Lexical and Language Modeling of Diacritics and Morphemes in Arabic Automatic Speech Recognition

by

Tuka Al Hanai

B.S., The Petroleum Institute, Abu Dhabi, UAE (2011)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2014

Author .....
Department of Electrical Engineering and Computer Science
January 31, 2014

Certified by .....
James R. Glass
Senior Research Scientist
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Theses

Lexical and Language Modeling of Diacritics and Morphemes in Arabic Automatic Speech Recognition

by

Tuka Al Hanai

Submitted to the Department of Electrical Engineering and Computer Science
on January 31, 2014, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Arabic is a morphologically rich language which rarely displays diacritics. These two features of the language pose challenges when building Automatic Speech Recognition (ASR) systems. Morphological complexity leads to many possible combinations of stems and affixes to form words, and produces texts with high Out Of Vocabulary (OOV) rates. In addition, texts rarely display diacritics which informs the reader about short vowels, geminates, and nunnations (word ending /n/). A lack of diacritics means that 30% of textual information is missing, causing ambiguities in lexical and language modeling when attempting to model pronunciations, and the context of a particular pronunciation. Intuitively, from an English centric view, the phrase *thwrtr wrt n thwrt* with 'morphological decomposition' is realized as, *th wrtr wrt n th wrt*. Including 'diacritics' produces, *the writer wrote in the writ*.

Thus our investigations in this thesis are twofold. Firstly, we show the benefits and interactions between modeling all classes of diacritics (short vowels, geminates, nunnations) in the lexicon. On a Modern Standard Arabic (MSA) corpus of broadcast news, this provides a 1.9% absolute improvement in Word Error Rate (WER) ($p < 0.001$). We also extend this graphemic lexicon with pronunciation rules, yielding a significant improvement over a lexicon that does not explicitly model diacritics. This results in a of 2.4% absolute improvement in WER ($p < 0.001$).

Secondly, we show the benefits of language modeling at the morphemic level with diacritics, over the commonly available, word-based, nondiacratized text. This yields an absolute WER improvement of 1.0% ($p < 0.001$).

Thesis Supervisor: James R. Glass
Title: Senior Research Scientist

Acknowledgments

My humble thanks to my advisor Jim Glass for his support, patience, and insight. I'm very grateful.

Thanks to the SLSers for their warmth, and hospitality.

Thanks to the, too many to count, people in my life, past, present, and future.

A heartfelt thanks to the Abu Dhabi Education Council (ADEC) for their tireless efforts, and gracious support.

Of course, special, special, special, and immeasurable thanks to my parents and siblings.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations for Arabic ASR

With over 350 million speakers and as the fifth most spoken language in the world, Arabic has a large user base, and like all languages, there exist many many more people that do not speak it [7]. This presents us with many potential applications in which speakers could benefit from, such as interfacing more naturally and efficiently with their mobile or car navigation devices through voice, as well as eliminating communication barriers between speakers and non-speakers of the language.

1.2 Challenges

Arabic exhibits a number of challenges when conducting research in Automatic Speech Recognition (ASR).

Dialects. Due to the large number of speakers, there are variations in pronunciation, phonetics, prosody, grammar, and vocabulary.

Script. The script generally does not contain diacritics which serve to inform the reader of the vowels associated with each written consonant. This leads to several potential pronunciations for a given word. Speakers are able to discern the underlying word through context and experience.

Morphology. The language also displays rich morphology which leads to a large vocabulary of words produced by the many possible combinations of affixes and stems.

1.3 Objectives

In this thesis we seek to answer the following questions in regards to Arabic ASR performance.

1. Can information contained in diacritics be captured and modeled in the lexicon?
2. What is to be gained from lexicon pronunciation rules?
3. How do diacritics and morphological decomposition impact language modeling?

1.4 Thesis Overview

This thesis is organized as follows.

Chapter 2: Arabic and ASR provides an overview of Arabic and ASR. We describe the orthography, phonetics, and morphology of the language. We also provide an overview of ASR system and the acoustic, lexical, and language modeling used in this thesis.

Chapter 3: Lexical Modeling describes work on the lexicon describing the impact of modeling diacritics and pronunciation rules. We run experiments to study the behavior of modeling different classes of diacritics (short vowels, geminates, nunnations) in a graphemic lexicon. We then add another dimension to the lexical by building it using pronunciation rules given all diacritics. We test three rules available from the literature to model pronunciations with phonemes and phones. Overall, ASR performance improves when modeling diacritics, as well as implementing pronunciation rules.

Chapter 4: Language Modeling presents a number of language modeling techniques using a range of texts formatted with and without diacritics, as well as at the morpheme level. We show how we tackled the issue of text normalization when diacratizing and tokenizing Arabic text, and the gains to be made when modeling with both diacritics and morphemes.

Chapter 5: Conculsion This concludes with the salient points presented in this thesis and closes with ideas for future explorations.

Chapter 2

Arabic and ASR

2.1 Arabic Language

A brief overview of Arabic and how it operates is presented in this section, to provide background on the language, terminologies, and ideas presented in upcoming chapters.

2.1.1 Forms of Arabic

Classed as an Afro-Asiatic language, there are differences in opinion as to what form Arabic exists today, whether it is a diglossia or a triglossia [8, 9]. For the purpose of our research we define Modern Standard Arabic (MSA) as the language spoken in official communications such as newspapers, books, magazine, TV documentaries, and public speeches [7]. Meanwhile, dialectical Arabic is used in regular day-to-day conversations and rarely appears in a written format.

2.1.2 Orthography

The Arabic script is written in a cursive format from right to left. It is composed of 28 characters. Most characters have unique one-to-one mappings to phonemes. Table 2.1 shows these characters and their Romanized version (using Buckwalter Transliteration convention), and Arpabet equivalent representation. Also in the Table are other characters common in Arabic but not considered as an alphabet as it is a

special form of some of the alphabets. Diacritic characters are also shown.

Table 2.1: *Characters of Arabic Script in Arabic, Buckwalter Transliteration, and Arpabet.*

Alphabet								
	د	خ	ح	ج	ث	ت	ب	ا
	d	x	H	j	v	t	b	A
	/d/	/kh/	/h/	/jh/	/th/	/t/	/b/	/ae:/
	ط	ض	ص	ش	س	ز	ر	ذ
	T	D	S	$	s	z	r	*
	/tt/	/dd/	/ss/	/sh/	/s/	/z/	/r/	/dh/
	م	ل	ك	ق	ف	غ	ع	ظ
	m	l	k	q	f	g	E	Z
	/m/	/l/	/k/	/kq/	/f/	/gh/	/ai/	/zh/
			ي	و	ه	ن		
			y	w	h	n		
			/y/	/w/	/hh/	/n/		
Additional Characters								
	ء	آ	أ	ؤ	إ	ىء	ى	‘	ة
	’	|	>	&	<	}	Y	‘	p
	/q/	/ae:/	/q/	/q/	/q/	/q/	/ae:/	/ae:/	/t/
Diacritic Characters								
	بَ	بُ	بِ	بْ	بّ	بًا	بٌ	بٍ
	a	u	i	o	~	F	K	N
	/ae/	/uh/	/ih/	null	/b b/	/ah n/	/uh n/	/ih n/

In the majority of Arabic texts diacritics are rarely displayed and the reader through syntax, context, and experience is able to decipher the pronunciation, grammar, and the true identity of the homographs that exist throughout the language. Word ambiguity is such that every third word in a text can be a homograph [10].

Table 2.2 shows the behaviour of diacritics in the language and script. Every consonant has a diacritic associated with it.

2.1.3 Articulary Phonetics

For each written character in the Arabic alphabet there exists a unique phoneme mapping. Table 2.3 displays these phonemes and their place of articulation [1, 2]. Some English phonemes do not appear in MSA, such as the affricate /ch/, stops

Table 2.2: *Example on Diacritics.*

Undiacratized	كتب الكاتب في الكتاب
Buckwalter	ktb AlkAtb fy AlktAb
Diacratized	كَتَبَ الكَاتِبُ فِي الكِتَابِ
Buckwalter	kataba AlkAtibu fiy AlkitAbi
Arpabet	/k at ae ba ae # ae: l k ae: t ih b uh # f ih y # ae: l k ih t ae: b ih/
Mapped	wrote the-writer in the-book
Translation	The writer wrote in the book.

Arabic is written from right to left. Underlined buckwalter characters correspond to long and short vowels

/g/ and /p/, and fricative /v/. Arabic distinguishes between the fricatives /th/ in *three* and /dh/ in *there*. The Arabic phonemes that do not appear in English are the emphatic version of the stops /t/ and /d/, and fricatives /s/ and /dh/ which are represented as /tt/, /dd/, /ss/, and /zh/. Additionally, the uvulars /kq/, /gh/, and /kh/ don't exist in English, as well as the pharyngeals /ai/ and /h/. The glottal stop /q/ is represented by some extra characters and behaves like other consonants appearing in the beginning (>, <), middle ($'$, >, &, }), and end ($'$, >, &, }) of words. The speaker makes a concerted effort to articulate the glottal stop. This is different from English where it appears as a byproduct of coarticulation, like the *t* in the word *button*.

Arabic has six vowels, three short (/ae/, /uh/, /ih/) and three long (/ae:/, /uh:/, /ih:/) as well as two diphthongs (/aw/, /ay/) all listed in Table 2.4 [1, 2, 3].

2.1.4 Morphology

When studying the structure of words it is important to consider two main components, derivational morphology (how words are formed), and inflectional morphology (how words interact with syntax). Derivational morphology is exhibited in the English word *true* when in the form of *truthfulness* and *untruthful*. Inflectional morphology can be observed in the English root word *s-ng* in certain contexts as *sing*, *song*, *sang*,

Table 2.3: *Phonemes in Arabic [1, 2].*

	Bilabial	Labio-dental	Inter-dental	Alveolar	Palatal	Velar	Uvular	Pha-ryngeal	Glottal
Stops	/b/			/d/ /dd/ /t/ /tt/		/k/	/kq/		/q/
Fricatives		/f/	/th/ /dh/ /zh/	/z/ /s/ /ss/	/sh/		/gh/ /kh/	/ai/ /h/	/hh/
Affricates					/j/				
Nasals	/m/			/n/					
Approximants	/w/			/l/ /r/	/y/				

Phonemes /tt/, /dd/, /ss/, and /zh/ are the emphatic version of /t/, /d/, /s/, and /dh/.

Table 2.4: *Vowels in Arabic [1, 2, 3].*

Short Vowels	/ae/ /uh/ /ih/
Long Vowels	/ae:/ /uh:/ /ih:/
Diphthongs	/aw/ /ay/

sung. We start to observe a pattern which is more pronounced in Arabic since the majority of words are considered to have some root, generally a tri-lateral root such as in *k-t-b*, the source word for *write*. The letters in these roots are interwoven with vowels (diacritics) to form different words, and are further combined with affixes for even more words. It is estimated that Arabic has 5,000 - 6,500 roots [11]. Examples of this morphology can be viewed in Table 2.5 with '+' indicating an affix.

2.2 Automatic Speech Recogntion

A speech recognizer is a system that transcribes speech into text, a tool used to decode what was said. Automatic Speech Recognition (ASR) systems have certain features which are composed of two parts: training, and decoding. To train an ASR we need a set of training waveforms and their transcripts in order to build the acoustic, lexical, and language models. The acoustic model captures statistics on phoneme realization, the lexicon contains mappings of words to their underlying acoustic units, and the language model defines the likelihood of one word following another. Once these

Table 2.5: *Examples of Morphology in Arabic*

No Diacritics	Diacritics	Meaning
Inflectional Morphology		
ktb	kataba	he wrote
KAtb	kAtaba	he corresponded
ktb	kutiba	it was written
ktAb	kitAb	book
ktb	kutub	books
KAtb	kAtib	writer
ktb	kut~ab	writers
Aktb	Auktub	write
Derivational Morphology		
AlktAb	Al+kitAb	the book
bAlktAb	bi+Al+kitAb	by the book
mktb	maktab	office
mktbp	maktabah	library
mktwb	maktuwb	written

models have been built we can recognize speech using a decoder [Fig 2-1].

To formalize the above description [12], an ASR transcribes the most likely spoken words W given acoustic observations O. We have the observation of acoustic input,

$$O = o_1, o_2, o_3, ..., o_t$$

and a string of words (or sentence),

$$W = w_1, w_2, w_3, ..., w_n$$

The most common technique for speech modeling is as a probabilistic system through the use of Hidden Markov Models (HMM). Thus we have,

$$W^* = \arg\max_W P(W|O) = \arg\max_W P(O, W) \tag{2.1}$$

This $P(O, W)$ can be further decomposed into a sequence of subword units U with the assumption that acoustics O are conditionally independent of the words W, given U.

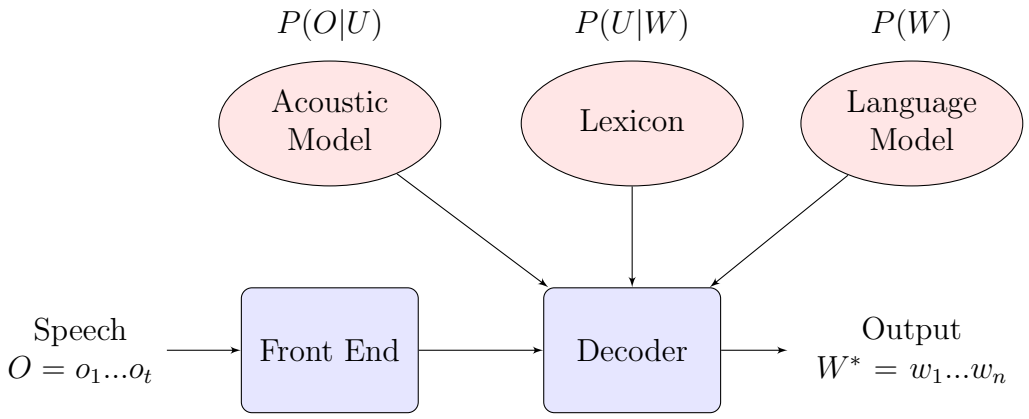

Figure 2-1: *Components of a Speech Recognition System*

$$P(O, W) = \sum_{U} P(O|U)P(U|W)P(W) \tag{2.2}$$

The observation likelihood $P(O|U)$ is the acoustic model, $P(U|W)$ is the lexicon, and the prior probability $P(W)$ is the language model.

These components can be represented using Finite State Transducers (FST) by composition building a search space for decoding. An FST is a graph that can represent distributions such as $P(U|W)P(W)$ in the form $L \circ G$. There are initial and final states with each edge representing an input, output, and weight [13].

The decoder computes the most likely word sequence for each utterance. The Viterbi and A* Algorithm are commonly used to output this 'best' hypothesis based on the information contained in the acoustic, language, and lexical models [12]. In practice, the Viterbi search is typically used to find the most likely U and W given O.

$$U^*, W^* = \arg\max_{U,W} P(U, W|O) \tag{2.3}$$

2.2.1 Acoustic Model

The acoustic model $P(O|U)$ models the probability of an observation sequence O given the subword unit sequence U. Assuming the observation feature vector o_t is

normally distributed, it is often mapped to a probability function $P(o_t|u_k)$ of M weighted Gaussian densities given subword unit u_k, formally known as a Gaussian Mixture Model (GMM) [14].

$$P(o_t|u_k) = \sum_{i=1}^{M} \alpha_i \mathcal{N}(o_t|\mu_i^k, \Sigma_i^k) \tag{2.4}$$

o_t is a D-dimensional feature vector,

α_i are mixture weights that satisfy $\sum_{i=1}^{M} \alpha_i = 1$,

μ_i^k is the mean vector,

Σ_i^k is the covariance matrix,

$\mathcal{N}(o_t|\mu_i^k, \Sigma_i^k)$ are component Gaussian densities of the form,

$$\mathcal{N}(o_t|\mu_i^k, \Sigma_i^k) = \frac{1}{\sqrt{(2\pi)^D \Sigma_i^k}} \exp\Big\{ -\frac{1}{2}(o_t - \mu_i^k)^T (\Sigma_i^k)^{-1} (o_t - \mu_i^k) \Big\} \tag{2.5}$$

The feature vector of o_t is often represented by Mel Frequency Cepstral Coefficients (MFCCs) [12, 15], which capture the spectral short-time distribution of energy using a frequency scale that approximates the cochlea of the human ear.

2.2.2 Lexicon

The lexicon $P(U|W)$ maps words W to their subword units U. Subword units are often phoneme or phone units, and as we will see later in this thesis, they can also be grapheme units. Grapheme units correspond to a written character in a word such as *cat* mapping to /c a t/ rather than the phonemic /k ae t/. Phonemes are the smallest units that would lead to a change in meaning such as the *c* and *b* in *cat* and *bat*.

2.2.3 Language Model

The language model $P(W)$ attempts to capture the word sequence characteristics of a language by modeling over all possible sequences of words in a vocabulary. It is conventionally computed using n-grams which can be a unigram, bigram, trigram or of higher order. The unigram is the relative occurrence of a word w_i in the corpus,

while a bigram is the estimate of w_i conditioned on w_{i-1} and normalized by all the occurrences of w_{i-1}. So the maximum likelihood estimate of $P(w_i|w_{i-1})$ would be,

$$P(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})} \tag{2.6}$$

The *tri*-gram model is similar except we condition on the two preceding words. The general formalization is,

$$P(w_i|w_{i-(n-1)}, ..., w_{i-1}) = \frac{count(w_{i-(n-1)}, ..., w_i)}{count(w_{i-(n-1)}, ..., w_{i-1})} \tag{2.7}$$

When building language models it is important to factor in words that have not been observed in the training data, so smoothing is introduced to allow for a more realistic probability distribution where non-zero probabilities can be assigned to words and word sequences. Numerous methods exist such as Good-Turing, Kneser-Ney, Witten-Bell, etc. [16].

2.3 Data Set

A range of data exists for Arabic such as telephone speech, broadcast news, and broadcast conversations. Data is also available for Arabic dialects and MSA. These include the NEMLAR corpus of broadcast news, ECA Callhome Egyptian Arabic telephone speech, Iraqi conversational telephone speech, and GALE broadcast conversation [17, 18, 19, 20, 21]. Having access to several of these corpora we were most interested in working on wide-band data, that is reasonably clean, well transcribed, MSA speech, and publicly available to other researchers in the field. Allowing for any more variables such as spontaneous speech, and dialects, would be too broad for the scope of this work, which are, in and of themselves, unique domains of research.

We settled with using the recently re-released GALE broadcast conversation phase 1 and 2 datasets which contain 200 hours of mixed broadcast conversation and broadcast news data [20, 21]. The transcripts are not diacratized, containing 1.5 million words, and 106K vocabulary.

Our experiments use only the data labeled 'report'. This is mostly scripted speech in MSA. The rest of the data is conversation which is beyond the scope of this thesis. The data were divided into Training, Development, and Evaluation sets for all experiments used in this thesis.

- Train set: 70 hours.
- Developmet set: 1.4 hours with audio from the show 'ALAM_WITHEVENT'
- Evaluation set: 1.7 hours with audio from the show 'ARABIYA_FROMIRAQ'
- Text is not diacratized, with 500K words, and 61K vocabulary.

2.4 Experimental Setup

The Kaldi Speech Recognition Toolkit was used to build the ASR system with triphone GMM acoustic models. A trigram language model was built using modified Kneser-Ney discounting with the SRILM Toolkit. The transcripts were processed using the MADA+TOKAN Toolkit to automatically diacratize and tokenize words. More details are provided in the following sections.

2.4.1 Kaldi Speech Recognition Toolkit

We use the Kaldi Speech Recognition Toolkit to run experiments, due to its versatility and public availability [22]. Its framework is built around finite-state transducers, and supports a number of feature such as MFCCs, and Perceptual Linear Prediction (PLP). Techniques applied to features include Vocal Tract Length Normalization (VTLN), Cepstral Mean and Variance Normalization (CMVN), Linear Discriminant Analysis (LDA), and Maximum Likelihood Linear Transform (MLLT). Kaldi also supports GMM based acoustic modeling, and both context-independent, and context-dependent phone modeling using Hidden Markov Models (HMMs) topology. Speaker adaptation techniques such as Maximim Likelihood Linear Regression (MLLR), and Speaker Adaptive Training (SAT), are also available.

Acoustic Model

Throughout this thesis, triphone context-dependent GMM-HMM models are used that are based on MFCC features. These were seeded from monophone models that were built using a flat start initialization according to the following standard recipe from the Kaldi Toolkit.

1. Build monophone models using a flat start initialization with 39-dimensional MFCC features (delta, delta-delta) applying CMVN.

2. Create forced alignments.

3. Build triphone models.

4. Build forced alignments using latest models.

5. Build triphone models using MFCC features applying LDA and MLLT.

6. Build forced alignments using latest models.

7. Build triphone models using MFCC+LDA+MLLT features, applying fMLLR.

We use 4,000 states in the HMM, and 128,000 GMMs.

Language Model

The language model remains the same for most of this thesis (Chapter 3 and part of Chapter 4). It is a trigram built on the nondiacratized training transcript with modified Kneser-Ney discounting using the SRILM Toolkit [23]. The text contains 500K words with a vocabulary of 61K words. The Out Of Vocabulary (OOV) rate is 5.27% on the combined Development (5.23%) and Evaluation (5.31%) set. The language model described above is used as the baseline. Chapter 4 explores the impact of language models built over different formats of the training transcript, the details of which are explained there.

2.4.2 Natural Language Processing

Some toolkits exist to automatically diacratize and tokenize Arabic text. The Standard Arabic Morphological Analyzer (SAMA), an updated version of the commonly used Buckwalter Arabic Morphological Analyser (BAMA) produces all possible diacratizations and morpheme boundaries of a word by processing all possible stem-affix variations [24]. The Stanford Word Segmenter is another toolkit that tokenizes Arabic text [25]. The MADA+TOKAN Toolkit scores a list of potential diacratizations for a given nondiacratized word that exists in the SAMA or BAMA database. The toolkit considers the context of the word with 19 morphological features using Support Vector Machines (SVMs) for classification [26].

We work with the MADA+TOKAN Toolkit as it conveniently diacratizes and tokenizes text, and accommodates different configurations for this process. Specifically, we use MADA+TOKAN Toolkit 3.2 with SAMA 3.1 on default settings, tokenizing according to the Penn Arabic Treebank (ATB) Standard 3.1 [27].

2.4.3 Evaluating performance

Throughout this thesis we use several types of texts for language modeling: nondiacratized, diacratized, and morpheme-based. All final evaluations are performed on the text format that came with our GALE corpus. The text is in the Romanized version of the Arabic script, using the Buckwalter transliteration scheme [28]. This scheme allows for one-to-one mapping between the Arabic and Roman script, therefore, there should be no loss of information, and converting between scripts should be straightforward.

To evaluate ASR performance we use the Word Error Rate (WER) metric to compare differences between the reference and hypothesis transcript. We also conduct statistical significance testing on the ASR hypotheses, by performing the Matched Pair Sentence Segment Word Error (MAPSSWE) using the NIST Scoring Toolkit [29].

Chapter 3

Lexical Modeling

Manually crafted Arabic lexicons are hard to come by, and with words lacking diacritics, pronunciation modeling becomes a challenging venture due to the lack of a lexical gold standard, and ambiguity in word pronunciations underlying a nondiacratized word. One solution is to emulate a human's ability to distinguish pronunciations by evaluating potential diacratizations from the context of a word. Once a diacratization of a word has been established, pronunciation rules can be implemented to capture the articulation of phonemes given their context. This chapter investigates the impact of modeling diacritics and the use of pronunciation rules in the lexicon in an ASR system.

3.1 Related Work

Initial work by *Billa et al.* built acoustic models over 40 hours of Arabic data without explicitly modeling diacritics. They go on to improve their ASR system by increasing data for language modeling increasing their lexicon size, and adding 20 hours of speech data. They use a nondiacratized graphemic lexicon [30].

Afify et al. compares the performance of graphemic lexicons that do, and do not include diacritics. The words are diacratized based on the Buckwalter Arabic Morphological Analyzer (BAMA). The authors compare the performance of versions 1.0 and 2.0. Using all possible diacratizations they train over 100 hours of data and

decode using the same lexicon. The results show that diacritics improve performance with BAMA 2.0 producing the best results. *Afify et al.* propose that this may be because 2.0 mainly differs from 1.0, by providing more diacratizations for word endings that include nunnation [31].

Continuing from initial work in [32], *Messaoudi et al.* investigate the use of diacratized language modeling in ASR performance [33]. They build a pronunciation lexicon from diacratized and nondiacratized data. They use BAMA to diacratize the nondiacratized data. They also apply rules to model word endings based on the prefix of a word, glottal stops, and the definite article *Al*. Unfortunately, the exact rules were not made clear enough for implementation. The lexicon is for the most part graphemic.

Ng et al. investigate a technique for Arabic morphological decomposition based on a word's Part Of Speech (POS) tag, extending previous work where only word-level information was considered [34, 35]. They do not decompose the top 128,000 most frequent decomposable words. The lexicon is graphemic using diacratized information from BAMA, with acoustic models built over 1,400 hours of GALE data.

Mangu et al. train on 1,800 hours of GALE data. They build the training dictionary from the top scoring MADA diacratizations with rules of *Biadsy et al.*, as well as using the second ranked diacratization if within a certain threshold [36, 6]. The decoding lexicon maps nondiacratized words to diacratized words of the top 2 MADA rankings.

Vergyri et al. describe work on 1,100 hours of GALE data [37]. Their training lexicon is constrained to a single MADA diacratization, is graphemic based, with some rules which they don't expand on. The test lexicon maps nondiacratized words to MADA diacratizations found in the training data. They show that text processing impacts WER.

El-Desoky et al. investigate morpheme based lexicons and language modeling using 1,100 hours of GALE data [38]. They experiment using two different types of language models, morphemic, and with and without diacritics, comparing these to a baseline of word-based nondiacratized language model. They experiment to find

the N most common frequent decomposable words that would minimize WER. For the diacratized language model, the top N most common decomposable words are not diacratized, nor are the affixes, although the lexicon maps to all their potential graphemic pronunciations. This does not outperform the equivalent nondiacratized experiments.

With regards to pronunciation rules, some work exists on modeling phonemes and phones for applications in MSA Text-To-Speech (TTS) systems, namely *Ahmed* and *El Imam*, which we will explore later in this chapter [4, 5]. On the ASR front, *Biadsy et al.* develop pronunciation rules for MSA based on diacratized words from the MADA+TOKAN Toolkit. Training over 100 hours of data, they compare the performance of their system at the phone level, as well as the word level, showing that performance improves with these rules. They also study the impact of using the top 1 or 2 MADA pronunciations in their training and decoding lexicon, where constraining the training data to the top MADA pronunciation, and decoding over the top 2, provides the best results. The authors were dealing with noisy data with high WERs in the 40s, which may have interfered with gains in lexical modeling [6].

The existing literature indicates that there is not a standard approach towards lexical modeling. Automatically diacratized graphemic lexicons are commonly used [31, 34, 35, 38]. Less common is work like *Billa et al.* that uses a nondiacratized lexicon [30]. Also, *Afify et al.* experiment with graphemic lexicons that are diacratized and nondiacratized [31].

A few researchers choose to apply pronunciation rules, such as *Messaoudi et al.*, *Vergyri et al.*, and *Mangu et al.* [32, 37, 36]. The first two use their own rules, with *Mangu et al.* applying the rules of *Billa et al.*. *Billa et al.* investigates pronunciation rules in the area of Arabic ASR [6], which we notice to be an uncommon research endevour. We find that rarely are investigations on the impact of diacritics and pronunciation rules conducted and compared. This provides good motivation for investigating the different lexical modeling techniques under a single setup.

3.2 Developing a Lexicon

The upcoming sections describe experiments performed on the lexicon at two levels of resolution. First we investigate the effect of diacritics on the lexicon, and how the different classes behave in this regard. Next, we evaluate the effect of pronunciation rules derived from Arabic ASR and TTS literature. We are motivated by the observation that past work in the field touches on diacritics, and sometimes pronunciation rules, but rarely is a systematic evaluation performed.

3.2.1 Diacritics

We first explore the influence of diacritics on ASR performance. Diacritics are expressed in their Romanized version (using Buckwalter transliteration) as in Table 3.1. They are organized based on their characteristic for representing short vowels, geminates ('twinning'/elongation of a sound), and nunnation (pronouncing /n/ at the end of words).

Table 3.1: *Diacritics in the Arabic Language in Both Buckwalter and Arpabet Representation with Examples of Their Use.*

Category	Short Vowels			
Diacritic	a	u	i	o
Arpabet	/ae/	/uh/	/ih/	null
Example	*kataboti* /k ae t ae b t ih/ - you wrote.			
Category	Geminates			
Diacritic	~ (tilde)			
Example	*kataba* /k ae t ae b ae/ - he wrote. *kat~aba* /k ae t t ae b ae/ - he made to write.			
Category	Nunnations			
Diacritic	F	K	N	
Arpabet	/ae n/	/uh n/	/ih n/	
Example	*kitAban* /k ih t ae: b ih n/ - a book.			

3.2.2 Diacratized Text

Let us form some perspective on the prevalence of diacritics at the textual level. Table 3.2 displays statistics on the frequency of each character in the diacratized

corpus text. By far, any single short vowel diacritic (*a, i, u*) is more prevalent than the majority of consonants. Geminates and nunnation fall to the background. Also common are the consonants corresponding to long vowels (*A, y, w*).

Table 3.2: *Arabic Character Distribution in GALE Corpus*

Char	Freq (%)	Char	Freq (%)	Char	Freq (%)	Char	Freq (%)	Char	Freq (%)
a	11.94	w	2.60	>	1.22	<	0.49	}	0.25
i	10.05	r	2.39	h	1.20	S	0.44	o	0.20
A	7.36	t	2.14	q	1.08	Y	0.42	‘	0.19
l	5.68	E	1.69	H	0.93	x	0.39	g	0.18
y	3.98	b	1.68	k	0.89	z	0.32	’	0.17
u	3.59	p	1.64	j	0.66	F	0.31	N	0.15
~	3.04	d	1.51	K	0.58	D	0.31	Z	0.07
m	2.99	s	1.29	$	0.53	v	0.30	&	0.06
n	2.71	f	1.27	T	0.51	*	0.27	|	0.05

We summarize the frequency of diacritics in Table 3.3 with short vowels constituting 25% of the characters in the text. Geminates constitute 3% and nunnation 1%. Consider that when short vowels are not specifically modeled in the lexicon, the acoustic models must be modeling more than one phoneme for a given consonant. This is a large amount of information that is not being differentiated potentially, leading to ambiguity in the models.

Table 3.3: *Frequency of Diacritics in GALE Corpus*

	Short Vowels				Geminates	Nunnations		
Diacritics	a	u	i	o	~	F	K	N
Frequency (%)	11.94	3.59	10.05	0.20	3.04	0.31	0.58	0.15

3.2.3 Experimental Setup

Acoustic and Language Models

To test the impact of modeling diacritics we build acoustic models over a number of lexicons and evaluate their performance. We use the baseline ASR setup described in Section 2.4, with HMM-GMM triphone models based on MFCCs, and a trigram language model.

Lexical Models

The lexicons we investigate are grapheme-based, and map the nondiacratized vocabulary from the training text to their nondiacratized form, and diacratized form, modeling four combinations of diacritics. Table 3.4 displays examples of each format with instances of multiple entries for a given word. This shows that the inclusion of certain diacritics may produce multiple pronunciations per word. For example, the word *ktb* when diacratized can be realized as /k a t a b a/ and /k a t~ a b a/. The short vowels and geminates produces different words, which in turn are pronounced differently. Otherwise it would only be modeled as /k t b/, which is ambiguous as to what the underlying word would be.

Table 3.4: *Example Entries in Lexicon*

Lexicon	Vocab	Grapheme
No diacritics	ktb	k t b
	ktAb	k t A b
Short Vowels Only	ktb	k a t a b a
	ktAb	k i t A b
No Geminates	ktb	k a t a b a
	ktAb	k i t A b
	ktAb	k i t A b N
No Nunnations	ktb	k a t a b a
	ktb	k a t~a b a
	ktAb	k i t A b
All Diacritics	ktb	k a t a b a
	ktb	k a t~a b a
	ktAb	k i t A b
	ktAb	k i t A b N

No Diacritics. This lexicon maps every word in the vocabulary of the nondiacratized training text to its grapheme form.

Diacritics - Short Vowels Only. This lexicon only models short vowels (*a, u, i*). It does not model nunnations (*F, K, N*) or geminates (*b~, f~, l~, . . .*).

Diacritics - No Geminates. This lexicon models short vowels and nunnations (*a, u, i, F, K, N*), but not geminates (*b~, f~, l~, . . .*).

Diacritics - No Nunnations. This lexicon models short vowels and geminates (*a, u, i, b~, f~, l~, ...*), but not nunnations (*F, K, N*). Keep in mind that modeling geminates means we model the word *kat~aba* in its grapheme form as */k a t~ a b a/* rather than */k a t ~ a b a/* or */k a t t a b a/*. The */k a t ~ a b a/* form would have modeled the geminate character as a unique phonetic unit rather than associating it with its consonant. */k a t t a b a/* would have represented the geminate consonant as a double consonant.

All Diacritics. Every vocabulary word in the training text is mapped to its diacratized grapheme form. The diacritics (short vowels, geminates, nunnations) are modeled as {*a, u, i, F, K, N, b~, f~, l~, ...*}.

Table 4.4 summarizes the vocabulary and grapheme size of each lexicon. Nondiacratized words may have more than one potential diacratized form. The table shows that the number of Pronunciations Per Word (PPW) is around 1.28. The nondiacratized lexicon does not distinguish between diacratized words, therefore it has a unique mapping between each word in the vocabulary and its graphemic representation.

Table 3.5: *Lexicon Sizes using Different Diacritics*

Lexicon	Vocab	Grapheme	PPW
Baseline - No diacratics	61 K	61 K	1
Short Vowels Only	61 K	77 K	1.25
No Geminates	61 K	79 K	1.28
No Nunnations	61 K	77 K	1.25
All Diacritics	61 K	79 K	1.28

Baseline

We use the nondiacratized lexicon as the baseline, with a WER of 25.1% and OOV of 5.31% on the Evaluation dataset. This seems to be a reasonable starting point, as the WER value falls within the range of results found in the literature, such as *Xiang et al.* and *Vergyri et al.* [34, 37]. Table 3.6 lists works in the domain of Arabic ASR, along with datasets used to train acoustic models, and the range of WERs presented in these papers.

Xiang et al. use a 64K vocabulary as the baseline, an OOV rate of 4.18%, and a diacratized lexicon. Acoustic modeling is performed over 150 hours of broadcast bews data. Their WER is 24.6% which is close to our baseline [34].

For an equivalent vocabulary size of 64K and an OOV of 5.36%, a diacratized lexicon, and acoustic modeling over GALE data, *Vergyri et al.* have a WER of 33.6%. Although higher than our baseline, it seems that the nature of that test set returned WERs in the 30s, since even with an expanded vocabulary and OOV rate of 1.07% the resulting WER was 32.6% [37].

Our baseline seems reasonable considering that we are not using a diacratized lexicon, and are using between half, and less than a tenth of the data in the previous works. This is while keeping in mind the different techniques used for acoustic modeling, differences in datasets, language model sizes, and n-grams.

Table 3.6: *Arabic WERs in Literature*

Reference	Duration (hrs)	Dataset	WER (%)
Biadsy et al. [6]	40	TDT4	43.1 - 47.3
Billa et al. [30]	60	BBN in-house News	15.3 - 31.2
Afify et al. [31]	100	FBIS, TDT4	14.2 - 21.9
Messaoudi et al. [33]	150	FBIS, TDT4, BBN in-house News	14.8 - 16.0
Xiang et al. [35]	150	FBIS, TDT4, BBN in-house News	17.8 - 31.8
Vergyri et al. [37]	1,100	GALE	8.9 - 36.4
El-Desoky et al. [38]	1,100	GALE	13.9 - 16.3
Ng et al. [34]	1,400	FBIS, TDT4, GALE, Iraqi Arabic	10.2 - 18.8
Mangu et al. [36]	1,800	GALE	7.1 - 12.6

3.2.4 Results

Table 3.7 displays the results of training and decoding using these lexicons that vary only in diacritics. We take the baseline to be the lexicon with no diacritics. Inserting acoustic units in the lexicon to model diacritics outperforms the nondiacratized lexicon by 1.7% absolute WER. This shows that modeling diacritics as part of consonants 'works', but is not as effective as having diacratized lexicon entries. Even partially including diacritic information in the lexicon helps.

Short vowels, the most common diacritics, are at a disadvantage when other diacritics are not modeled, resulting in a 1.0% absolute WER improvement over the baseline. Additionally, nunnation, the most uncommon diacritic, has a more significant impact than would be anticipated. When modeled along with short vowels, an absolute WER improvement of 1.9% is obtained. Contrasting this with modeling short vowels, we see an additional 0.9% absolute WER improvement over the 1.0% absolute improvement achieved by short vowels.

Moreover geminates, more common than nunnations, do not have as much of an impact. Geminates produce an absolute improvement of 1.2% when modeled with short vowels. Not as impressive as the 1.9% absolute WER improvement of geminates plus short vowels. In other words, geminates help when nunnations are missing, but offer no gain when nunnations are modeled. There is actually a loss when geminates are modeled with other diacritics. We observe this with a 1.7% absolute WER improvement when all diacritics are modeled. It could be that, although geminates compose 3% of the characters in a text, the acoustic units modeling them are prone to sparsity. This is because geminate consonants are rare. So the 3% occurrence of geminates are being divided over 30 possible consonants. This corresponds to the high number of acoustic units, occurring whenever geminates are modeled.

Overall, the combined effect of modeling the different classes of diacritics is greater than modeling its parts. However, geminates seem to have a negative impact when combined with all other diacritics. All results were found to be statistically significant with $p < 0.007$, using MAPSSWE.

Table 3.7: *Impact of Modeling Diacritics in Lexicon on ASR Performance.*

Diacritics	# phones	Freq. (%) in Text	Dev WER (%)	Eval WER (%)	Significance at $p <$
Baseline - No Diacritics	36	-	24.2	25.1	-
Short Vowels only	39	25	23.4	24.1	0.007
No geminates	42	3	22.6	23.2	0.001
No nunnations	69	1	22.8	23.9	0.001
All Diacritics	72	29	22.6	23.4	0.001

3.3 Pronunciation Rules

We have established the benefits of including diacritics in the lexicon, so let us consider taking this further by modeling phonemes in the lexicon, and applying pronunciation rules. Intuitively, there are many graphemes that may correspond to multiple phonemes, and various realizations of these phonemes where it would be more useful to include this information as additional acoustic units in the lexicon. We experiment with a number of pronunciation rules available in the literature by *Ahmed*, *El Imam*, and *Biadsy et al.* under a single setup for evaluation [4, 5, 6].

3.3.1 Experimental Setup

As before, we build acoustic models over the lexicons produced by the different pronunciation rules and evaluate their performance. The language model remains the same throughout, which is a trigram built on the nondiacratized training text with modified Kneser-Ney discounting using the SRILM Toolkit. The details for both the acoustic and language models are in Section 2.4.

The lexicon maps the nondiacratized vocabulary from the training text to their diacratized form after the alterations introduced by pronunciation rules. Thus the entries are composed of either phonemes or phones depending on the rules applied.

Rules I by *Ahmed* was originally developed for MSA TTS systems. They cover glottal stops, short vowels, coarticulation of the definite article *Al*, nunnation, diphthongs, word ending *p* (/t/), as well as phones in regular, pharyngealized, and emphatic contexts, a few geminates, aspirated representation of phones, and retroflexed vowels. Only cross-word rules were not implemented [4]. Table 3.10 lists the rules that were implemented.

Rules II by *El Imam* was also developed for MSA TTS systems, and covers glottal stops, short vowels, coarticulation of the definite article *Al*, nunnation, diphthongs, pharyngealized vowels and non-emphatic consonants, a few rules for unvoiced stops, and without modeling geminates [5]. Table 3.11 details the

rules that were implemented.

Rules III by *Biadsy et al.* describes rules for representing glottal stops, short vowels, coarticulation of the definite article *Al*, nunnation, diphthongs, word ending *p* (/t/), and case endings, while ignoring geminates [6]. Table 3.12 presents the rules that were implemented.

Table 3.8 summarizes the vocabulary and grapheme size of each lexicon. Nondiacratized words may have more than one potential diacratized form. The table shows that the number of Pronunciations Per Word (PPW) is around 1.28. The nondiacratized lexicon does not distinguish between diacratized words, therefore it has a unique mapping between each word in the vocabulary and its graphemic representation. When pronunciation rules are built on top of the diacritics they maintain the same PPW since most words have a single phone/phonemic realization. Rules III is the exception, it allows for variations in pronunciations. In particular, it models the endings with and without vowels, and *p* (/t/). The result is that the Rule III lexicon has 1.85 PPW.

Table 3.8: *Sizes of Lexicons Produced by Pronunciation Rules*

Lexicon	Vocab	Grapheme	PPW
Baseline - No diacratics	61K	61K	1
All Diacritics	61K	79K	1.28
Rules I - *Ahmed*	61K	78K	1.27
Rules II - *El Imam*	61K	78K	1.27
Rules III - *Biadsy et al.*	61K	114K	1.85

3.3.2 Results

After building each lexicon according to their pronunciation rules, training their corresponding acoustic models, and then decoding with that same lexicon, the results are as recorded in Table 3.9. We take the baselines, as before, to be the nondiacratized graphemic based lexicon assessed over the Evaluation data. All lexicons perform better than the baseline, with two out of the three performing better than the diacratized

graphemic lexicon.

Table 3.9: *Impact of Lexicon Pronunciation Rules on ASR Performance.*

Lexicon	# phones	Dev WER (%)	Eval WER (%)	Significance at $p <$
No Diacritics	36	24.2	25.1	-
Diacritics	72	22.6	23.4	0.001
Rules I - *Ahmed*	135	22.9	24.0	0.004
Rules II - *El Imam*	63	22.4	23.3	0.001
Rules III - *Biadsy et al.*	34	22.3	22.7	0.001

The poorest performing lexicon is that based on Rules I. It performs better than the baseline by 1.1% absolute, but it does not match that of the diacratized graphemic lexicon. This may be due to data sparsity when modeling the acoustics of these phones, with almost double the number of phones compared to the diacratized grapheme lexicon, and almost four times that of the baseline.

The other two rule-based lexicons fair better. Rules II slightly outperforms the diacratized graphemic lexicon with a 1.8% absolute WER improvement over baseline. Rules III performs the best with a 2.4% absolute WER improvement. Interestingly, Rules III manages this with the smallest number of phones. This is an even lower number of acoustic units than the baseline lexicon, which does not model diacritics. Therefore, boosts in performance seem more to do with modeling key phonemes rather than trying to model the nuanced realizations of dozens of phones. Phone modeling, which is the nature of the rules developed for MSA TTS systems, is more applicable to those applications, and do not necessarily carry over to decoding speech.

Although Rules III are simpler than the other rules, *Biadsy et al.* apply certain rules which may better accommodate the nature of spoken words. First, they model words with and without their vowel, and p endings which would capture cross-word coarticulation. Second, they ignore modeling geminates, whereas the other rules double the consonant with which the geminate is associated. This may not necessarily be realistic as to how geminates are realized, since geminates look the same as any consonant, except they are longer in duration [39]. It is important to note that rules III produced 1.86 PPW, whereas the other lexicons are composed of less than 1.3

PPW. Having more than 1 PPW would allow flexibility in the choice of pronunciations during both training and decoding.

Overall, there are benefits to building rule-based lexicons on top of the diacratized grapheme lexicon that would already exist. It would seem that it hurts to model phones too finely with the size of the audio we are working with (70 hours). Simple rules that attempt to capture coarticulation in speech, and ignore sparser data such as geminates, seem to be most effective. All results were found to be statistically significant with $p < 0.004$, using MAPSSWE.

3.4 Summary

In this chapter we experimented with lexicons at two levels. We first examined the role of diacritics in a grapheme-based lexicon and how each class of diacritic affected ASR performance. We found that each class of diacritics, short vowels, geminates, and nunnations, affected performance. It was found that modeling the whole provided a clear reduction in WER, with a 1.7% absolute improvement over the baseline nondiacratized graphemic lexicon.

The next level of lexical modeling investigated the application of pronunciation rules to model phonemes and phones. There was a negative return in improvement when attempting to include too many acoustic units in the lexicon. While it was found that rules helped improve ASR performance, it was simple rules that proved to be the most effective, with a 2.4% absolute WER improvement over the baseline lexicon.

It was also observed that the results were consistent when not including geminates in graphemic modeling of diacritics, and in pronunciation rules. This indicates that geminates do not seem to be valuable enough to model, probably due to their rarity.

Table 3.10: *Summary of Rules I by* Ahmed *[4].*

Rule	Source	Target
	Letter to Phoneme	
Glottal Stop	/[&><']/	/q/
Long Vowel	/[|{]/	/ae:/
Long Vowel	/ae Y/	/ae:/
	/Y/	/ae:/
Diphthongs (*Waw Al Jama'a*)	/uh e ae:$/	/uw/
Definite Article *Al* (Sun Letters)	/^ae: l $*sun*/	/^ae: $*sun*/
	$*sun* = {t, th, d, dh, t, z, s, sh, tt, ss, dd, zh, l, n}	
Nunnation	/F/	/ae n/
	/K/	/ih n/
	/N/	/uh n/
Coarticulation	/ae n b/	/ae m b/
Coarticulation - Geminates	/dd tt/	/dd~/
	/dd t/	/tt~/
	/d t/	/t~/
Geminates	/$*cons*~/	/$*cons* $*cons*/
	$*cons* = { b, d, dd, t, tt, k, kq, q, f, th, dh, zh, z, s, ss, sh, gh, kh, ai, h, hh, jh, m, n, w, l, r, y}	
Dark 'L'	/l l l '/	/L L '/

Phoneme to Phone

Short Vowels	/{tt, dd, ss, zh} ae/	/\1 ah/
	/{kq, kh, gh, r} ae/	/\1 ah/
	/ae {tt, dd, ss, zh, kq, kh, gh, r}/	/aa1 \1/
	/ae {tt, dd, ss, zh, kq, kh, gh, r} $*cons*/	/aa1 \1 \2/
	/{n,m} {ae,uh.ih} {n,m}/	/\1 {an,un,in} \2/
	/{kq, kh, gh, r} {uh,ih}/	/{kq, kh, gh, r} {ux,ix}/
Long Vowels	/uh w/	/uh:/
	/ih y/	/ih:/
	/{kq,kh,gh,r} ae:/	/\1 aa:/
	/L {` ,ae:}/	/L aa:/
	/{tt,dd,ss,zh} ae:/	/\1 ah:/
	/ah {tt,dd,ss,zh,kq,kh,gh,r}/	/aa1 \1/
	/ah {tt,dd,ss,zh,kq,kh,gh,r} $*cons*/	/aa1 \1 \2/
	/{n,m} ae: {n,m}/	/\1 an: \2/
	/{n,m} uh: {n,m}/	/\1 un: \2/
	/{n,m} ih: {n,m}/	/\1 in: \2/
Diphthongs	/{kq,kh,gh,r} {ae, aa, ae:, aa:, aa1}/	/\1 ay/
	/$*cons* {ae, aa, ae:, aa:, aa1} y/	/\1 ay1/g
	/{tt,dd,ss,zh} {ae, aa, ae:, aa:, aa1}/	/\1 ay2/
	/y uh:/	/y uw/
	/{tt,dd,ss,zh,kq,kh,gh,r} uw/	/\1 ow/
	/{ae, aa, ae:, aa:, aa1} w/	/aw/

Voiced Fricatives	/z ae:/	/z1 ae:/
	/z $*cons*/	/z1 $*cons*/
	/dh $*cons*/	/dh2/
	/zh $*cons*/	/zh2/
	/gh $*cons*/	/gh2/
	/ai {tt,dd,ss,zh}/	/ai1 \1/
	/{tt,dd,ss,zh} ai/	/\1 ai1/
	/ai {tt,dd,ss,zh} {all vowels}/	/ai1 \1 \2/
	/{tt,dd,ss,zh} {all vowels} ai/	/\1 \2 ai1/
Voiceless Fricatives	/f f/	/f1/
	/f $*cons*/	/f1/
	/s$/	/s1/
	/s $*cons*/	/s1 \1/
	/$*cons* s/	/\1 s1/
	/s $*cons* {all vowels}/	/s1 \1 \2/
	/$*cons* {all vowels} s/	/\1 \2 s1/
	{all vowels} = {ae, ah, aa1, aa:, ay1, ay2, aw, uw, ow, uh, uh:, ih, ih:}	
	/sh {all vowels} $*cons*/	/sh1 \1 \2/
	/$*cons* {all vowels} sh/	/\1 \2 sh1/
	/h {all vowels}/	/h1 \1/
	/{uh, uh:, uw, ow} hh/	/\1 hh1/
	/hh {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/hh2 \1 \2/
	/{tt,dd,ss,zh,kq,kh,gh,r} {all vowels} hh/	/\1 \2 hh2/

Affricates	/jh jh/	/jh1/
	/jh $*cons*/	/jh2 \1/
Nasals	/m m/	/m:/
	/m {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/m1 \1 \2/
	/{tt,dd,ss,zh,kq,kh,gh,r} {all vowels} m/	/\1 \2 m1/
	/{all vowels} n {b,m}/	/\1 nm/
	/n n/	/n1:/
	/n {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/n1 \1 \2/
	/{tt,dd,ss,zh,kq,kh,gh,r} {all vowels} n/	/\1 \2 n1/
	/n {all vowels} {tt,dd,ss,zh}/	/nn1 \1 \2/
	/n {all vowels}/	/nn2 \1/
	/n {t,th,s,sh,jh,dh,z,ss,dd,tt,zh,f,k,kq,q}/	/nn3 \1/
	/n {h,hh,ai,kh,gh}/	/nn: \1/
Resonants	/w w/	/w:/
	/w {all vowels} {kq,ai,gh,r}/	/w1 \1 \2/
	/{kq,ai,gh,r} {all vowels} w/	/\1 \2 w1/
	/w {kq,ai,gh,r}/	/w1 \1/
	/{kq,ai,gh,r} w/	/\1 w1/
	/r$/	/r1/
	/r $*cons*/	/r1 \1/
	/r {ae, ae:, uh, uh:, uw, ow, ux}	/r2 \1/
	/L L/	/L1/
	/y y {uh, uh:, uw}/	/y1 \1/
	/y {all vowels} {tt,dd,ss,zh}/	/y2 \1 \2/

Voiced Stops	/b b/	/bb1/
	/b$/	/bb2/
	/b {b,d,dd,f,h,hh,k,kh,kq,s,ss,sh,t,th,tt}/	/bb2 \1/
	/{bb1,bb2,b} {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/{b1,b2,b3} \1 \2/
	/{tt,dd,ss,zh,kq,kh,gh,r} {all vowels} {bb1,bb2,b}/	/\1 \2 {b1,b2,b3}/
	/d $*cons*/	/d1 \1/
	/d$/	/d2/
	/d {b,d,dd,f,h,hh,k,kh,kq,s,ss,sh,t,th,tt}/	/d3 \1/
	/d {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/d4 \1 \2/
	/{tt,dd,ss,zh,kq,kh,gh,r} {all vowels} d/	/\1 \2 d4/
	/dd ae/	/dd1 \1/
	/dd uh/	/dd2 \1/
	/dd ih/	/dd3 \1/
	/dd {tt,dd,ss,zh,kq,kh,gh,r}/	/dd4 \1/
	/dd !{tt,dd,ss,zh,kq,kh,gh,r}/	/dd5 \1/
	/dd$/	/dd5/
Voiceless Stops	/t $*cons*/	/t1/
	/t t/	/t2/
	/t {all vowels} {tt,dd,ss,zh,kq,kh,gh,r}/	/t3 \1 \2/
	/tt ae/	/tt1 \1/
	/tt uh/	/tt2 \1/
	/tt ih/	/tt3 \1/
	/tt $*cons*/	/tt4 \1/
	/tt$/	/tt5/
	/k ae/	/k1 \1/
	/k uh/	/k1 \1/
	/k $*cons*/	/k2 \1/
	/k {ih, ih:}/	/k3 \1/
	/$*cons* {all vowels} k/	/k4/

	/k {all vowels} $*cons*/	/k5/
	/kq {tt,dd,ss,zh,kq,kh,gh,r}/	/kq1 \1/
	/kq ae/	/kq2 \1/
	/kq uh/	/kq3 \1/
	/kq ih/	/kq4 \1/
	/kq $*cons*/	/kq5 \1/
	/kq $/	/kq5/
	/kq/	/kq2/
	/'/	/q1/
	/>/	/q2/
	/|/	/q3/
	/^</	/q4/
	/^>/	/q5/
R-controlled Vowels	/{tt, dd, ss, zh, kq, kh, gh, r} {ae,ae:} r/	/\1 ar/
	/ih: r/	/yr/
	/{uh,uh:} r/	/ur/
	/{ae,ae:} r/	/ar1/
	/{ih,ix} r/	/ir1/
	/{tt, dd, ss, zh, kq, kh, gh, r} {ih,ix} r/	/\1 ir2/
Pharyngealized Controlled Vowels	/{tt, dd, ss, zh, kq, kh, gh, r} ih ss/	/\1 is/
	/{tt, dd, ss, zh, kq, kh, gh, r} uh ss/	/\1 us/
	/{tt, dd, ss, zh, kq, kh, gh, r} ae ss/	/\1 as/
	/ae ss/	/\1 ass/
	/{tt, dd, ss, zh, kq, kh, gh, r} ih dd/	/\1 id/
	/{tt, dd, ss, zh, kq, kh, gh, r} uh dd/	/\1 ud/
	/{tt, dd, ss, zh, kq, kh, gh, r} ae dd/	/\1 ad/
	/ae dd/	/\1 add/

/{tt, dd, ss, zh, kq, kh, gh, r} ih tt/	/\1 it/
/{tt, dd, ss, zh, kq, kh, gh, r} uh tt/	/\1 ut/
/{tt, dd, ss, zh, kq, kh, gh, r} ae tt/	/\1 at/
/ae dd/	/\1 att/
/{tt, dd, ss, zh, kq, kh, gh, r} ih zh/	/\1 izh/
/{tt, dd, ss, zh, kq, kh, gh, r} uh zh/	/\1 uzh/
/{tt, dd, ss, zh, kq, kh, gh, r} ae zh/	/\1 azh/
/ae zh/	/\1 azzh/
/{tt, dd, ss, zh, kq, kh, gh, r} ih kq/	/\1 ikq/
/{tt, dd, ss, zh, kq, kh, gh, r} uh kq/	/\1 ukq/
/{tt, dd, ss, zh, kq, kh, gh, r} ae kq/	/\1 akq/
/ae kq/	/\1 akkq/
/{tt, dd, ss, zh, kq, kh, gh, r} ae kh/	/\1 akh/
/ae kq/	/\1 akkh/
/{tt, dd, ss, zh, kq, kh, gh, r} ae gh/	/\1 agh/
/ae kq/	/\1 aggh/

Using regular expressions. ^ means first character. /$/ is end of word. $*italics* indicates variable. {a,b, ...} is a set. \# means to save variable in that location number.

Table 3.11: *Summary of Rules II by* El Imam *[5].*

Rule	Source	Target
Letter to Phoneme		
Drop Diacritics (*sukoon*)	/o/	//
Elision	/w ae:$/	/uw/
	/A F/	/F/
End of word *Y*	/Y/	/ae/
Glottal Stop	/'/	/q/
	/>/	/qae/
	/</	/qih/
	/&/	/quh/
	/}/	/qih/
	/|/	/qae:/
Nunnation	/F/	/an/
	/K/	/in/
	/N/	/un/
Definite Article *Al* (Sun Letters)	/^Al$*sun*/	/^Al $*sun* $*sun*/
	/^Al$*sun*~/	/^Al $*sun* $*sun*/
	$*sun* = {t, th, d, dh, t, z, s, sh, tt, ss, dd, zh, l, n}	
Geminates	/$*cons*~/	/$*cons* $*cons*/
	$*cons* = { b, d, dd, t, tt, k, kq, q, f, th, dh, zh, z, s, ss, sh, gh, kh, ai, h, hh, jh, m, n, w, l, r, y}	
Long Vowels	/ae ae:/	/ae:/
	/ih y/	/ih:/
	/uh w/	/uw/
	/uh uw/	/uw/
Diphthong	/ae y/	/ay/
	/ae w/	/aw/
Phoneme to Phone		
Pharyngealized Vowels due to Neighboring Emphatics	/$*emph* $*v*/	/$*emph* $*emph_v*/
	/$*v* $*emph*/	/$*emph_v* $*emph*/
	$*emph* = {tt, dd, zh, ss}	
	$*v* = {ae, uh, ih, ae:, uh:, ih:, aw, ay}	
	$*emph_v* = {ae', uh', ih', ae:', uh:', ih:', aw', ay'}	
Pharyngealized Consonants due to Neighboring Emphatics	/$*emph* $*v* $*non_emph*/	/$*emph* $*v* $*non_emph*'/
	$*non_emph* = {t, d, s, dh, l, r}	
Nasalization of Vowel	/$*v* {m,n}/	/$*v*N {m,n}/
Overlapping *t* or *k* with Long Vowel	/{t,k} {uh:,ih:}/	/{t,k}V {uh:,ih:}/

Using regular expressions. ^ means first character. /$/ is end of word.
$*italics* indicates variable. {a,b, ...} is a set.

Table 3.12: *Summary of Rules III by* Biadsy et al. *[6].*

Rule	Source	Target
Long Vowel (*Dagger Alif*)	/‘/	/ae:/
Long Vowel (*Madda*)	/|/	/ae:/
Nunnation	/AF/	/ae n/
	/F/	/ae n/
	/K/	/ih n/
	/N/	/uh n/
Glottal Stop (*Hamza*)	/[’}&<>]/	/q/
p word ending (*tah-marbuta*)	/p/	/t/
Long Vowel (*Alif Maqsura*)	/Y/	/ae:/
Geminates (*Shadda*)	/~/	//
Diphthongs	/u w $*cons*/	/uw/
	/ih y $*cons*/	/ih:/
	$*cons* = { b, d, dd, t, tt, k, kq, q, f, th, dh, zh, z, s, ss, sh, gh, kh, ai, h, hh, jh, m, n, w, l, r, y}	
Suffix ’uwoA’ (*Waw Al Jama’a*)	/uh w ae:$/	/uw/
Definite Article (*Al*)	/Al/	/ae l/
Word Middle Long Vowel (*Hamzat Wasl*)	/{/	//
Definite Article *Al* (Sun Letters)	/^aw l $*sun*/	/^ae $*sun*/
	$*sun* = {t, th, d, dh, t, z, s, sh, tt, ss, dd, zh, l, n}	

Additional Variants to Lexicon.

p Word Ending (*tah-marbuta*)	/p {ae, uh, ih, F, K, N} $/	//
Short Vowel Word Ending	/{ae, uh, ih}$/	//

Using regular expressions. ^ means first character. /$/ is end of word.
$*italics* indicates variable. {a,b, . . . } is a set.

Chapter 4

Language Modeling

Texts in Arabic generally do not contain diacritics, which makes any given word have multiple diacratizations. Given the context, these potential diacratizations can be narrowed down. The Arabic language is also morphologically rich, leading to many possible combinations of word stems and affixes, causing high Out Of Vocabulary (OOV) rates, which hamper ASR performance. Tackling these factors in language modeling, as we shall see, leads to better ASR performance while reducing the size of the language model. This is vital for use in areas such as on-line ASR, where memory is a significant constraint, and fast performance is crucial. This chapter explores gains made when modeling diacritics in the language model, and modeling at the morpheme level, as opposed to the word level. We also establish our approach when dealing with text normalization.

4.1 Related Work

Continuing from initial work in [32], *Messaoudi et al.* investigate the use of diacratized language modeling in ASR performance [33]. Training over 150 hours of data *Messaoudi et al.* show that a manually diacratized language model outperforms a nondiacratized one. However, including the automatically diacratized language model was found to negatively impact WERs. The automatic diacratization was hypothesized by the decoder.

Ng et al. investigate a technique for Arabic morphological decomposition based on a word's Part Of Speech (POS) tag, extending on previous work by *Xiang et al.*, where only word-level information was considered [34, 35]. They do not decompose the top 128,000 most frequent decomposable words. With acoustic models built over 1,400 hours of GALE data they find that this language model performs better than the baseline word-based language model, and improves over the morpheme language model that was built by decomposing using word-level information.

El-Desoky et al. investigate morpheme-based lexicons and language models using 1,100 hours of GALE data [38]. They experiment using two different types of language models, morphemic, and with and without diacritics, and compare these models to a word-based, nondiacratized language model. They experiment to find the N most common frequent decomposable words that would minimize WER. For the diacratized language model, the top N most common decomposable words are not diacratized, nor are the affixes, although the lexicon maps to all their potential graphemic pronunciations. This does not outperform the equivalent nondiacratized experiments.

The affixes used in the above papers differ from what we will be presenting, while we implement diacratizations in a fuller manner.

4.2 Text Formats

As in the previous chapter, we want to evaluate our ASR system on a single nondiacratized text. However, with language modeling, we will be working with multiple text formats which are nondiacratized, diacratized, words, and morphemes. Table 4.1 lists examples of each text. Note that we use the phrases 'morpheme based text' and 'tokenized text' interchangeably.

Depending on the language model built from these texts, our ASR system will be outputting hypotheses in these formats which requires a little post-processing. For diacratized text we remove the diacritics (*a, u, i, o, ~, F, K, N*), and for the morpheme based text we connect affixes and stems, as indicated by the '+' sign, *w+ ktb* becomes

Table 4.1: *Examples of script styles used in experiments.*

No Diacritics	*wktbt fy ktAbk*
Diacritics	*wakatabotu fiy kitAbuka*
No Diacritics Morphemes	*w+ ktbt fy ktAb +k*
Diacritics Morphemes	*wa+ katabotu fiy kitAbu +ka*

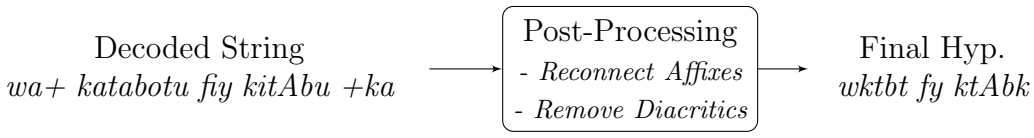

Figure 4-1: *Example of Post-Processing Step after Decoding to Produce Final Hypothesis*

wktb (Figure 4-1). This sounds simple enough except that the MADA+TOKAN Toolkit performs some text normalization, which proved challenging. We describe how we tackled this in the next section.

4.2.1 Text Normalization

One of the challenges of working with Arabic text is the issue of text normalization. There are commonly used words, spelt slightly differently, that do not necessarily affect comprehension, but may not be following the spelling rules of the language [11].

Text in ASR literature are normalized in various ways. For example, the glottal stop can be written in several forms {>,<,',&,}, and is sometimes normalized by being mapped to a single character, as performed by *Billa et al.* [30]. They further show the gains that can be made when evaluating over normalized text, and rebuilding acoustic models based on this text. They achieve an absolute improvement of 2.6% in WER (19.1% to 16.5%). *Messaoudi et al.* work with normalized text, but do not specify in what manner this is implemented [33]. *Ng et al.* normalize text by mapping {<,>,|} to a single character depending, on word position. They also map {Y} in certain words to {y}, and vice versa [34, 35]. *Mangu et al.* normalize text by mapping {{,<,>} to the character {A} [36]. *Vergyri et al.* and *El-Desoky et*

al. perform evaluations on MADA normalized text with *Vergyri et al.*, showing how WERs are impacted by this text normalization [37, 38]. Meanwhile, *Biadsy et al.* don't seem to normalize their text [6]. We find that there is no single standard for text normalization, which forces us to look more deeply at this topic.

Normalization Techniques

Some common normalizations performed by the MADA+TOKAN Toolkit after the removal of diacritics are listed in Table 4.2.

Table 4.2: *Examples of Text Normalizations by MADA+TOKAN Toolkit*

Original	MADA Norm	Validity
l>nh	lAnh	incorrect
|xrh	Axrh	incorrect
Al<ntxAbAt	AlAntxAbAt	correct
$glp	$glh	incorrect
ElY	Ely	incorrect
ybqY	ybqy	incorrect
Al<HtlAl	AlAHtlAl	correct
<bn	Abn	correct

Some normalizations may be valid while others are not. We notice that some even lose information on the way a word is pronounced, if we were to implement pronunciation rules. For example, *l>nh* is pronounced as /l q ae n h/, assuming we don't know the diacritics, while *lAnh* would be /l ae: n h/, which is untrue because a speaker would most likely emphasize the glottal stop /Q/. Another example is the word *ElY* /ai l ae:/ and *Ely* /ai l ih:/. They both exist in the language, but to model only *Ely* hurts the acoustics, which would be modeling /ae:/ and /ih:/ as one unit, whereas /ae:/ and /ih:/ would be modeled as unique units with other word entries in the lexicon.

These normalizations cause the text WER of the MADA diacratized training data to be 3.1%, and the tokenized data to be 24.1%, when compared to the original reference at the word level, and after the removal of diacritics. Thus, we are faced with a number of options. One option would be to normalize all text and work with that,

however, according to the above examples, some normalizations are valid while others are not. Unless we want to establish new rules as part of the evolution of language, we'd best follow language rules already described and codified in the literature [11].

Another option would be to use the normalized text for lexical and language modeling while evaluating on the original Romanized[1] text. The hypothesis produced by the ASR system would then be post-processed to match the format of the original text. A seemingly straightforward technique would be to swap word confusion-pairs due to normalizations found in the training text. We experimented with this on the morpheme based text, after reconnecting stems and affixes, to evaluate at the word level. We found that swapping all confusion-pairs does not lead to a WER of 0% because the pairs are not deterministic. For example, swapping all *Ely* (Ali - common name) with *ElY* (on - preposition) would mean that words correctly meant to be *Ely* would disappear. We looked at the optimum N confusion-pairs that would allow for a minimum WER in a subset of the complete GALE corpus, containing 1.4 million words (out of 1.5 million). Figure 4-2 summarizes the effect of swapping the top 4000 word confusion-pairs out of a possible 39,000 confusion pairs containing 340,000 confused words in total. The optimum N would be at 1000 where 80% of confused words are swapped reducing the WER to 9.8%. Beyond that WER starts to increase again.

It is disheartening to find such a gap between the original and normalized text, so this leads us to search for another solution because this gap potentially causes even more problems. This technique might work reasonably well when the data over which the decoding is performed is of the same nature as that from which the list of confusion-pairs were produced. If not, then new confusion-pairs could be introduced into the hypothesis when they don't originally exist, adding further confusion to the confusion-pairs. For example, if the data is about someone called 'Ali', and *Ely* is correctly decoded many times more often than *ElY* but is swapped due to it

[1]Working with the Romanized text causes no loss of information when converting from the Arabic script because the mappings between both are unique. Why Romanized to begin with? To avoid any hiccups when working with terminals, text processors, and toolkits that may treat 'utf8', and right-to-left script in an unfriendly manner.

being ranked high on the confusion-pairs list, the quality of the hypothesis will have degraded unnecessarily. Moreover, this technique swaps words as isolated occurrences without considering their context.

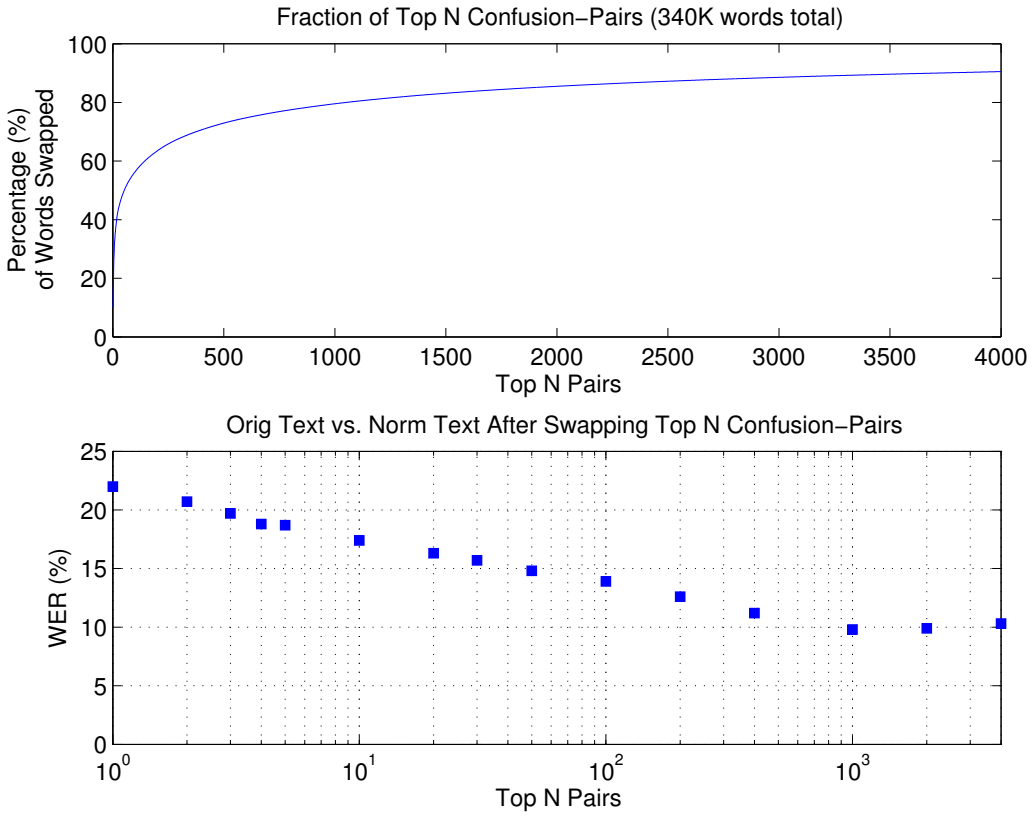

Figure 4-2: *Fraction of total confused words swapped (top), text WER when swapping top N confusion pairs (bottom). Minimum WER of 9.8 % at 1000 pairs.*

Our Solution

Now we ask if there is another way to consistently retrieve the original text from any normalization that may happen when producing diacratized and morpheme based text. For diacratized text we choose to use the following method, where we tie the original word to the diacratized word in the text. For example, *ktAb ElY >lTAwlp* would be formatted as *ktAb_kitAbN ElY_Ealay >lTAwlp_AlTAwilah*, which we use in the lexicon and language model, and is the format output by the ASR system. This is a hybrid between the original and diacratized word in the format *orig_diacratized*. Simple post-processing is done to extract the original word underlying the diacratized

word form. And so we would correctly hypothesize *ktAb ElY >lTAwlp* instead of *ktAb Ely AlTAwlh* which would not have matched the reference.

For morpheme based text we choose to retokenize the original text by comparing it to the MADA tokenized text. In other words, we are substituting the words from the original text in the tokenized text and splitting them where applicable. Thus, the hypothesis is already in the format of the original text, and all that is required is to reconnect stems and affixes. This method eliminates the issue of normalizing text.

Evaluations on the original text allows comparisons with future research work. The original corpus text is more accessible than attempting to implement normalizations techniques which aren't standardized across the literature.

4.3 Experimental Setup

The following experiments investigate the impact language modeling can have on Arabic ASR performance. We compare nondiacratized and diacratized text, as well as their morpheme equivalent.

Acoustic Models

The acoustic models are built for each text using their corresponding language and lexical models. We use the baseline ASR setup described in Section 2.4, with HMM-GMM triphone models based on MFCCs, and a trigram language model.

Lexical Models

For each text format a corresponding lexicon is built. The entries are graphemic using the underlying MADA diacratization for each word in the vocabulary. All consonants are modeled, as well as diacritics, including short vowels (*a, u ,i*), geminates (*f~, b~, l~, . . .*), and nunnations (*F, K, N*). The logic in using this lexicon is explained in the next section under 'Baseline'.

Language Models

We prepare the texts to be used for language modeling in a specific manner to work around the issue of normalizing text. Table 4.3 (an extension of Table 4.1) summarizes examples of the final script formats used in the experiments.

Table 4.3: *Examples of script formats used in experiments.*

No Diacritics	*wktbt fy ktAbk*
Diacritics	*wktbt_wakatabotu fy_fiy ktAbk_kitAbuka*
No Diacritics Morphemes	*w+ ktbt fy ktAb +k*
Diacritics Morphemes	*w+_wa+ ktbt_katabotu fy_fiy ktAb_kitAbu +k_+ka*

Nondiacratized text

This is the original text from the corpus in Romanized form.

Diacratized text

The words in this text are in a hybrid format of *orig_diacratized* between the original nondiacratized text and the diacratized produced by MADA. We take the top scoring diacratization as the true identity of the nondiacratized word.

The question that now comes to mind would be how this format affects language modeling, followed by the concern that it would make the data more sparse. This may be true; however, since the mappings between diacratized and nondiacratized words is not deterministic, we expect that more accurate pattern and word relationships are maintained and captured in the language model, compensating for potentially incorrect word normalizations. For example, if *ktb ElY >lTAwlp* means *book on the table*, rather than some other esoteric meaning, its diacratized form would be *kitAb EalaY AlTAwilah*. However, it may be that in another part of the text, MADA decided to normalize it as *kitAb Ealiy AlTAwilah* (*book Ali the table*). If we were to model these diacratizations without their underlying nondiacratized word, the word *Ealiy* would be modeled along with other appearances of *Ealiy*, which is underlyingly *Ely* and not the normalized *ElY*.

Modeling *orig_diacratized* offers a slightly more constrained language model increasing the vocabulary by 2.0% from 80,270 to 82,050 words, and an OOV of 6.99% rather than 6.68% if the diacratized words only were used.

Nondiacratized Tokenized Text

This text has words split from their stems and affixes where possible. To avoid issues with normalization, the MADA words, stems, and affixes are substituted with their equivalents from the original text. 38% of vocabulary words were split into morphemes, and 18% of words in the text were split. This reduces the vocabulary by 33% from 61,000 words to 41,000, with an OOV of 2.75% rather than 5.37% at the word level. Here are the list of affixes:

Prefix: { b+, f+, fy+, k+, l+, A+, hA+, lA+, mA+, yA+ s+, w+}

Suffix: { +h, +hA, +hm, +hmA, +hn, +k, +km, +kmA, +kn, +mA, +mn, +m, +nA, +ny, +y}

Diacratized Tokenized Text

The nondiacratized text is diacratized by building the *orig_diacratized* hybrid at the morpheme level. Here are the list of affixes on the diacratized side only:

Prefixes: { >a+, bi+, buw+, fA+, fa+, fiy+, hA+, kA+, ka+, kul+, l+, lA+, la+, li+, mA+, sa+, wA+, wa+, yA+}

Suffixes: { +h, +hA, +him, +himA, +hin~a, +hum, +humA, +hun~a, +ka, +kam, +kamA, +ki, +kum, +kumA, +kun~a, +lA, +l~A, +m, +mA, +ma, +man, +min, +m~A, +m~a, +m~an, +nA, +niy, +niy~i, +n~A, +n~iy, +y, +ya, +y~a}

The same questions regarding language modeling arises as when using the diacratized word level text. Using the hybrid format, the vocabulary increases by 3.4% from 62,400 words to 64,500, and the OOV increases from 4.08% to 11.02%. Take into consideration that this high increase in OOV may be counterintuitive to what would be expected to happen to WER rates, since hypotheses output by the ASR are post-processed, thus compensating for what one would anticipate to be a hit in performance.

Once the decoder has output the hypotheses, which would be the format the language model has been built, additional post-processing is performed, where applicable, to build the text in the format of the original nondiacratized text. Thus all final evaluations are performed on the same format, i.e., the nondiacratized text. Figure 4-3 provides an example of this post-processing step.

Table 4.4 summarizes the vocabulary and grapheme size of the lexicons produced from using different texts. Nondiacratized words may have more than one potential diacratized form. The table shows that the number of Pronunciations Per Word (PPW) is around 1.28. Diacratized text produces a lexicon with a constrained mapping, thus in all diacratized texts the PPW is 1. Nondiacratized morphemes have the potential to map to multiple diacratizations with the highest PPW of 1.51.

Table 4.4: *Lexicon Sizes When Using Different Texts.*

Lexicon	Vocab	Grapheme	PPW
No Diacritics	61K	79K	1.28
Diacritics	82K	82K	1
No Diacritics Morphemes	41K	62K	1.51
Diacritics Morphemes	65K	65K	1

Baseline

We use the diacratized graphemic lexicon as the baseline, with a WER of 23.4% and OOV of 5.31% on the Evaluation dataset. This is different from the baseline established in the previous chapter which uses a nondiacratized lexicon (25.1% WER). We choose to use a diacratized lexicon since we are less concerned about the impact of the lexicon, and more concerned with the impact of language modeling. Half of

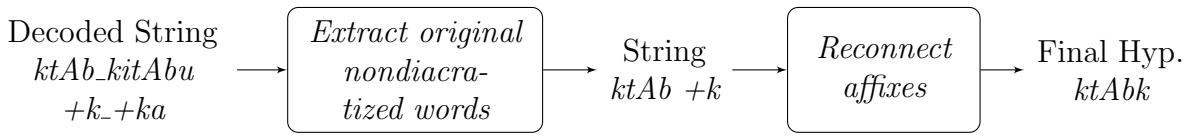

Figure 4-3: *Example of Post-Processing Step after Decoding to Produce Final Hypothesis*

our experiments are are built on diacratized text, so for fairness, we include diacritics in all lexicons. We chose not to include pronunciation rules because they might bias results in favor of word-based language modeling over morphemes. This is because the rules were developed with Arabic words in mind, rather than morphemes.

The baseline in the last chapter was compared to ASR performance presented in the literature, to present how it seems reasonable to use, as it falls within the range of WER values reported under somewhat similar parameters. The old baseline was in harmony with values reported by *Xiang et al.*, and *Vergyri et al.*, who were using broadcast news data, a similar sized 64K vocabulary, OOV rate of 4.18% and 5.26%, and a diacratized lexicon. Their WERs were 24.6% and 33.6%. With our new baseline using a diacratized lexicon, we have a more relevant WER of 23.4%, that allows for better comparisons. Like before, there are other potential sources for differences, such as, acoustic modeling techniques, language modeling differences, and training and test sets that are non-matching.

Overall, our new baseline seems reasonable as it builds on top of the old one, and because it is still within range of what is found in the literature.

4.4 Results

We take the language model built using the nondiacratized text as the baseline system. As a reminder, a diacratized graphemic lexicon is used for all experiments, with the results in Table 4.5 only displaying the effects of using different language models: without diacritics, with diacritics, morphemes without diacritics, and morphemes with diacritics.

We observe that training and decoding over the diacratized text helps ASR performance, with a modest 0.3% absolute improvement on the Evaluation set, from 23.4% to 23.1% WER. This improvement is in light of the vocabulary size being 34% larger and the OOV rate being 33% higher.

Using a morpheme-based (tokenized) language model outperforms that of the word-based. The nondiacratized tokenized text outperforms the diacratized text with

Table 4.5: *Impact of Language Modeling on ASR Performance.*

Language Model	Vocab	OOV (%)	Search Graph (MB)	Dev WER (%)	Eval WER (%)	Significance at $p <$
Baseline - No Diacritics	61 K	5.27	253	22.6	23.4	-
Diacritics	82 K	6.99	249	21.8	23.1	0.187
No Diacritics Morph.	41 K	2.75	229	21.0	23.2	0.465
Diacritics Morph.	65 K	11.02	193	19.9	22.4	0.001

another modest 0.2% absolute WER improvement (23.4% to 23.2% WER). It is interesting to note that constraining the word level language model with the inclusion of diacritics is slightly more advantageous than modeling with morphemes. So given equivalent text, it would be more beneficial to diacratize it over performing morphological decomposition.

Finally, we see that modeling the tokenized text with diacritics provides the largest gain with an absolute improvement of 1.0% from the baseline of 23.4% to 22.4% WER. This occurs with a higher vocabulary than the baseline, and over double the OOV rate. As anticipated earlier, these two metrics of vocabulary size and OOV may be deceiving when evaluating the actual gains in performance of constraining the language to such a degree.

The characteristic of the lexicons may be one reason that diacratized text outperforms nondiacratized text, regardless of whether it is tokenized or not. When text is diacratized the lexicon has only one pronunciation mapped to each entry, whereas the nondiacratized text has 1.28 PPW and 1.51 PPW for the equivalent nondiacratized tokenized text. Therefore during training there is more than one possible pronunciation available, which could potentially be producing less accurate acoustic models than if constrained over a single pronunciation, as in the case of the diacratized language models.

Overall it seems that including information on both diacritics and morphemes maximizes the gains to be made in ASR performance. Supplying this information separately provides a maximum absolute WER improvement of 0.3% over baseline (23.4% to 23.1% WER), while including both provides a 1.0% absolute improvement

over the baseline (23.4% to 22.4% WER). This final result shows to be statistically significant with $p < 0.001$, using MAPSSWE.

In addition to improving ASR performance, a more constrained language model allows for the use of smaller and less memory consuming decoding graphs. The size of these graphs can be reduced by 24% (253MB to 193MB) for the best performing system over the baseline.

4.5 Summary

In this chapter we explore the issue of Arabic text normalizations, and the method we pursued. To allow for an accessible standard when evaluating ASR performance with a reference transcript, we chose to use the original text provided with the corpus. In order to use the normalized outputs of the MADA+TOKAN Toolkit we chose to work with the diacratized texts, using a hybrid word format between the original nondiacratized text and the diacratized text in the form *orig_diacratized*. This allows for easy post-processing with a small difference between this format and the diacratized only format.

We also explored the use of several texts for language modeling; nondiacratized, diacratized, and the equivalents tokenized. We found that given more information on the diacratization and morphology of the text, WERs improved. What was most promising was that combining both information into the language model allowed for even stronger ASR performance. This information also contributed to building smaller decoding graphs which are necessary for applications with limited memory, and time constraints.

Chapter 5

Conclusion

5.1 Summary of Work

A running theme in this thesis was the influence of diacritics in lexical and language modeling. Most texts available are not diacratized, but when they are, diacritics compose around 30% of this data, as shown with our GALE corpus. We sought to investigate the impact of modeling the different classes of diacritics in the lexicon (short vowels, geminates, nunnations), followed by the inclusion of pronunciation rules given these diacritics. We then investigated different text formats for language modeling covering nondiacratized, diacratized, word, and morphemic texts. We also presented the challenges faced when building our text corpora, and a technique for working with various normalized data.

No class of diacritics can be neglected in lexical modeling. A graphemic lexicon that includes all diacritics provides an absolute improvement of 1.7% in WER ($p < 0.001$) when compared to a nondiacratized graphemic lexicon. However, ignoring geminates yields a 1.9% in absolute WER improvement ($p < 0.001$). This boost in performance may be attributed to the sparsity of geminates.

Beyond a graphemic lexicon, adding another dimension to lexical modeling, is the inclusion of pronunciation rules. Simple rules that model key behavior in word pronunciation showed to be the most effective with a 2.4% absolute improvement in WER ($p < 0.001$) when compared to a nondiacratized graphemic lexicon. It also

showed to be consistent with results on the graphemic lexicon, where geminates were not modeled, and no loss in performance was observed.

We faced the challenge of text normalization while automatically diacratizing text. To run all evaluations on the original corpus transcripts, we compensated for these normalizations by building our text using a hybridized format. This format contains both the original word and MADA diacratization as *orig_diacratized*. This allowed us to perform a simple post-processing step to retrieve the original text for evaluation.

Finally, because Arabic is a morphologically rich language, including diacritics or morphemes in the language model improved ASR performance. When including both information simultaneously, we observed the largest gains, with an absolute improvement of 1.0% in WER over the baseline system of using word level, nondiacratized text ($p < 0.001$).

5.2 Future Explorations

The work presented in this thesis provides momentum towards further explorations in lexical modeling. One path to pursue would be to apply the best performing pronunciation rules to morphemes. Other paths to explore would be to investigate use of stochastic lexicons, the generation of lexicons from other Arabic resources, and expanding these techniques to dialects.

5.2.1 Pronunciation Rules in Morpheme-based Lexicons

A natural extension would be to explore the impact of pronunciation modeling of a lexicon composed of morphemes. This combines the work developed in Chapters 3 and 4. In this thesis our experiments on language modeling used a graphemic lexicon to avoid potential biases introduced by word-based pronunciation rules. For example, the word *biAlkitAbi* would be modeled according to Rules III as:

/w ae b ih ae: l k ih t ae: b (ih | ϵ)/

A morphemic lexicon would map entries *bi* and *AlkitAbi* in *bi+AlkitAbi* to:

/b (ih | ϵ) # ae: l k ih t ae b (ih | ϵ)/

It is hard to predict whether in the case of *bi*, rules that would accommodate coarticulation between /ih/ and /ae:/ would help, or whether /b/ is too small an acoustic unit for a single lexicon entry. To verify this impact we would need to run further experiments.

5.2.2 Stochastic Lexicons

We have also established ASR gains to be made while keeping an MSA lexicon fairly constrained to less than 2 PPW. The lexicon had this PPW because we used only the top MADA diacratization hypothesis. For words with multiple occurrences, there could be more than one hypothesis returned by MADA. Additionally, some pronunciation rules described in this thesis produced multiple pronunciations for certain words. It would be interesting to loosen this parameter and assume less knowledge of the potential diacratizations of a word.

It might be fruitful to explore how using all diacratizations would impact ASR performance and the use of stochastic lexicons using for example, Pronunciation Mixture Modeling (PMM) as presented by *McGraw et al.* [40]. This would allow us to accommodate variations in pronunciation since we allow the data to inform us about word pronunciations, rather than using a normative approach to pronunciation modeling, through the use of automatic diacratization toolkits. It might also compensate for mistaken diacratizations hypothesized by these toolkits. Additionally, stochastic lexicons would allow for a more realistic distribution of pronunciation probabilities.

Furthermore, we can extract the most probable pronunciation for a given word and rebuild the text in its diacratized form which could provide additional gains in ASR performance. It would be worth investigating if, a single or multiple iterations of updating the PMM and language model provides any gains, and the impact of seeding lexicon weights using scores provided by the MADA hypotheses of diacratized words.

5.2.3 Lexical Modeling using other Arabic Resources

Building on top of stochastic lexicons is the question of generating diacratizations and pronunciations for words which may not exist in a database because they may be rare, from a dialect, or of foreign origin. This would accommodate the nature of languages and speech, which evolve by discarding some words and introducing new words. This allows us to look beyond MSA scripted speech, at conversations which include colloquial words, names of people and places, and topics that are trending.

We could use joint-sequence models (Letter to Sound (L2S), Grapheme to Phoneme (G2P)) as presented by *Bisani et al.* [41] to build models from other texts. One source to train models could be either manually or automatically diacratized text [17], or potentially, manually constructed dictionaries. One idea to consider would be whether these joint-sequence models can accommodate contextual information by building models that incorporate word history. Diacritics, are after all, context based.

5.2.4 Beyond MSA

Many of the experiments and conclusions in this thesis should hold when applied to dialects. Additional information on diacritics incorporated into the lexicon will probably help ASR performance. This is because we are modeling short vowels, among other phenomenon such as nunnation, which compose a large and significant part of speech.

Pronunciation rules would help if they model behavior that is consistent across dialects, otherwise, dialect specific rules may need to be developed to observe gains beyond those introduced by modeling diacritics.

Diacratized and tokenized language models might help as well. The challenge here would be diacratizing colloquial words and texts, since dialects tend not to be codified. This is where the ideas above, on using joint-sequence models and PMMs to predict diacratizations and phonetic composition of words, might bear fruitful results.

Bibliography

[1] M. E. Ahmad, "Toward an Arabic Text-to-Speech System," *The Arabian Journal for Science and Engineering*, vol. 16, pp. 565–583, October 1991.

[2] Y. A. El-Imam, "Phonetization of Arabic: Rules and algorithms," *Computer Speech & Language*, vol. 18, no. 4, pp. 339–373, 2004.

[3] Y. Alotaibi and A. Husain, "Formant Based Analysis of Spoken Arabic Vowels," in *Biometric ID Management and Multimodal Communication* (J. Fierrez, J. Ortega-Garcia, A. Esposito, A. Drygajlo, and M. Faundez-Zanuy, eds.), vol. 5707 of *Lecture Notes in Computer Science*, pp. 162–169, Springer Berlin Heidelberg, 2009.

[4] M. A. Ahmed, "Toward an Arabic Text-to-Speech System," *The Arabic Journal for Science and Engineering*, vol. 16, no. 4B, pp. 565–583, 1991.

[5] Y. A. El-Imam, "Phonetization of Arabic: Rules and algorithms," *Computer Speech & Language*, vol. 18, no. 4, pp. 339–373, 2004.

[6] F. Biadsy, N. Habash, and J. Hirschberg, "Improving the Arabic pronunciation dictionary for phone and word recognition with linguistically-based pronunciation rules," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, (Stroudsburg, PA, USA), pp. 397–405, Association for Computational Linguistics, 2009.

[7] L. M. Paul, G. F. Simons, and C. D. Fennig, "Ethnologue: Languages of the World, Seventeenth edition," 2013.

[8] M. Van Mol, *Variation in Modern Standard Arabic in Radio News Broadcasts: A Synchronic Descriptive Investigation Into the Use of Complementary Particles*. Orientalia Lovaniensia Analecta, 117, Isd, 2003.

[9] D. Newman, "The phonetic status of Arabic within the world's languages : the uniqueness of the lughat al-daad.," *Antwerp papers in linguistics.*, vol. 100, pp. 65–75, January 2002.

[10] S. Abu-Rabia, "Reading in Arabic orthography: The effect of vowels and context on reading accuracy of poor and skilled native Arabic readers," *Reading and Writing*, vol. 9, pp. 65–78, 1997.

[11] K. Ryding, *A Reference Grammar of Modern Standard Arabic*. Reference Grammars, Cambridge University Press, 2005.

[12] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1st ed., 2000.

[13] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Computer Speech & Language*, vol. 16, no. 1, pp. 69–88, 2002.

[14] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[15] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, pp. 357–366, Aug 1980.

[16] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL-1996)*, pp. 310–318, 1996.

[17] M. Yaseen, M. Attia, B. Maegaard, K. Choukri, N. Paulsson, S. Haamid, S. Krauwer, C. Bendahman, H. Fersøe, M. Rashwan, *et al.*, "Building annotated written and spoken Arabic LRÕs in NEMLAR project," in *Proceedings of LREC*, 2006.

[18] G. Z. Alexandra Canavan and D. Graff, "CALLHOME Egyptian Arabic Speech," 1997.

[19] A. P. Ltd, "Iraqi Arabic Conversational Telephone Speech," 2006.

[20] "LDC2013S02, GALE Phase 2 Arabic Broadcast Conversation Speech Part 1 Linguistic Data Consortium," 2013.

[21] W. at al., "LDC2013S07, GALE Phase 2 Arabic Broadcast Conversation Speech Part 2 Linguistic Data Consortium," 2013.

[22] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, no. Idiap-RR-04-2012, (Rue Marconi 19, Martigny), IEEE Signal Processing Society, Dec. 2011. IEEE Catalog No.: CFP11SRW-USB.

[23] A. Stolcke, "SRILM - An Extensible Language Modeling Toolkit," pp. 901–904, 2002.

[24] M. Maamouri, D. Graff, B. Bouziri, S. Krouna, and S. Kulick, "LDC Standard Arabic Morphological Analyzer (SAMA) v. 3.1," *LDC Catalog No. LDC2010L01. ISBN*, pp. 1–58563, 2010.

[25] S. Green and J. DeNero, "A class-based agreement model for generating accurately inflected translations," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pp. 146–155, Association for Computational Linguistics, 2012.

[26] O. R. Nizar Habash and R. Roth, "MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization," in *Proceedings of the Second International Conference on Arabic Language Resources and Tools* (K. Choukri and B. Maegaard, eds.), (Cairo, Egypt), The MEDAR Consortium, April 2009.

[27] M. Maamouri, A. Bies, T. Buckwalter, and W. Mekki, "The Penn Arabic Treebank: Building a large-scale annotated Arabic corpus,"

[28] N. Y. Habash, "Introduction to Arabic natural language processing," *Synthesis Lectures on Human Language Technologies*, vol. 3, no. 1, pp. 1–187, 2010.

[29] "NIST Scoring Toolkit,"

[30] J. Billa, M. Noamany, A. Srivastava, J. Makhoul, and F. Kubala, "Arabic speech and text in TIDES OnTAP," in *Proceedings of the second international conference on Human Language Technology Research*, HLT '02, (San Francisco, CA, USA), pp. 7–11, Morgan Kaufmann Publishers Inc., 2002.

[31] M. Afify, L. Nguyen, B. Xiang, S. Abdou, and J. Makhoul, "Recent progress in Arabic broadcast news transcription at BBN," in *INTERSPEECH'05*, pp. 1637–1640, 2005.

[32] A. Messaoudi, L. Lamel, and J.-L. Gauvain, "Modeling vowels for Arabic BN transcription.," in *INTERSPEECH*, pp. 1633–1636, 2005.

[33] A. Messaoudi, J.-L. Gauvain, and L. Lamel, "Arabic Broadcast News Transcription Using a One Million Word Vocalized Vocabulary," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, p. I, may 2006.

[34] T. Ng, K. Nguyen, R. Zbib, and L. Nguyen, "Improved morphological decomposition for Arabic broadcast news transcription," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 4309 –4312, april 2009.

[35] B. Xiang, K. Nguyen, L. Nguyen, R. Schwartz, and J. Makhoul, "Morphological decomposition for Arabic broadcast news transcription," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, pp. I–I, IEEE, 2006.

[36] L. Mangu, H.-K. Kuo, S. Chu, B. Kingsbury, G. Saon, H. Soltau, and F. Biadsy, "The IBM 2011 GALE Arabic speech transcription system," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, pp. 272–277, 2011.

[37] D. Vergyri, A. Mandal, W. Wang, A. Stolcke, J. Zheng, M. Graciarena, D. Rybach, C. Gollan, R. Schlüter, K. Kirchhoff, A. Faria, and N. Morgan, "Development of the SRI/nightingale Arabic ASR system," in *INTERSPEECH*, pp. 1437–1440, 2008.

[38] A. E.-D. Mousa, R. Schlüter, and H. Ney, "Investigations on the use of morpheme level features in Language Models for Arabic LVCSR," in *ICASSP*, pp. 5021–5024, IEEE, 2012.

[39] S. Kawahara, "Sonorancy and geminacy," *University of Massachusetts occasional papers in linguistics*, vol. 32, pp. 145–186, 2007.

[40] I. McGraw, I. Badr, and J. Glass, "Learning Lexicons From Speech Using a Pronunciation Mixture Model," 2013.

[41] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.