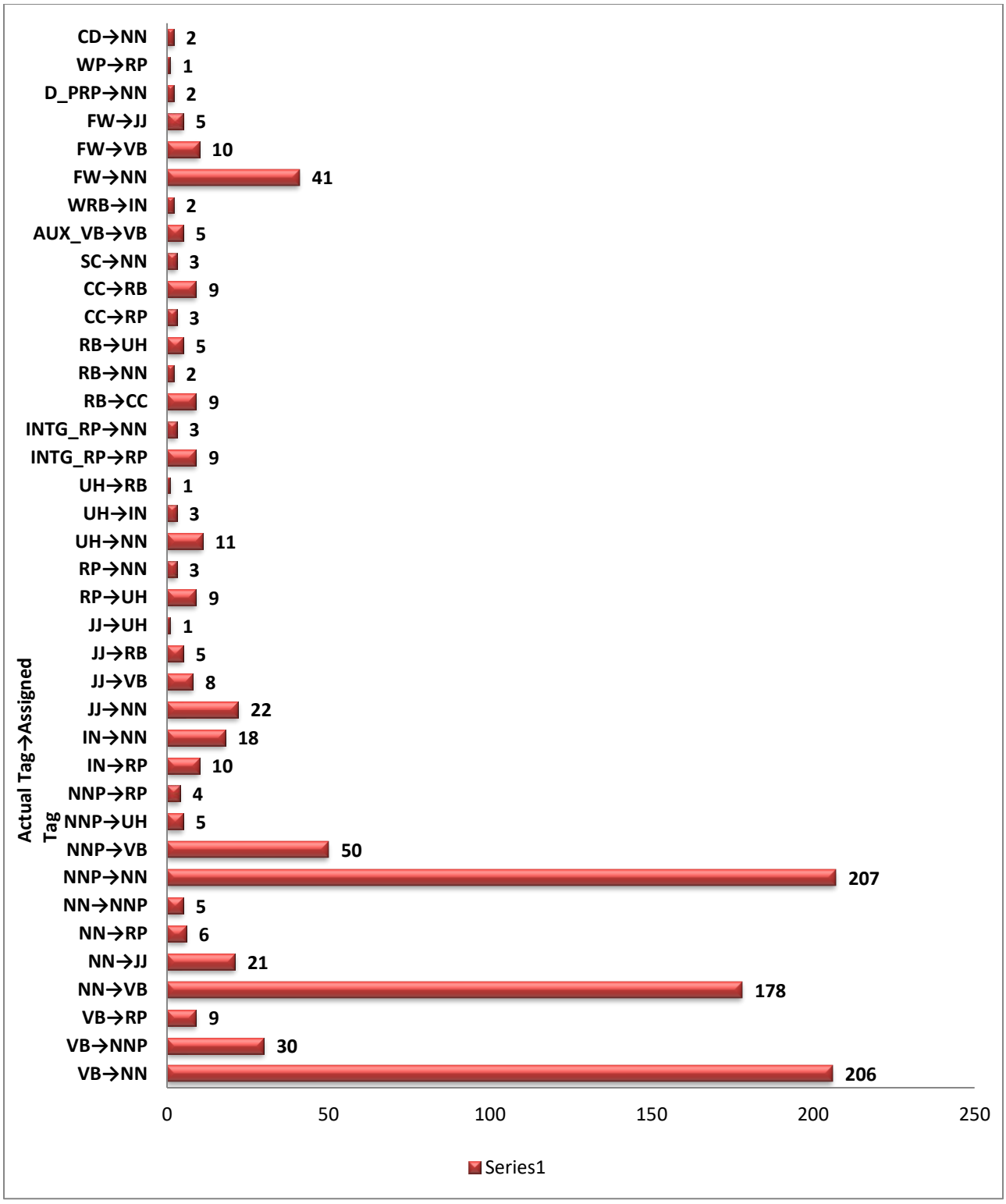


Figure 7.7 The error types in the second test set

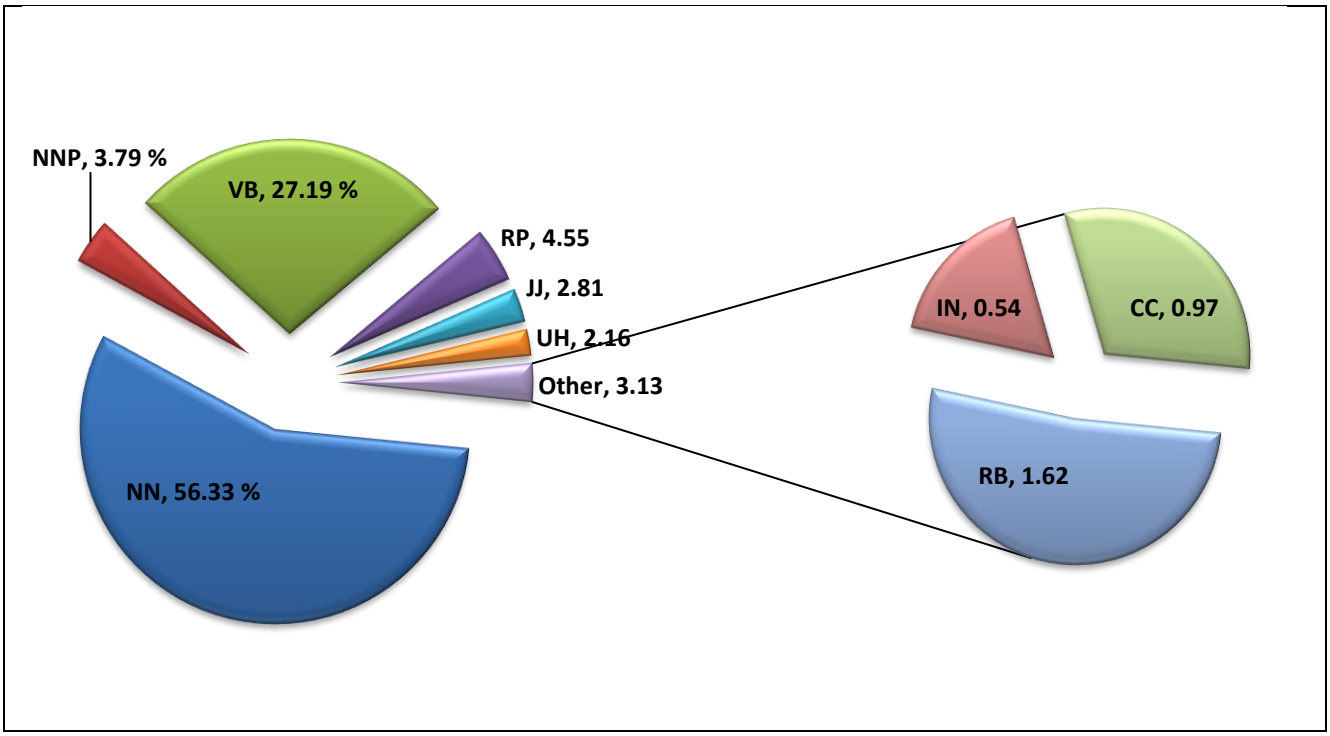


Figure 7.8 The percentage of the error types in the second test set

7.4 Discussion

The causes of errors differ with the test set being used. However, we can state that most of the time, proper nouns NNP are incorrectly tagged as nouns NN. In fact, in both test sets, error analysis indicates that the number of NN is the highest, which can be related to the frequency of nouns in the training corpus. Other errors result from some spelling variations that are not present in the training data. Additionally, the limited size of the unique words in the training corpus, which is 24k, is another factor of errors.

Another cause of errors is related to the mixed data available in the second test set, which constitutes 10 % of the test data size. Moreover, the non-functionality of punctuation marks and the absence of short vowels (diacritics) from the text influenced the output negatively.

In spite of the above, the accuracy rate of the tagger, i.e., 85.89% is considered reasonable taking into consideration the data type which is social media data.

7.5 Summary

This chapter evaluates the BI-GRUs-CRF parts-of-speech tagger describing the test sets' results and the errors analysis. We chose the accuracy measure to perform the evaluation. The overall accuracy of the tagging system is 85.86%. The first text set registered the highest accuracy rate of 88.10 % and the second test set reported 83.56%. Most of the errors reported were tagged as NN (common nouns), which is related to the accuracy of common nouns in the training data. Chapter Seven has five main sections: 7.1 introduction, 7.2 results, 7.3error analysis, 7.4 discussion, and 7.5 summary.

CHAPTER EIGHT
CONCLUSIONS AND FUTURE WORK

8.1. Conclusions

At the beginning of this research, when we approached the area of Arabic computational linguistics, we directly noticed that the undivided attention of research was given to MSA and occasionally Classical Arabic. The main bulk of research placed MSA in the spotlight, neglecting other Arabic forms. However, during the last decade, attention expanded gradually towards Arabic dialects. The main motivation for such change was the prevalence of dialectal interchange through social media platforms.

However, work on dialectal Arabic is still in elementary stages due to several challenges, including the paucity of data and resources. Such challenges, among others, influence the selection of the dialect/s to work on it/them. As a matter of fact, Egyptian, Gulf, and Levantine dialects are primarily targeted while other Arabic dialects are barely touched. Yemeni Arabic dialects, including San'ani Arabic, are some of these dialects that are in dire need of NLP research. This motivated us to work on San'ani Arabic dialect, developing a fundamental resource, a parts-of-speech tagger.

In this chapter, subsection 8.1.1 describes how the main aim and objectives are achieved. Then section 8.2 discusses the research contribution. Section 8.3 presents the research limitations, and section 8.4 suggests future works and recommendations.

8.1.1 Aim and Objectives

The aim and objectives of this thesis, which were introduced in Chapter One, section 1.3, were successfully accomplished. The main aim of this research work was to develop an automatic parts-of-speech tagger for tagging San'ani Arabic. To fulfil this primary aim, we had to meet the following specific objectives:

- Constructing a grammatically annotated social media corpus of San'ani Arabic of at least 200k tokens for training.

One of the novel contributions of this research is the development of a grammatically annotated San'ani Arabic social media corpus as described in chapters IV and V. This corpus was developed to train our parts of speech tagger. It consists of more than 200k tokens and 24k types. Chapter Four describes the process of developing the corpus in terms of raw data selection, collection, and pre-processing. The data pre-processing was an essential step to overcome the challenging noisy social media data. Hence, data pre-processing was performed in two steps: data cleaning and text normalization. After cleaning the data, we had to take care of the text normalization, so we ended up setting normalization standards for San'ani Arabic social media text. The following step in pre-processing is tokenization. Two types of tokenization were performed: word and sentence tokenization. After that, the corpus was statistically analysed.

- Adapting a suitable Arabic tagset to perform the data annotation and parts-of-speech tagging. (see Chapter Five)

This objective was achieved in Chapter Five, where the tagset adoption and manual annotation process were described. Benefitting from the examination of the available parts-of-

speech tagset for Arabic conducted in Chapter Three Section 3.3, the Bies tagset was adopted. Therefore, tagset justification and description are presented. Furthermore, the parts-of-speech/grammatical annotation process is introduced, followed by statistical analysis for the grammatically tagged corpus.

- Building and training a deep learning-based parts-of-speech tagger using the BI-GRUs-CRF model.

The development of a parts-of-speech tagger for San'ani Arabic is the ultimate aim of this work. Chapter Six and Seven reveal that our aim has been successfully accomplished. Throughout this chapter, we explain the architecture and functionalities of the Bidirectional-Gated Recurrent Units-Conditional Random Fields model components. The project pipeline was introduced in detail. In addition, we developed a user-friendly graphical user interface. The user interface allows for a smoother experience providing the user the options of either copying or downloading the output in a portable document format (pdf). Chapter Seven tests the tagger and reports the accuracy levels. The overall accuracy of the system reached 85.86%.

- Evaluating the Part-of Speech tagger output using testing data of extra 11k tokens

This objective was met as described in Chapter Seven. The accuracy measure was used to perform the evaluation. Two test sets of 11k tokens of new/unseen data were prepared and tested compared to the tagger output. The first and second test sets reported 88.1% and 83.56 accuracy rates, respectively. The overall result reaches 85.8%.

8.2 Research Contributions

The novel contributions of this research are as follows:

- A grammatically annotated corpus of San'ani Arabic social media text

The availability of data is an essential requirement in building our tagger. However, as illustrated in Chapter Three's literature review, few San'ani Arabic dialect data resources are available. Hence, we had to develop our corpus of San'ani Arabic, utilizing open data sources.

Therefore, one of the novel contributions of this research is the development of a grammatically annotated San'ani Arabic social media corpus as described in chapter Four and Five. We selected and collected our data from popular social media platforms in Yemen, namely, Facebook and Telegram. The corpus size surpasses 200k tokens training corpus and 11k tokens testing corpus. After the collection of data, data pre-processing and grammatical annotation took place. Since our data is social media-based, further pre-processing is needed to remove noise and standardize ill-formed data. The data pre-processing includes three critical techniques: noise cleaning, data tokenization (word and sentence tokenization), and text normalization. All the three pre-processing stages were applied to the data systematically. Then the pre-processed corpus was enriched with parts-of-speech tags. The data annotation process abides by Leech's (1993) maxims of corpus annotation. The annotation was conducted manually and by a native speaker of San'ani Arabic following the guidelines of the Penn Arabic Treebank (PATB) (Maamouri et al. 2008).

- Standardization guidelines for San'ani Arabic social media text

  In corpus development, we had to work on data pre-processing, overcoming certain challenges related to the social media data nature and the non-conventional dialectal orthography. Therefore, we had to work on text normalization, setting standardization guidelines for San'ani Arabic social media text. Chapter Four, Section 4.4.2, describes these guidelines clearly.

- An automatic parts-of-speech tagger for San'ani Arabic

The ultimate goal of this thesis was to develop a parts-of-speech tagger for San'ani Arabic. Chapter Six reveals that our aim has been successfully accomplished. We developed an automatic San'ani Arabic parts-of-speech tagger using the BI-GRUs-CRF model. The tagger achieves an overall accuracy of 85.8%, which is reasonable. We believe that the accuracy rate can be increased by expanding the training corpus's size and utilizing morphological information.

8.3 Review of Research Questions

4. Does San'ani Arabic NLP resources exist?

As seen in Chapter Three, the literature review, few NLP tools were developed for San'ani Arabic dialect. Most Arabic tools and resources are built primarily for MSA and CA. Though some attention was directed to some Arabic dialects in the last decade, San'ani Arabic is one of the less fortunate dialects that are still hiding in the shadows.

5. Are there any Arabic NLP resources or tools that can benefit dialectical/San'ani Arabic tagging? More specifically,

    a- Which tag set available for MSA parts-of-speech tagging can be adapted for San'ani Arabic parts-of-speech tagging?

    The findings of Chapter Three, Section 3.4 and Chapter Five, Section 5.2, and Chapter Six show that the Bies/LDC/RTS tagset was successfully adapted for San'ani Arabic parts-of-speech tagging. It was used to annotate the training data explained in chapter five and the test

data introduced in chapter seven. It is also the tagset fed into the BI-GRUs-CRF tagging model to perform tagging.

b- Is there a reasonable size corpus of San'ani Arabic text? And if so, is it enriched with parts-of-speech annotation?

The answer to this question was illustrated in Chapter Three, section 3.4, which clearly shows that there is no reasonable size corpus of San'ani Arabic. The only San'ani Arabic corpus reported is (Al-Shargi et al. 2016), consisting of 33k. This corpus is small in size, and the major part of it is a transcription of spoken data rather than written text. Moreover, the corpus URL link is broken; hence it is ineffectual.

6. Can BI-GRUs-CRF model be used to develop an efficient part-of speech tagger for San'ani Arabic?

Chapters Seven and eight answer this question distinctly. The BI-GRUs-CRF model was successfully adopted to build and train a parts-of-speech tagger for San'ani Arabic using our developed corpus of 200k tokens for training and 11k tokens for evaluation. The tagger evaluation reported 88.1% on the first test set and 83.56% on the second. The overall accuracy of the tagger reached 85.8%.

8.4 Research Limitations

The limitations of our research include:

- Our tagger is trained to tag San'ani Arabic text. Hence other Yemeni Arabic dialects and other Arabic forms would have a low-quality output.

- The domain of our training data is limited only to fictional novels, out of which the main themes are drama, romance, and tragedy. We wanted to include other domains and genres such as news, sports, and tourism; unfortunately, it was not possible as there are few San'ani Arabic electronic data.
- Compared to other MSA corpora, our corpus is limited in size as it consists of only 200k words.
- Since our adopted tagset is coarse, our corpus is limited for specific NLP applications that do not require detailed morphological information.

8.5 Future Work

Future research can be extended in many ways, some of which:

- Our tagger was trained using only 200k tokens of San'ani Arabic social media data. We believe that increasing the training data size can improve the accuracy of the tagger. Additionally, utilizing other San'ani resources can positively influence the output.
- Since the BI-GRUs-CRF model is successfully applied in Part-of-Speech tagging, further investigation is required to utilize this model in developing other NLP resources; for instance, it can be utilized in performing other sequence labelling tasks, such as Named Entity Recognition (NER) and text chunking.
- Our corpus introduced in Chapters Four and Five can be targeted for future research enhancing the corpus verities and genres. Moreover, it can be enriched with other annotation types, such as morphological features.

References

Aboul-Fetouh and Hilmi Mohamed. 1959. *The Plural Morpheme of Egyptian Arabic Nouns*. M.A. Thesis. Texas University.

Aboul-Fetouh and Hilmi Mohamed. 1969. *A Morphological Study of Egyptian Colloquial Arabic.* The Hague: Mouton.

Abumalloh, Rabab Ali, Hassan Maudi Al-Sarhan, Othman Ibrahim, and Waheeb Abu-Ulbeh. 2016. "Arabic part-of-speech tagging." *Journal of Soft Computing and Decision Support Systems*, Vol. 3 (2): 45-52. Retrieved from http://jscdss.com/index.php/files/article/view/94

Abumalloh, Rabab Ali, Hasan Muaidi Al-Serhan, Othman Bin Ibrahim, and Waheeb Abu-Ulbeh. 2018. "Arabic Part-of-Speech Tagger, an Approach Based on Neural Network Modelling." *International Journal of Engineering & Technology*, 7(2.29): 742-746.

Alabbas, Maytham, and Allan Ramsay. 2012. "Improved POS-tagging for Arabic by combining diverse taggers." In: Iliadis L., Maglogiannis I., Papadopoulos H. (eds) Artificial Intelligence Applications and Innovations. AIAI 2012. IFIP Advances in Information and Communication Technology, vol. 381. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-33409-2_12

Albared, Mohammed, Nazlia Omar, Mohd Juzaiddin Ab Aziz, and Mohd Zakree Ahmad Nazri. 2010. "Automatic Part of Speech Tagging for Arabic: An Experiment Using Bigram Hidden Markov Model." In: Yu J., Greco S., Lingras P., Wang G., Skowron A. (eds) Rough Set and Knowledge Technology. RSKT 2010. Lecture Notes in Computer Science, vol 6401: 361-370. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-16248-0_52

Al-Batal, Mahmoud. 1994. "Connectives in Arabic diglossia: the case of Lebanese Arabic" Amsterdam Studies in The Theory and History of Linguistic Science Series 4: 91-91. Amsterdam. https://doi.org/10.1075/cilt.115.10alb

Albogamy, Fahad, and Allan Ramsay. 2015. "POS tagging for Arabic tweets." In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, 1-8. https://aclanthology.org/R15-1001.pdf

AlGahtani, Shahib, William Black, and John McNaught. 2009. "Arabic part-of-speech tagging using transformation-based learning." In *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt: 66-70. http://www.elda.org/medar-conference/pdf/43.pdf

Ali, Bilel Ben, and Fethi Jarray. 2013. "Genetic approach for Arabic part of speech tagging." *International Journal on Natural Language Computing (IJNLC).* Vol. 2(3). https://arxiv.org/abs/1307.3489

AlKhwiter, Wasan, and Nora Al-Twairesh. 2020. "Part-of-speech tagging for Arabic tweets using CRF and Bi-LSTM." *Computer Speech & Language*, Vol. 65. https://doi.org/10.1016/j.csl.2020.101138

Almeman, Khalid, and Mark Lee. 2013. "Automatic building of Arabic multi dialect text corpora by bootstrapping dialect words." *2013 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, 1-6, https://ieeexplore.ieee.org/document/6487247

Alosaimy, Abdulrahman, and Eric Atwell. 2017. "Tagging classical Arabic text using available morphological analysers and part of speech taggers." *Journal for Language Technology and Computational Linguistics*, Vol. 32(1) (2017): 1-26. ISSN 2190-6858 https://eprints.whiterose.ac.uk/126376/#:~:text=pp.%201%2D26.-,ISSN%202190%2D6858,-Abstract

Al-qrainy, Shihadeh, and Aladdin Ayesh. 2006. "Developing a tagset for automated POS tagging in Arabic." WSEAS Transactions on Computers. Vol. 5(11): 2787−92. https://dora.dmu.ac.uk/bitstream/handle/2086/1002/ShihadehAlqrainy-Ayesh-WSEAS-CONFERENCE.pdf?sequence=3

Alqrainy, Shihadeh, Hasan Muaidi AlSerhan, and Aladdin Ayesh. 2008. "Pattern-based algorithm for part-of-speech tagging Arabic text." In *2008 International Conference on Computer Engineering & Systems*,119-124. IEEE. https://ieeexplore.ieee.org/abstract/document/4772979

Alqrainy, Shihadeh. 2008. "A Morphological-Syntactical Analysis Approach for Arabic Textual Tagging." Leicester, UK, De Montfort University. PhD, 197. https://dora.dmu.ac.uk/handle/2086/4819

Al-Sabbagh, Rania, and Roxana Girju. 2012. "A supervised POS tagger for written Arabic social networking corpora." In *KONVENS*, Vol. 5: 39-52. Vienna, Austria.

Al-Shamsi, Fatma, and Ahmed Guessoum. 2006. "A hidden Markov model-based POS tagger for Arabic." In *Proceeding of the 8th International Conference on the Statistical Analysis of Textual Data*, France, 31-42.

Al-Shargi, Faisal, and Owen Rambow. 2015. "DIWAN: A dialectal word annotation tool for Arabic*." In Proceedings of the Second Workshop on Arabic Natural Language Processing*, 49-58. https://aclanthology.org/W15-3206.pdf

Al-Shargi, Faisal, Aidan Kaplan, Ramy Eskander, Nizar Habash, and Owen Rambow. 2016. "A Morphologically Annotated Corpus and a Morphological Analyzer for Moroccan and Sanaani Yemeni Arabic." In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia: European Language Resources Association (ELRA) https://aclanthology.org/L16-1207

Alshargi, Faisal, Shahd Dibas, Sakhar Alkhereyf, Reem Faraj, Basmah Abdulkareem, Sane Yagi, Ouafaa Kacha, Nizar Habash, and Owen Rambow. 2019. "Morphologically Annotated Corpora for Seven Arabic Dialects: Taizi, Sanaani, Najdi, Jordanian, Syrian, Iraqi and Moroccan." In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, 137-147. https://aclanthology.org/W19-4615.pdf

Alshutayri, Areej, and Eric Atwell. 2019. “A social media corpus of Arabic dialect text.” *Computer-Mediated Communication and Social Media Corpora. Clermont-Ferrand: Presses Universitaires Blaise Pascal*: 1-23.

Al-Taani, Ahmad T., and Salah Abu Al-Rub.2009. “A rule-based approach for tagging non-vocalized Arabic words.” *The International Arab Journal of Information Technology*, Vol. 6(3): 320-328. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.182.9109&rep=rep1&type=pdf

Al-Toma, Salih J. 1969. The Problem of Diglossia in Arabic: A Comparative Study of Classical and Iraqi Arabic. Cambridge,Mass: Harvard University Press.

American Heritage® Dictionary of the English Language, Fifth Editions’. "arabization." Retrieved June 28 2021 from https://www.thefreedictionary.com/arabization

Arberry, A. J. 1957. The Seven Odes. London: George Allen and Unwin.

Aronoff, Mark. 1976. Word Formation in Generative Grammar. Cambridge: MIT Press.

Aronoff, Mark. 1994. “Morphology by Itself: Stems and Inflectional Classes.” Language 70 (4): 811–17. https://doi.org/10.2307/416331.

Awwalu, Jamilu, Saleh El-Yakub Abdullahi, and Abraham Eseoghene Evwiekpaefe. 2020. “Parts of speech tagging: a review of techniques.” *Fudma Journal of Sciences* Vol. 4 (2): 712-721. https://pdfs.semanticscholar.org/054e/455943b6789b98cc748c825196f447d19e14.pdf?_ga=2.259298190.521384528.1638683928-1871479276.1631173298

Bahl, L. R., and Mercer, R. L. 1976. “Part of speech assignment by a statistical decision algorithm.” In *IEEE International Symposium on Information Theory*. 88-89. https://www.bibsonomy.org/bibtex/2992cbd29a2f93894022358b03a2e5ecd/nlp

Barbara B. Greene, Gerald M. Rubin. 1971. *Automatic Grammatical Tagging of English.* Providence: Department of Linguistics, Brown University.

Baum, Leonard E. 1972. “An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes.” *Inequalities.* Vol. 3, (1): 1-8. https://files.library.northwestern.edu/public/Files/Baum.pdf

Beeston, A. F. L. 1970. The Arabic language today. Taylor & Francis Ltd. ISBN: 9781138698987, 9781138698987

Benello, Julian, Andrew W. Mackie, James A. Anderson. 1989. “Syntactic category disambiguation with neural networks.” *Computer Speech & Language.* Vol. 3(3): 203-217. https://doi.org/10.1016/0885-2308(89)90018-1

Biadsy, Fadi, Julia Hirschberg, and Nizar Habash. 2009. “Spoken Arabic dialect identification using phonotactic modeling.” In Proceedings of the eacl 2009 workshop on computational approaches to semitic languages, pp. 53-61.

Bouamor, Houda, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid et al. 2018. "The madar arabic dialect corpus and lexicon." In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*. https://aclanthology.org/L18-1535.pdf

Brants, Thorsten. 2000. "TnT-a statistical part-of-speech tagger." In *Proceedings of the Sixth Applied Natural Language Processing Conference ANLP-2000*. Seattle, WA. *https://arxiv.org/pdf/cs/0003055.pdf*

Brill, Eric. 1992. "A simple rule-based part of speech tagger." In *Proceedings of the Third Conference on Applied Natural Language Processing*. 152-155. Trento, Italy.

Brill, Eric. 1994. "Some advances in transformation-based part of speech tagging." In *Proceedings of the Twelfth International Conference on Artificial Intelligence (AAAI94*). Seattle, Washington. https://arxiv.org/abs/cmp-lg/9406010

Brill, Eric, and Mihai Pop. 1999. "Unsupervised Learning of Disambiguation Rules for Part-of-Speech Tagging." In: Armstrong S., Church K., Isabelle P., Manzi S., Tzoukermann E., Yarowsky D. (eds) *Natural Language Processing Using Very Large Corpora. Text, Speech and Language Technology*, vol. 11: 27-42, Springer, Dordrecht. https://doi.org/10.1007/978-94-017-2390-9_3.

Btoush, Mohammad Hjouj, Abdulsalam Alarabeyyat, and Isa Olab. 2016. "Rule based approach for Arabic part of speech tagging and name entity recognition." *International Journal of Advanced Computer Science and Applications*, 7(6): 331-335. https://pdfs.semanticscholar.org/35e2/528713fdd2e1e4cf75aefe3dc98bac428912.pdf

Buckwalter, Tim. 2002. "Buckwalter Arabic Morphological Analyzer version 1.0." *LDC2002L49.* Philadelphia: Linguistic Data Consortium, University of Pennsylvania https://doi.org/10.35111/7vzm-mb15

Carstairs-McCarthy, Andrew. 1994. "Inflection Classes, Gender, and the Principle of Contrast." Language 70 (4): 737–88. https://doi.org/10.2307/416326.

Charfi, Anis, Wajdi Zaghouani, Syed Hassan Mehdi, and Esraa Mohamed. 2019. "A fine-grained annotated multi-dialectal Arabic corpus." In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, Varna, Bulgaria: INCOMA Ltd. 198-204. https://aclanthology.org/R19-1023

Church, Kenneth Ward. 1989. "A stochastic parts program and noun phrase parser for unrestricted text." In *International Conference on Acoustics, Speech, and Signal Processing*. Vol. 2: 695-698. doi:10.1109/ICASSP.1989.266522

Cutting, Doug and Julian Kupiec and Jan Pedersen and Penelope Sibun. 1992. "A practical part-of-speech tagger." In *Third Conference on Applied Natural Language Processing*. 133-140. https://aclanthology.org/A92-1018.pdf

Darwish, Kareem, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T. Al-Natsheh, Houda Bouamor, Karim Bouzoubaa, et al. 2021. "A Panoramic Survey of Natural Language Processing in the Arab World." *Communications of the ACM* 64 (4): 72–81. https://doi.org/10.1145/3447735 .

DeRose, Steven J. 1988. “Grammatical category disambiguation by statistical optimization.” *Computational linguistics*. Vol. 14(1): 31-39. https://dl.acm.org/doi/abs/10.5555/49084.49087

Diab, Mona, Kadri Hacioglu, and Dan Jurafsky. 2004. “Automatic tagging of Arabic text: From raw text to base phrase chunks.” In *Proceedings of HLT-NAACL 2004: Short papers*. 149-152. https://aclanthology.org/N04-4038.pdf

Diab, Mona. 2007. “Improved Arabic base phrase chunking with a new enriched POS tag set.” In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, Prague, Czech Republic: Association for Computational Linguistics. 89-96. https://aclanthology.org/W07-0812.pdf

Diab, Mona. 2009. “Second generation AMIRA tools for Arabic processing: Fast and robust tokenization, POS tagging, and base phrase chunking.” In *2nd International Conference on Arabic Language Resources and Tools*, vol. 110. Columbia University, New York, NY. https://www2.seas.gwu.edu/~mtdiab/files/publications/refereed/50.pdf

Diab, Mona, Nizar Habash, Owen Rambow, and Ryan Roth. 2013. “LDC Arabic treebanks and associated corpora: Data divisions Manual.” Version 1. Technical Report No. CCLS-13-02. Center for Computational Learning Systems-CCLS, Columbia University. https://arxiv.org/abs/1309.5652

Diab, Mona, Mohamed Al-Badrashiny, Maryam Aminian, Mohammed Attia, Heba Elfardy, Nizar Habash, Abdelati Hawwari, Wael Salloum, Pradeep Dasigi, and Ramy Eskander. 2014. "Tharwa: A large scale dialectal arabic-standard arabic-english lexicon." In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*: 3782-3789.

Duh, Kevin, and Katrin Kirchhoff. 2005. “POS tagging of dialectal Arabic: a minimally supervised approach.” In *Proceedings of the ACL workshop on computational approaches to Semitic languages*, 55-62.

ElHadj, Y.O.M., I.A. AlSughayeir, A.M. Khorsiand A.M. Alansari, 2009. “Morphology analysis ofthe Holy Quran: An indexed Quran text database (in Arabic).” *Proceeding of the 5th International Conference on Computer Sciences Practice in Arabic*, Rabat, Morocco: 72-84.

Elhadj, Yahya. 2009. “Statistical part-of-speech tagger for traditional Arabic texts.” *Journal of computer science*, Vol. 5(11): 794-800. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.8858&rep=rep1&type=pdf

El-Haj, Mahmoud, and Rim Koulali. 2013. “KALIMAT a multipurpose Arabic Corpus.” In *Second workshop on Arabic corpus linguistics (WACL-2)*, Vol. 2: 22-25. https://eprints.lancs.ac.uk/id/eprint/71282/1/KALIMAT_ELHAJ_KOULALI.pdf

Freeman, Andrew. 2001. “Brill’s POS tagger and a morphology parser for Arabic.” In *Proc. of ACL Workshop on Arabic Language Processing*. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.21.2717&rep=rep1&type=pdf

Gahbiche-Braham, Souhir, Hélene Bonneau-Maynard, Thomas Lavergne, and François Yvon. 2012. “Joint Segmentation and POS Tagging for Arabic Using a CRF-based Classifier.” In LREC, 2107-2113.

Garside, Roger. 1987. "The CLAWS word-tagging system." In *The Computational Analysis of English: a corpus-based approach*, edited by R. Garside, G. Leech and G. Sampson, 30-41. London: Longman. https://www.bibsonomy.org/bibtex/2108d2bd04674266ed7630df43dc0cec8/cbrewster

Ghaneim, K. (2014). Arabization Mechanisms and New Industry Terminology, the Palestinian Arabic Language Academy,Gaza

Giménez, Jesus, and Màrquez, Lluıs. 2004. "A general pos tagger generator based on support vector machines." In *Proceedings of the 4th LREC Conference*. *https://www.cs.upc.edu/~nlp/SVMTool/SVMTutorial.v1.2.2.pdf*

Gimpel, Kevin; Schneider, Nathan; O'Connor, Brendan; Das, Dipanjan; Mills, Daniel; Eisenstein, Jacob; Heilman, Michael; Yogatama, Dani; Flanigan, Jeffrey; and Smith, Noah A. (2010). *Part-of-speech tagging for twitter: Annotation, features, and experiments*. (Technical rept.) Carnegie-Mellon Univ Pittsburgh Pa, School of Computer Science. https://apps.dtic.mil/sti/pdfs/ADA547371.pdf

Graja, Marwa, Maher Jaoua, and L. Hadrich Belguith. 2010. "Lexical study of a spoken dialogue corpus in Tunisian dialect." In *The international Arab conference on information technology (ACIT),* Benghazi–Libya. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.452.7847&rep=rep1&type=pdf

Guiasu, Silviu, Shenitzer, Abe. 1985. "The principle of maximum entropy." *The Mathematical Intelligencer* 7(1): 42–48. https://doi.org/10.1007/BF03023004

Güngör, Tunga. 2010. "Part-of-speech tagging." In: Nitin Indurkhya, Fred J. Damerau (*eds*) *Handbook of Natural Language Processing*, 2nd edn, Chapman & Hall/CRC, Boca Raton, FL, 205–235. https://www.taylorfrancis.com/chapters/mono/10.1201/9781420085938-19/part-speech-tagging-tunga-g%C3%BCng%C3%B6r-nitin-indurkhya-fred-damerau?context=ubx

Habash, Nizar, and Owen Rambow. 2005. "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop." In *Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05*), 573-580.

Habash, Nizar, and Owen Rambow. 2006. "MAGEAD: A Morphological Analyzer and Generator for the Arabic dialects." In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, 681-688. Sydney, Australia: Association for Computational Linguistics. https://doi.org/10.3115/1220175.1220261

Habash, Nizar. 2007. "Arabic morphological representations for machine translation." In Abdelhadi Soudi, Antal van den Bosch, and Guenter Neumann, editors, *Arabic Computational Morphology: Knowledge based and Empirical Methods*. Dordrecht: Springer.

Habash, Nizar, and Ryan Roth. 2009. "Catib: The Columbia Arabic treebank." In *Proceedings of the ACL-IJCNLP 2009 conference short papers*, 221-224. https://aclanthology.org/P09-2056.pdf

Habash, Nizar, Owen Rambow, and Ryan Roth. 2009. "MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and

lemmatization." In *Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR)*, vol. 41, Cairo, Egypt.

Habash, Nizar Y. 2010. *Introduction to Arabic Natural Language Processing: Synthesis Lectures on Human Language Technologies* 3 (1): 1–187. Morgan & Claypool Publishers. https://doi.org/10.2200/s00277ed1v01y201008hlt010

Habash, Nizar, Owen Rambow, and Ryan Roth. 2012a. "MADA+TOKAN Manual." Technical Report CCLS-12-01, Columbia University, New York, NY.

Habash, Nizar, Diab, M., and Rambow, O. 2012b. "Conventional orthography for dialectal Arabic." In *Proceedings of the Eighth Language Resources and Evaluation Conference*, 711–718. Istanbul, Turkey: European Language Resources Association. http://www.lrec-conf.org/proceedings/lrec2012/pdf/579_Paper.pdf

Habash, Nizar, Fadhl Eryani, Salam Khalifa, Owen Rambow, Dana Abdulrahim, Alexander Erdmann, Reem Faraj et al. 2018. "Unified guidelines and resources for Arabic dialect orthography." In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. https://aclanthology.org/L18-1574.pdf

Hadni, Meryeme, Said Alaoui Ouatik, Abdelmonaime Lachkar, and Mohammed Meknassi. 2013. "Hybrid part-of-speech tagger for non-vocalized Arabic text." *International Journal on Natural Language Computing (IJNLC)* 2(6): 1-15. https://doi.org/10.5121/IJNLC.2013.2601

Hamdi, Ahmed, Alexis Nasr, Nizar Habash, and Núria Gala. 2015. "POS-tagging of Tunisian dialect using standard Arabic resources and tools." In *Workshop on Arabic Natural Language Processing*, 59-68. https://hal.archives-ouvertes.fr/hal-01464860/

Harris, Zellig. 1962. *String analysis of language structure*. The Hague: Mouton and Co.

Hasan, Fahim Muhammad. 2006. "Comparison of different POS tagging techniques for some South Asian languages." Bachelor Thesis (published online), Dhaka: BRAC University. http://dspace.bracu.ac.bd/bitstream/handle/10361/83/Comparison%20of%20diffferent%20pos%20tagging%20techniques.pdf?sequence=1

Ibn Jinni, A. 1983. *al-Khasaais*.[in Arabic], 3rd edition. Vol. 1. Beirut: Daar al-Kutub.

Jarrar, Mustafa, Nizar Habash, Faeq Alrimawi, Diyam Akra, and Nasser Zalmout. 2017. "Curras: an annotated corpus for the Palestinian Arabic dialect." *Language Resources and Evaluation*, 51(3): 745-775. https://doi.org/10.1007/s10579-016-9370-7

Jurafsky, Daniel; and James H. Martin. 2000. Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition. New Jersey: Prentice Hall, chapter8: 285-286.

Jurafsky, Danial, and Martin, H. James. 2020. *Speech and language processing*: *an introduction to natural language processing, computational linguistics, and speech recognition.* United States: Pearson**.** *https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf*

Karlsson, Fred, Atro Voutilainen, Juha Heikkilae, and Arto Anttila, (eds). 1994. *Constraint Grammar: a Language-Independent System for Parsing Un-restricted Text*. Berlin: Mouton de Gruyter.

Karlsson, Fred, Atro Voutilainen, Juha Heikkilae, and Arto Anttila, (eds). 2011. *Constraint Grammar: a language-independent system for parsing unrestricted text*. Vol. 4. Walter de Gruyter.

Khalifa, Salam, Nizar Habash, Fadhl Eryani, Ossama Obeid, Dana Abdulrahim, and Meera Al Kaabi. 2018. "A morphologically annotated corpus of Emirati Arabic." In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018).* Miyazaki, Japan. https://aclanthology.org/L18-1607.pdf

Khoja, Shereen, Roger Garside, and Gerry Knowles. 2001. "An Arabic tagset for the morphosyntactic tagging of Arabic." Lancaster: Lancaster University. https://eprints.lancs.ac.uk/id/eprint/11985

Khoja, Shereen. 2001. "APT: Arabic part-of-speech tagger." In *Proceedings of the Student Workshop at the Second Meeting of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Pittsburgh. 20-25. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.92.4038&rep=rep1&type=pdf

Khoja, Shereen. 2003. "APT: An automatic Arabic part-of-speech tagger." PhD Thesis (published online), Lancaster University. https://eprints.lancs.ac.uk/id/eprint/12350/

Klein, Sheldon, and Robert F. Simmons. 1963. "A computational approach to grammatical coding of English words." *Journal of the ACM (JACM),* Vol.10 (3): 334–347. DOI: https://doi.org/10.1145/321172.321180

Köprü, Selçuk. 2011. "An efficient part-of-speech tagger for arabic." In: Gelbukh A.F. (eds) *Computational Linguistics and Intelligent Text Processing. CICLing 2011. Lecture Notes in Computer Science,* vol. 6608. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-19400-9_16

Kučera, Henry and Francis, W Nelson. 1967. *Computational analysis of present-day American English*. Providence: Brown University Press.

Kumawat, Deepika, and Vinesh Jain. 2015. "POS tagging approaches: a comparison." *International Journal of Computer Applications*. Vol.118 (6)" 32-38. https://research.ijcaonline.org/volume118/number6/pxc3903148.pdf

Kupiec, Julian. 1992. "Robust part-of-speech tagging using a hidden Markov model." *Computer Speech & Language.* Vol. 6(3): 225-242. https://doi.org/10.1016/0885-2308(92)90019-Z

Lafferty, John, Andrew McCallum, and Fernando CN Pereira. 2001. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." In: *Proceedings of the Eighteenth International Conference on Machine Learning*. San Francisco, CA, USA. Morgan Kaufmann Publishers Inc, 282–289. https://repository.upenn.edu/cis_papers/159/

Maamouri, M., Bies, A., Krouna, S., Gaddeche, F., Bouziri, B. (2009). "Arabic Tree Banking Morphological Analysis and POS Annotation," Ver. 3.8, Univ. Pennsylvania: Linguistic Data Consortium.

Maamouri, Mohamed, and Ann Bies. 2004. "Developing an Arabic treebank: Methods, guidelines, procedures, and tools." In *Proceedings of the Workshop on Computational Approaches to Arabic Script-based languages*, 2-9. https://aclanthology.org/W04-1602.pdf

Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. "The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus." In *NEMLAR Conference on Arabic Language Resources and Tools*, 102-109, Cairo, Egypt. https://www.marefa.org/images/e/e8/The_penn_arabic_treebank_Building_a_large-scale_an_%281%29.pdf

Marcus, Mitchell, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. "Building a large annotated corpus of English: The Penn Treebank." https://repository.upenn.edu/cgi/viewcontent.cgi?article=1246&context=cis_reports

Merialdo, Bernard. 1994. "Tagging English text with a probabilistic model." *Computational linguistics*. Vol. *20*(2): 155-171. https://dl.acm.org/doi/abs/10.5555/972525.972526

Mohamed, Emad; and Sandra K¨ubler. 2010. "Arabic part of speech tagging." Proceedings of LREC, Valetta: Malta.

Muaidi, Hasan. 2014. "Levenberg-Marquardt learning neural network for part-of-speech tagging of Arabic sentences." *Wseas Transactions On Computers* 13: 300-09. http://www.wseas.us/journal/pdf/computers/2014/a185705-494.pdf

Neunerdt, Melanie, Bianka Trevisan, Michael Reyer, and Rudolf Mathar. 2013. "Part-Of-Speech Tagging for Social Media Texts." In: Gurevych I., Biemann C., Zesch T. (eds) *Language Processing and Knowledge in the Web*. Lecture Notes in Computer Science, vol. 8105: 139-150. Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-642-40722-2_15

Obeid, Ossama, Salam Khalifa, Nizar Habash, Houda Bouamor, Wajdi Zaghouani, and Kemal Oflazer. 2018. "MADARi: A web interface for joint Arabic morphological annotation and spelling correction." In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018).* Miyazaki, Japan. https://arxiv.org/abs/1808.08392

Othmane, Chiraz Zribi Ben, Fériel Ben Fraj, and Ichraf Limam. 2017. "POS-tagging arabic texts: A novel approach based on ant colony." *Natural Language Engineering* 23(3), 419-439. https://doi.org/10.1017/S1351324915000480

Owens, Jonathan 1988. The Foundations of Grammar: An Introduction to Medieval Arabic Grammatical Theory. Amsterdam, Philadelphia: John Benjamins.

Paroubek, Patrick. 2007. "Evaluating Part-of-Speech Tagging and Parsing Patrick Paroubek." In *Evaluation of Text and Speech Systems*, 99-124. Springer, Dordrecht.

Pasha, Arfath, Mohamed Al-Badrashiny, Mona T. Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. "Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic." *In Proceedings of LREC*, 14, 1094-1101. Reykjavik, Iceland: European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2014/pdf/593_Paper.pdf

Qafisheh, Hamdi. 1990. “The Phonology of Sanʿani Arabic.” Journal of King Saudi University 2 (2): 167–82.

Rabiee, Hajder Shouhani. 2011. Arabic Language Analysis Toolkit, Dissertation. University of Leeds: School of Computing Studies. http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-130767

Ramsay, A., and Y. Sabtan. 2009. “Bootstrapping a lexicon-free tagger for Arabic.” In *Proceedings of the 9th Conference on Language Engineering (ESOLEC’2009)*, 202-215.

Rathod, Shubhangi, and Sharvari Govilkar. 2015. “Survey of various POS tagging techniques for Indian regional languages.” *International Journal of Computer Science and Information Technologies (IJCSIT)*, Vol. 6 (3): 2525-2529. http://www.ijcsit.com/docs/Volume%206/vol6issue03/ijcsit20150603118.pdf

Ryding, Karin C. 2005. A Reference Grammar of Modern Standard Arabic. New York: Cambridge University Press.

Sawalha, Majdi, and Eric Atwell. 2013. “A standard tag set expounding traditional morphological features for Arabic language part-of-speech tagging.” *Word Structure* 6(1): 43-99. https://www.euppublishing.com/doi/full/10.3366/word.2013.0035

Schiller, A., Teufel, S., &Thielen, C. (1995). Guidelines fur das Tagging deutscherTextcorporamit STTS. *Universität Stuttgart, UniversitätTübingen, Germany*.

Schmid, Helmut. 1994. “Part-of-speech tagging with neural networks.” In *Proceedings of the 15th conference on Computational linguistics*. Vol. 1: 172-176. https://arxiv.org/abs/cmp-lg/9410018

Schmid, Helmut. 1999. “Improvements in part-of-speech tagging with an application to German.” In: Armstrong S., Church K., Isabelle P., Manzi S., Tzoukermann E., Yarowsky D. (eds) *Natural Language Processing Using Very Large Corpora. Text, Speech and Language Technology*, vol. 11. Dordrecht: Springer. https://doi.org/10.1007/978-94-017-2390-9_2

Sharaf Addin Mohammed, Sabah Al-Shehabi. 2020. “Developing Social-Media Based Text Corpus for Sanʿani Dialect (SMTCSD).” In: *Satapathy S.C., Raju K.S., Shyamala K., Krishna D.R., Favorskaya M.N. (eds) Advances in Decision Sciences, Image Processing, Security and Computer Vision. Learning and Analytics in Intelligent Systems, vol 3*. Springer, Cham. https://doi.org/10.1007/978-3-030-24322-7_60

Sproat, Richard. 2007. Preface. *Arabic computational morphology: knowledge-based and empirical methods*, by Soudi, Abdelhadi, Günter Neumann, and Antal Van den Bosch. In Arabic computational morphology, vii-viii. Dordrecht, The Netherlands: Springer.

Stolz, S. Walter S., Percy H. Tannenbaum, and Frederick V. Carstensen. 1965. “Stochastic approach to the grammatical coding of English.” *Communications of the ACM*, Vol. 8 (6): 399–405. DOI: https://doi.org/10.1145/364955.364991

Tapanainen, Pasi., and Voutilainen, Atro. 1994. “Tagging accurately--Don't guess if you know.” https://arxiv.org/abs/cmp-lg/9408009

Tlili-Guiassa, Y. 2006. "Hybrid method for tagging Arabic text." *Journal of Computer science*, *2*(3), 245-248. https://pdfs.semanticscholar.org/4717/22c88764693a164562780556a595796c5fe0.pdf

Toutanova, Kristina, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. "Feature-rich part-of-speech tagging with a cyclic dependency network." In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 252-259. https://aclanthology.org/N03-1033.pdf

Van Guilder, Linda. "Automated part of speech tagging: a brief overview." 1995. (A handout for LING361, Georgetown University. Retrieved from http://ccl.pku.edu.cn/doubtfire/NLP/Lexical_Analysis/Word_Segmentation_Tagging/POS_Tagging_Overview/POS%20Tagging%20Overview.htm

Watson, Janet. 1993. *A Syntax of Saṇʿānī Arabic*. Wiesbaden: O. Harrassowitz.

Watson, Janet CE. 2002. *The phonology and morphology of Arabic*. Oxford: Oxford University Press Inc.

Watson, Janet. 2008. "Sanani Arabic." In: Encyclopedia of Arabic Language and Linguistics 4: 106-115. Publisher: Brill.

Zaghouani, Wajdi. 2017. "Critical survey of the freely available Arabic corpora." In *Proceedings of the International Conference on Language Resources and Evaluation (LREC'2014)*, OSACT Workshop. Reykjavik, Iceland, 26-31. https://arxiv.org/abs/1702.07835

Zaghouani, Wajdi, and Anis Charfi. 2018. "Arap-tweet: A large multi-dialect twitter corpus for gender, age and language variety identification." In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. https://arxiv.org/abs/1808.07674

Zribi, Chiraz Ben Othmane, Aroua Torjmen, and Mohamed Ben Ahmed. 2007. "A Multi-Agent System for POS-Tagging Vocalized Arabic Texts." *The International Arab Journal of Information Technology*. Vol. 4(4): 322-329. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.451.8653&rep=rep1&type=pdf

Appendix A: San'ani Arabic Parts-of-Speech Tagger Code

```
!pip install terminaltables

!pip install seqeval

!pip install git+https://www.github.com/keras-team/keras-contrib.git

!pip install keras

!pip install tensorflow


import tensorflow

import keras

from keras.layers import Dense, Input, GRU, Embedding, Dropout,
Activation, Masking

from keras.layers import Bidirectional, GlobalMaxPool1D,
TimeDistributed

from keras.models import Model, Sequential

from keras_contrib.layers import CRF

import numpy as np

import tensorflow as tf

import re

import gensim

import gensim.models.keyedvectors as word2vec

from keras.utils import to_categorical

import pandas as pd

import matplotlib.pyplot as plt

from seqeval.metrics import accuracy_score

from seqeval.metrics import classification_report

from seqeval.metrics import f1_score


embedding =
word2vec.KeyedVectors.load_word2vec_format('/home/mohammed/POS
Training/wiki.ar.vec', binary=False)
```

```
#specialchar = '!"#$%&\'()*+,-./:;<=>?@[\]^_`{|}~’:؟_،«» '
specialchar = '#$%&\+-/<=>@[]^_`{|}~:_«» ' #so set specialchar
variable to this
def clean_word(word):
    word = word.translate(str.maketrans({key: None for key in
specialchar}))

    #remove tashkeel
    p_tashkeel = re.compile(r'[\u0617-\u061A\u064B-\u0652]')
    word = re.sub(p_tashkeel,"", word)

    return word



def load():
    data = pd.read_excel('/home/mohammed/POS
Training/training_data1st.xlsx')
    s=''
    l=[]
    for i in range(len(data)):
        is_NaN = data.iloc[i].isnull()
        if is_NaN.any() == False:
            s+=str(str(data.iloc[i][0])+str(" ") +
str(data.iloc[i][1])+str('\n'))
        if is_NaN.any() == True:
            l.append(s[:-1])
            s=''
    m=[]
    for i in l:
        if i != '':
            m.append(i)
    sents=m
    # tokenize words
    words = [None]*len(sents)
    tokens = [None]*len(sents)
    for i, sent in enumerate(sents):
        sent = sent.split('\n')
        words[i] = []
        tokens[i] = []
        for word in sent:
            line = word.rsplit(' ', 1)
            line[0] = clean_word(line[0])
            if len(line[0]) > 0:
                words[i].append(line[0])
```

```
                tokens[i].append(line[1])


    return [d for d in words if len(d) > 0], [d for d in tokens if
len(d) > 0]

# load data
sents, labels = load()

data = pd.read_excel('/home/mohammed/POS
Training/training_data1st.xlsx')
data.head()
Values = data["Unnamed: 1"].values.ravel()
tags = [x for x in list(pd.unique(Values)) if str(x) != 'nan' ]

# Replace tag_classes with tags
tag_classes =
['NN','NNP','PRP','WP','D_PRP','JJ','RB','WRB','CD','FCD','FW','CC','S
C','DT','RP','INTG_RP','IN','AUX_VB','VB','UH','PUNC','SYM','DD']

# embed words
for i, sent in enumerate(sents):
    for j, word in enumerate(sent):
        try:
            sents[i][j] = embedding[word]
        except KeyError:
            sents[i][j] = embedding['unk']

# embed labels
for i, tokens in enumerate(labels):
    labels[i] = [to_categorical(tag_classes.index(tag),
num_classes=len(tag_classes)) for tag in tokens]

################################
# No. sentences: 4898
# No. all words: 135717
# No. 3/4 all words: 101787
# Index of 3/4 sentences: 3569
################################

# pad sequences
max_sent_length = 140
sents_lengths = []
for i, sent in enumerate(sents):
    sents_lengths.append(len(sent))
    l = max_sent_length - len(sent)
    sents[i] += [[0]*300]*l
```

```
for i, label in enumerate(labels):
    l = max_sent_length - len(label)
    labels[i] += [[0]*22+[0]]*l


# split data it depends on your data 0.7*N train / 0.3*N test for
example; with N the number of observation of your data
train_x, train_y = sents[:7261], labels[:7261]
test_x, test_y = sents[7262:], labels[7262:]


def build_model():
    crf_layer = CRF(23)
    input_layer = Input(shape=(None, 300,))    #embedding =
Embedding(212, 20, input_length=None, mask_zero=False)(input_layer)
    mask_layer = Masking(mask_value=0., input_shape=(140,
300))(input_layer)
    bi_gru = Bidirectional(GRU(32, return_sequences=True))(mask_layer)
    bi_gru = TimeDistributed(Dense(32, activation="relu"))(bi_gru)
    output_layer = crf_layer(bi_gru)
    return Model(input_layer, output_layer), crf_layer


# build model
train_model, crf_layer = build_model()
train_model.compile(optimizer="rmsprop", loss=crf_layer.loss_function,
metrics=[crf_layer.accuracy])
train_model.summary()

# train model
history = train_model.fit(np.array(train_x), np.array(train_y),
epochs=10, verbose=1, validation_data=(np.array(test_x),
np.array(test_y)))

train_model.save_weights('weights.hd5f')


# plot accuracy
hist = pd.DataFrame(history.history)
plt.style.use("ggplot")
plt.figure(figsize=(6,6))
plt.plot(hist["val_crf_viterbi_accuracy"], color='red')
plt.plot(hist["crf_viterbi_accuracy"],color='blue')
plt.show()

# plot accuracy
hist = pd.DataFrame(history.history)
plt.style.use("ggplot")
```

```
plt.figure(figsize=(6,6))
plt.plot(hist["val_loss"], color="red")
plt.plot(hist["loss"], color='blue')
plt.show()


# testing
pred = train_model.predict(np.array(test_x, dtype='float64'))
pred_x = []
pred_y = []
for i, sent in enumerate(pred):
    pred_x.append([tag_classes[np.argmax(w)] for w in
pred[i][:sents_lengths[i]]])
    pred_y.append([tag_classes[np.argmax(w)] for w in
test_y[i][:sents_lengths[i]]])
#print(classification_report(pred_y, pred_x, target_names =
tag_classes))
print(classification_report(pred_y, pred_x))
print('f1_score: ')
print(f1_score(pred_y, pred_x))


print(accuracy_score(pred_y, pred_x))
```

Appendix B: Examples of Parts-of-Speech tagger Output

In this Appendix, we show the output of the Parts-of-Speech tagger the input is extracted from our San'ani Arabic social media corpus.

Word	Prediction
في	IN
بيت	NN
الحج	NN
حميد	NNP
..	PUNC
أماني	NNP
:	PUNC
بتهدي	VB
سميره	NNP
...	PUNC
يا	RP
اختي	NN
خلاص	UH
لاعاد	AUX_VB
تكبريهاش	VB
وتسيري	VB
بيتكم	NN
ابي	NN

طيبين	JJ
سيري	VB
اعتذري	VB
منهم	IN
وبس	RB
...	PUNC
سميره	NNP
:	PUNC
بتدخل	VB
لداتها	NN
في	IN
الشنطه	NN
..	PUNC
لا	RP
شاسير	VB
اعتذر	VB
ولا	RP
أنا	PRP
غلطانه	JJ
أنا	PRP
ما	RP
ضبح	VB
عمي	NN
يمكني	VB

اخرجي	VB
من	IN
بيتي	NN
تمام	UH
قد	RP
وذا	D_PRP
شخرج	VB
ويفتهن	VB
سوى	RB
هو	PRP
ورزق	NNP
..	PUNC

Word	Prediction
أحلام	NNP
:	PUNC
لحقته	VB
..	PUNC
ليش	INTG_RP
مايشتوش	VB
أبوك	NN
يدخل	VB
..	PUNC
أحمد	NNP

:	PUNC
شرجع	VB
اجابركم	VB
بعدا	CC
المهم	JJ
قوى	UH
لو	SC
جو	VB
لا	RP
تخلوهمش	VB
يدخلوا	VB
لايتعبوا	VB
سيدي	NN
..	JJ
.....	PUNC
.....	PUNC
.....	PUNC
في	IN
بيت	NN
عبدالكريم	NNP
...	PUNC
سميره	NNP
:	PUNC
ذلحين	RB

ماعتفعل	VB
عتسير	VB
المستشفى	NN
ولا	RP
مع	RB
..	PUNC
عبدالكريم	NNP
:	NNP
ايووه	UH
قدهو	NN
مايسبرش	VB
افلت	VB
..	PUNC
سميره	NNP
:	UH
ما	INTG_RP
رأئيك	NN
لو	SC
تسجل	VB
المحل	NN
بإسمي	NN
لو	SC
افتهن	VB
عمي	NN
وقال	VB

يشتي	VB
حقه	NN
شاقل	VB

هو	PRP
حقي	NN
اني	RP
بعت	VB
الذهب	NN
وخليتك	VB
تشغله	VB
لي	IN
..	PUNC
عبدالكريم	NNP
:	PUNC
بدون	NN
ما	RP
اسجله	VB
عنقل	VB
انتي	PRP
بعتي	VB
ذهبش	NN
اهم	JJ
شي	NN
لاتلبسيش	VB

هذه	D_PRP
الفتره	NN
..	PUNC

Development of San’ani Arabic Parts-Of-Speech Tagger: a BI-GRUs-CRF Model

by Sabah Al-shehabi

Submission date: 27-Dec-2021 09:27PM (UTC+0530)
Submission ID: 1735931756
File name: Sabah_AL-Shehabi.pdf (3.18M)
Word count: 40385
Character count: 210277

Development of San'ani Arabic Parts-Of-Speech Tagger: a BI-GRUs-CRF Model

ORIGINALITY REPORT

6%	4%	4%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	etheses.whiterose.ac.uk Internet Source	1%
2	"Advances in Decision Sciences, Image Processing, Security and Computer Vision", Springer Science and Business Media LLC, 2020 Publication	1%
3	"Intelligent Computing in Engineering", Springer Science and Business Media LLC, 2020 Publication	1%
4	eprints.whiterose.ac.uk Internet Source	<1%
5	feng.stafpu.bu.edu.eg Internet Source	<1%
6	qisar.fssr.uns.ac.id Internet Source	<1%
7	courses.washington.edu Internet Source	<1%

8	Submitted to Robert Morris University Student Paper	<1 %
9	www.aclweb.org Internet Source	<1 %
10	Diganta Baishya, Rupam Baruah. "Highly Efficient Parts of Speech Tagging in Low Resource Languages with Improved Hidden Markov Model and Deep Learning", International Journal of Advanced Computer Science and Applications, 2021 Publication	<1 %
11	jscdss.com Internet Source	<1 %
12	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2018 Publication	<1 %
13	snl.no Internet Source	<1 %
14	K. K. Akhil, R. Rajimol, V. S. Anoop. "Parts-of-Speech tagging for Malayalam using deep learning techniques", International Journal of Information Technology, 2020 Publication	<1 %
15	core.ac.uk Internet Source	<1 %

16	etheses.bham.ac.uk Internet Source	<1 %
17	www.cambridge.org Internet Source	<1 %
18	Wasan AlKhwiter, Nora Al-Twairesh. "Part-of-speech Tagging for Arabic Tweets using CRF and BiLSTM", Computer Speech & Language, 2020 Publication	<1 %
19	powcoder.com Internet Source	<1 %
20	Ndl.ethernet.edu.et Internet Source	<1 %
21	acl.ldc.upenn.edu Internet Source	<1 %
22	Jihene Younes, Emna Souissi, Hadhemi Achour, Ahmed Ferchichi. "Language resources for Maghrebi Arabic dialects' NLP: a survey", Language Resources and Evaluation, 2020 Publication	<1 %
23	Suvarna G Kanakaraddi, Suvarna S Nandyal. "Survey on Parts of Speech Tagger Techniques", 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), 2018	<1 %

Publication

24	www.garph.org Internet Source	<1 %
25	Submitted to Victoria University Student Paper	<1 %
26	www.nowpublishers.com Internet Source	<1 %
27	J. B. Moore. "Loop recovery via H∞ / H2 sensitivity recovery", International Journal of Control, 4/1/1989 Publication	<1 %
28	www.airccse.org Internet Source	<1 %
29	Zhege Liu, Junxing Cao, Jiachun You, Shuna Chen, Yujia Lu, Peng Zhou. "A lithological sequence classification method with well log via SVM-assisted bi-directional GRU-CRF neural network", Journal of Petroleum Science and Engineering, 2021 Publication	<1 %
30	Getachew Mamo, Million Meshesha. "Parts of Speech Tagging for Afaan Oromo", International Journal of Advanced Computer Science and Applications, 2011 Publication	<1 %
31	www.lrec-conf.org Internet Source	

		<1 %
32	Anna Feldman, Jirka Hana. "A resource-light approach to morpho-syntactic tagging", Brill, 2010 Publication	<1 %
33	Ruo-Hong Huan, Jia Shu, Sheng-Lin Bao, Rong-Hua Liang, Peng Chen, Kai-Kai Chi. "Video multimodal emotion recognition based on Bi-GRU and attention fusion", Multimedia Tools and Applications, 2020 Publication	<1 %
34	Submitted to Stockhom University & The Royal Institute of Technology Student Paper	<1 %
35	Submitted to University of Leeds Student Paper	<1 %
36	mafiadoc.com Internet Source	<1 %
37	www.coursehero.com Internet Source	<1 %
38	Ghassan Kanaan ., Riyad al-Shalabi ., Majdi Sawalha .. "Improving Arabic Information Retrieval Systems Using Part of Speech Tagging", Information Technology Journal, 2005 Publication	<1 %

39	Submitted to University of California, Los Angeles Student Paper	<1%
40	eprints.soas.ac.uk Internet Source	<1%
41	"Cross-Linguistic Aspects of Processability Theory", John Benjamins Publishing Company, 2005 Publication	<1%
42	Simran Kaur Jolly, Rashmi Agrawal. "Chapter 14 Parts of Speech Tagging for Punjabi Language Using Supervised Approaches", Springer Science and Business Media LLC, 2020 Publication	<1%
43	Submitted to University of Birmingham Student Paper	<1%
44	acl-bg.org Internet Source	<1%
45	"Arabic Language Processing: From Theory to Practice", Springer Science and Business Media LLC, 2018 Publication	<1%
46	Hebah ElGibreen, Mohammed Faisal, Mansour Al Sulaiman, Sherif Abdou et al. "An Incremental Approach to Corpus Design and	<1%

	Construction: Application to a Large Contemporary Saudi Corpus", IEEE Access, 2021 Publication	
47	Submitted to University of Malaya Student Paper	<1 %
48	www.mdpi.com Internet Source	<1 %
49	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
50	Submitted to Nanyang Technological University Student Paper	<1 %
51	Submitted to University of Northumbria at Newcastle Student Paper	<1 %
52	era.ed.ac.uk Internet Source	<1 %
53	"Advances in Asian Mechanism and Machine Science", Springer Science and Business Media LLC, 2022 Publication	<1 %
54	"Handbook of Natural Language Processing and Machine Translation", Springer Science and Business Media LLC, 2011 Publication	<1 %

55	Daniel Currie Hall. "Phonological contrast and its phonetic enhancement: dispersedness without dispersion", Phonology, 2011 Publication	<1 %
56	Nicola Lampitelli. "The Romance plural isogloss and linguistic change: A comparative study of Romance nouns", Lingua, 2014 Publication	<1 %
57	Www.merriam-webster.com Internet Source	<1 %
58	"Multimodal Analyses enabling Artificial Agents in Human-Machine Interaction", Springer Science and Business Media LLC, 2015 Publication	<1 %
59	clok.uclan.ac.uk Internet Source	<1 %
60	ivythesis.typepad.com Internet Source	<1 %

Exclude quotes	On	Exclude matches	< 14 words
Exclude bibliography	On		

INTERNATIONAL CONFERENCE ON EMERGING TRENDS IN ENGINEERING (ICETE)

UNIVERSITY COLLEGE OF ENGINEERING
OSMANIA UNIVERSITY
HYDERABAD, INDIA

Certificate of Participation

This is to certify that Sabah Mohammed M. Alshehabi of University of Hyderabad has Presented a paper titled Developing Social-Media Based Text Corpus for San'ani Dialect (SMTCSD) at International Conference on Emerging Trends in Engineering (ICETE) held during 22 - 23 March 2019.

Dr. D. Vijay Kumar
Convener

Prof. Kumar Molugaram
Chief Patron, Alumni Association Univ. College of Engg. O.U.
(Principal, Univ. College of Engineering (A), O.U.)

Er. P. Ram Reddy
Chairperson

Dr. D. Rama Krishna
Convener

Prof. P. V. N. Prasad
Chairperson

Springer

Organised by

Alumni Association, UCE, OU

IFERP®
connecting engineers... developing research

CERTIFICATE

OF PRESENTATION

chalapathi
Institute of Technology
Affiliated to JNTUK | MOTHADAKA

INTERNATIONAL CONFERENCE ON
RECENT CHALLENGES IN SCIENCE, ENGINEERING AND TECHNOLOGY(ICRCSET-2020)

28th & 29th February 2020

This is to certify that **Sabah Al-Shehabi** of University of Hyderabad, Hyderabad, India & Sana'a University, Yemen presented his/her research paper titled *A Grammatically Annotated Corpus for Sana'ani Arabic Dialect* in the "International Conference on Recent Challenges in Science, Engineering and Technology (ICRCSET-2020)" Organized by Chalapathi Institute of Technology (CIT), Guntur, Andhra Pradesh on 28th - 29th February 2020 at Guntur, Andhra Pradesh.

Prof. K. Naga Srinivasa Rao
Principal
Chalapathi Institute of Technology
Guntur.

Dr. P. Balamuralikrishna
Convener | Professor & Dean of R & D
Chalapathi Institute of Technology
Guntur

Mr. Rudra Bhanu Satpathy
Chief Executive Officer
IFERP

IFERP®
connecting engineers... developing research

CERTIFICATE

OF PRESENTATION

chalapathi
Institute of Technology
Affiliated to JNTUK | MOTHADAKA

INTERNATIONAL CONFERENCE ON
RECENT CHALLENGES IN SCIENCE, ENGINEERING AND TECHNOLOGY(ICRCSET-2020)

28th & 29th February 2020

This is to certify that

Sabah AL-Shehagi & Mohammed Sharaf-Addin

has presented his/her research paper titled

A Grammatically Annotated Corpus for Sana'ani Arabic Dialect

which has been awarded as BEST RESEARCH PAPER in ICRCSET-2020

held on 28th - 29th February 2020 at Chalapathi Institute of Technology (CIT), Guntur, Andhra Pradesh.

Prof. K. Naga Srinivasa Rao
Principal
Chalapathi Institute of Technology
Guntur.

Dr. P. Balamuralikrishna
Convener | Professor & Dean of R & D
Chalapathi Institute of Technology
Guntur

IFERP

Mr. Rudra Bhanu Satpathy
Chief Executive Officer
IFERP

March - April 2020
ISSN: 0193-4120 Page No. 4953 - 4961

A Grammatically Annotated Corpus for Sana'ani Arabic Dialect

[1][2]Sabah Al-Shehabi, [3][4]Mohammed Sharaf-Addin
[1]CALTS, University of Hyderabad, Hyderabad, India
[2]Department of English, Faculty of Education, Mahweet, Sana'a University, Sana'a, Yemen.
[3]CAS in Linguistics, Osmania University, Hyderabad, India
[4] Department of English, Faculty of Arts, Thamar University, Thamar, Yemen
[1][2] sabahmohammed986@gmail.com, [3][4]ma.alshami22@gmail.com

Article Info
Volume 83
Page Number: 4953 - 4961
Publication Issue:
March - April 2020

Article History
***Article Received**: 24 July 2019*
***Revised**: 12 September 2019*
***Accepted**: 15 February 2020*
***Publication**: 27 March 2020*

Abstract
In this paper, we introduce a new resource for Sana'ani Arabic dialect. This grammatically tagged corpus is basically a collection of social media texts that is primarily developed as a training data for developing Sana'ani Arabic Part Of Speech (POS) tagger. The corpus consists of 7,295 tokenized sentences with an average of 15 tokens in each sentence and with a total number of 112,517 tokens and 15,940 types. The corpus is manually annotated using a modified tagset from The Biestagset which covers 24 tags. The manual annotation performed is rather a grammatical annotation ignoring morphological inflections and concentrating on the syntactic features using the context to identify the part of speech of each token.

Index Terms; *Corpus Annotation, Dialectal Arabic,Parts of Speech, Sana'ani Arabic, Tagset*

I. INTRODUCTION

Arabic language is one of the most spoken languages of the world. One of the markers of Arabic language is the diglossic nature of the language [1] where two varieties (Modern Standard Arabic (MSA) and Dialectal Arabic (DA) exists side-by-side and are closely related. MSA is a predominant variety over dialectal Arabic in formal settings which restrict almost all the written content to the standard variety. However, recently and with the advent of technology and the vast spread of social media networking sites, a strong presence of DA is noticed and more individual-driven data becomes accessible and available as users of these sites feel free and encouraged to jot down their thoughts, interact or comment about their daily social life in their own dialects. The challenge, however, remains in obtaining such dialectal datasets which can be viable, and usable by machines. This challenge is tested when it comes to building Natural language Processing (NLP) tools and applications. Therefore, obtaining a clean, preprocessed, valid and machine readable text is a crucial necessity for developing any NLP applications. Online data can be collected from the networking sites either manually or automatically using tools for crawling and compiling. This collection of texts, after being cleaned and preprocessed, which is now called a raw corpus can be considered a standard reference for the language variety which it is supposed to represent. This type of corpus can be used for developing many NLP tools and applications. However, machines are still not smart enough to disambiguate similar contents unless being provided with some added values to the texts. This process is called corpus annotation which [2] defines as the process of 'adding such interpretative, linguistic information to an electronic corpus of spoken and/or written language data'. The advantages of such annotated corpus is suggested by

Published by: The Mattingley Publishing Co., Inc.

Learning and Analytics in Intelligent Systems 3

Suresh Chandra Satapathy · K. Srujan Raju · K. Shyamala · D. Rama Krishna · Margarita N. Favorskaya *Editors*

Advances in Decision Sciences, Image Processing, Security and Computer Vision

International Conference on Emerging Trends in Engineering (ICETE), Vol. 1

Springer

Developing Social-Media Based Text Corpus for San'ani Dialect (SMTCSD)

Mohammed Sharaf Addin[1,2(✉)] and Sabah Al-Shehabi[3,4]

[1] CAS in Linguistics, Osmania University, Hyderabad, India
ma.alshami22@gmail.com
[2] Department of English, Faculty of Arts, Thamar University, Dhamar, Yemen
[3] CALTS, University of Hyderabad, Hyderabad, India
sabahmohammed986@gmail.com
[4] Department of English, Faculty of Education, Mahweet, Sana'a University, Sana'a, Yemen

Abstract. This paper aims at developing and designing a social media based text corpus of San'ani Dialect (SMTCSD). The corpus is considered the first in the research area that codifies one of the most popular and spoken dialects in Yemen representing nearly 30% of Yemeni speakers. Our primary objective is a compilation of authentic and unmodified texts gathered from different open-source social media platforms mainly Facebook and Telegram Apps. As a result, we obtained a corpus of 447,401 tokens and 51,073 types with an 11.42% Token:Type Ratio (TTR) that is composed in entirely manual and non-experimental conditions. The corpus represents daily natural conversations which are found in the form of fictional dialogues, representing different situations and topics during the years 2017 and 2018. The data is preprocessed and normalized which then is classified into ten different categories. The analysis of the corpus is made using LancsBox, and different statistical analyses are performed.

Keywords: Corpus design · San'ani dialect · Social media · Token · Type · Category · LancsBox · Statistical analysis

1 Introduction

Arabic Language is one of the six main languages of the world with approximately thirty dialects. It has three major varieties. The first form is classical Arabic which is the form of the Holy Quran and historical literature. The second form is Modern Standard Arabic (henceforth MSA) which covers the written form mostly and rarely formal speech that is used in media, academics, and news. The third form is Colloquial Arabic or Dialectal Arabic (DA) that presents the regional dialects used as informal speech. So Arabic Language is a good example of diglossia where two varieties of the same language are used by the speakers for formal and informal interaction. MSA is the high variety that represents the official language in all the Arab countries while Colloquial Arabic or DA is the low variety that is used for informal speech.

The original version of this chapter was revised: The affiliations of author group have been updated. The correction to this chapter is available at https://doi.org/10.1007/978-3-030-24322-7_93

S. C. Satapathy et al. (Eds.): ICETE 2019, LAIS 3, pp. 491–500, 2020.
https://doi.org/10.1007/978-3-030-24322-7_60

> University of Hyderabad,
> Hyderabad, India

Nazrin Laskar <nazrinlaskar@gmail.com> Mon, Dec 20, 2021 at 9:27 PM
To: Sabah Mohammed <sabahmohammed986@gmail.com>

Dear Mr Sabah Mohammed
We are very pleased to inform you that your co-authored paper titled **"PRE-PROCESSING AND ANNOTATION OF SOCIAL MEDIA TEXT FOR SAN'ANI ARABIC POS TAGGING SYSTEM"** has been accepted for publication in ***AJL 11.***
Best wishes
Dr Nazrin
Editorial Team ***AJL***
[Quoted text hidden]

Sabah Mohammed <sabahmohammed986@gmail.com> Mon, Dec 20, 2021 at 9:34 PM
To: Mohammed Alshami <ma.alshami22@gmail.com>

[Quoted text hidden]