Lexical Modeling for Arabic ASR: A Systematic Approach

Tuka Al Hanai, James Glass

MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA 02139, USA

{tuka,glass}@mit.edu

Abstract

Arabic has an ambiguous mapping between words and pronunciations, making it a deep orthographic system. This ambiguity can be resolved through diacritics, which if displayed, would compose 30% of characters in a text. We investigate the different dimensions of lexical modeling, covering diacritics, pronunciation rules, and acoustic based pronunciation modeling.

We show the impact of explicitly modeling the different classes of diacritics (short vowels, geminates, nunnations). We further show that a phonetic lexicon, derived by applying simple pronunciation rules to diacritized words, offers the best gains in ASR performance. Finally, deriving pronunciations from acoustics, yields improvements, beyond a canonical lexicon.

Index Terms: automatic speech recognition, Arabic, diacritics, pronunciation rules, language model, lexical model, joint sequence model, pronunciation mixture model.

1. Introduction

Arabic orthography presents a challenge when working in the area of speech and language processing. It has a deep orthographic system, with an ambiguous mapping between words and their phonetic realization. Although the Arabic script may include diacritics to inform the reader on the underlying pronunciation of a word, it is rarely used. Thus, in Arabic ASR, lexical modeling becomes a non-trivial pursuit.

The classical way of deriving Arabic word pronunciations is by parsing the text with NLP tools to diacritize words. Diacritized words generally provide a consistent mapping between graphemes and phonemes. Rather than using this normative technique of determining word pronunciations, we would like the acoustics to inform us about the observed pronunciations of a word. A data driven approach to lexical modeling allows for words to be modeled based on observed pronunciations, and accommodates words that may not exist in an NLP database due their rarity, colloquial origins, or foreign nature.

The rest of this paper proceeds with a literature review (Sec. 2) highlighting the area open for exploration, followed by a background on the nature of the Arabic language, and theory of the applied techniques (Sec. 3). Next, we establish a framework (Sec. 4) to evaluate the effects of diacritics (Sec. 5.1), and pronunciation rules (Sec. 5.2), in Arabic ASR performance. We then explore the utility of a stochastic lexicon built using a generative framework of candidate pronunciations (Sec 5.3). We conclude with future directions of this work (Sec. 6).

2. Related Work

The existing literature indicates that there is not a standard approach towards lexical modeling for Arabic. Automatically diacritized (D) graphemic lexicons are commonly used [1, 2, 3, 4]. Less common is work like Billa et al. that uses a nondiacritized (ND) lexicon [5]. While, Afify et al. experiment with graphemic lexicons that are both D and ND [1].

A few researchers choose to apply pronunciation rules, such as Messaoudi et al., Vergyri et al., and Mangu et al. [6, 7, 8]. The first two use their own rules, with Mangu et al. applying the rules of Biadsy et al. Biadsy et al. investigate pronunciation rules in the area of Arabic ASR [9], which we observed to be an uncommon research endevour. Some work exists on phonetic modeling for applications in Modern Standard Arabic (MSA) Text-To-Speech (TTS) systems, namely Ahmed and El-Imam [10, 11]. We find that investigations on the impact of diacritics and pronunciation rules are rarely conducted and compared. This provides good motivation for investigating the different lexical modeling techniques under a single setup.

There exists some work in the domain of Arabic ASR that uses stochastic lexicons. Mangu et al. build a stochastic lexicon based on confidence scores for each D the MADA+TOKAN toolkit hypothesizes, and the pronunciation returned by the decoder [12, 13]. Vergyri et al. compare the use of non-stochastic and stochastic lexicons. Pronunciation probabilities were smoothed empirical frequencies of pronunciations returned by the decoder. WER improved when using a stochastic lexicon [7]. Al-Haj et al. also investigate the use of stochastic lexicons for dialectical Arabic, deriving the pronunciation probabilities in the same manner as Vergyri et al. [14]. In all cases, pronunciations were derived from an existing database.

Given previous work, we seek to establish a more systematic framework for evaluating the impact of modeling diacritics, applying pronunciation rules, and deriving pronunciations acoustically, beyond the canonical. All under a single setup.

3. Background

3.1. Arabic Language

To discern the underlying identity (pronunciation) and meaning of a word, diacritics are inserted in Arabic orthography. They are most commonly employed in texts for those learning to read, such as children, and older texts whose style may be unfamiliar, or sensitive to ambiguity. Diacritics are expressed in their Romanized version (using Buckwalter transliteration) as in Table 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).

3.2. Joint Sequence Modeling

The task of converting graphemes to phonemes is formalized as

$$\mathbf{b}^* = \underset{b \in B}{\operatorname{argmax}} P(\mathbf{w}, \mathbf{b}) \quad (1)$$

where the most likely pronunciation $\mathbf{b}$, in the set of phonetic units B, is sought for the orthographic form $\mathbf{w}$ in the set of graphemes W.

Table 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.		

Joint sequence models (JSM) represent the relationship between graphemes and phonetic units by constructing an M-gram over joint units [15]. In a joint unit, any grapheme or ϵ (empty) character may map to any phoneme or ϵ character, with ϵ to ϵ being redundant. Below is a sequence of 6 joint units of grapheme-phoneme pairs for the word *Alshms* (the sun).

$Alshms$	=	A	l	sh	ϵ	m	s
$/A\ sh\ ae\ m\ s/$		A	ϵ	sh	ae	m	s

We use Bisani and Ney's implementation to model joint sequences over 5-grams, a higher M-gram would produce too constrained a model [15]. We map between grapheme L and phoneme R with a 0 or 1 input-output mapping ($L = R = 1$). We generate K = 50 candidate pronunciations for each word.

3.3. Pronunciation Mixture Modeling

Using the Pronunciation Mixture Modeling (PMM) framework developed by McGraw et al. [16], we are interested in learning weights of different word pronunciations for any given word. The PMM is parameterized with $\theta = P(\mathbf{b}, \mathbf{w})$, assuming that there is a mapping between word w and baseform b. EM is used to update these parameters using the data (u_1^N, w), with utterance u. The log likelihood of the data is formalized as

$$L(\theta) = \sum_{i=1}^{N} \log p(u_i, \mathbf{w}; \theta) = \sum_{i=1}^{N} \log \sum_{\mathbf{b} \in B} \theta_{w,b} p(u_i | \mathbf{w}, \mathbf{b}) \tag{2}$$

In this paper, pronunciations are scored over an N-best list of 100 hypotheses, and are discarded if less than T = 0.01.

4. ASR System

The Kaldi Speech Recognition Toolkit was used to build the ASR system with triphone GMM acoustic models [17]. A trigram language model was built using modified Kneser-Ney discounting with the SRILM Toolkit.[1] The transcripts were automatically diacratized using the MADA+TOKAN Toolkit [12].

We conduct experiments using the GALE Broadcast Conversation Phase 1 and 2 dataset.[2] Using only the data labeled 'report' which is mostly scripted speech in MSA. The training set is 70 hours in duration, with the Development (Dev) and Evaluation (Eval) set having 1.4 and 1.7 hours in duration. The Dev set contains utterances from the show (ALAM_WITHEVENT) while the Eval set is from the show (ARABIYA_FROMIRAQ). The transcripts provided are not diacritized. For JSM we use manually diacritized text from two other sources, Nemlar broadcast news [18], and the Quran[3].

Triphone context-dependent GMM-HMM models were used with MFCC+LDA+MLLT features, and applying fMLLR [19, 20, 21]. These models were seeded from monophone models that were built using a flat start initialization according to the standard recipe from the Kaldi Toolkit. We use 4,000 HMM states, and 128,000 GMMs.

In order to build a lexicon that captures the underlying pronunciation of a ND word, we diacritize the GALE and Nemlar transcripts using the MADA+TOKAN toolkit [12]. Specifically, MADA+TOKAN Toolkit 3.2 with SAMA 3.1 on default.

Only text from the training set has been used to build the language model. The text contains 500K words with a vocabulary of 61K words. Unless otherwise stated, language models used are trigrams with modified Knesser-Ney discounting built with the SRILM toolkit. The Out-Of-Vocabulary (OOV) rate is 5.27% on the combined Dev (5.23%) and Eval (5.31%) sets.

Although we investigate lexicons built from several sources, all map ND words to phonetic representations.

5. Experiments

5.1. Diacritics

We first explore the influence of diacritics on ASR performance. The lexicons we investigate are grapheme-based, and map the ND words from the training text to their D form. We compare with a baseline ND lexicon that models the word as is, as well as four configurations of the three classes of diacritics. The D form is based on the top MADA+TOKAN hypothesis. We observed that diacritics compose ~30% of characters in the dataset.

Table 2: *Example Entries in Lexicon.*

Lexicon	Vocab	Grapheme
No Diacritics (ND)	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 (D)	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

Table 2 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 diacritized 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/, obscuring the underlying pronunciation.

ND. This lexicon maps every word in the vocabulary of the ND training text to its grapheme form.

[1] A. Stolcke, SRILM - An Extensible Language Modeling Toolkit

[2] LDC2013S02,LDC2013S07, GALE Phase 2 Arabic Broadcast Conversation Speech Part 1 and 2, Linguistic Data Consortium.

[3] http://tanzil.net/download, Tanzil-Quran.

Table 3: *Impact of Modeling Diacritics in Lexicon on ASR Performance.*

Lexicon	PPW	# phones	Freq. in text (%)	Dev WER(%)	Eval WER(%)	Sig. at $p <$
Baseline - ND	1	36	-	24.2	25.1	-
D - Short Vowels only	1.25	39	25	23.4	24.1	0.007
D - No geminates	1.28	42	3	22.6	23.2	0.001
D - No nunnations	1.25	69	1	22.8	23.9	0.001
All D	1.28	72	29	22.6	23.4	0.001

D - 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∼, . . .* }.

D - No Geminates. Models short vowels and nunnations {*a, u, i, F, K, N*}, but not geminates {*b∼, f∼, l∼, . . .* }.

D - No Nunnations. This lexicon models short vowels and geminates {*a, u, i, b∼, f∼, l∼, . . .* }, but not nunnations {*F, K, N*}. Note that 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/.

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

5.1.1. Baseline

We use the ND lexicon as the baseline, with a WER of 25.1% and OOV of 5.31% on the Eval 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. [3, 22]. This is considering the smaller size (70 hours) of our training corpus compared to 150 and 1000 hours, using an ND lexicon, similar nature (Arabic Broadcast News) of the dataset, similar vocabulary size (61K), similar OOV, with some differences in acoustic modeling. Xiang et al. does not detail the models they used, however Vergyri et al. describe several techniques, using MFCC, PLP, and MLP features, as well as fMPE training.

5.1.2. Results

Table 3 displays the results of training and decoding using these lexicons that vary only in diacritics. Inserting acoustic units in the lexicon to model diacritics outperforms the baseline by 1.7% absolute WER. This shows that modeling diacritics as part of consonants 'works', but is not as effective as diacritized lexicon entries. Even partially including diacritics helps.

Short vowels and nunnations provide an almost equivalent gain in ASR performance. Short vowels result in a 1.0% absolute WER improvement over the baseline, modeling nunnations leads to a 1.9% absolute WER improvement over the baseline. Geminates produce an absolute WER improvement of 1.2% when modeled with short vowels. Geminates help performance when nunnations are missing, but offer no gain when nunnations are modeled. There is actually a loss when geminates are modeled with other diacritics, leading to a 1.7% absolute WER improvement rather than a 1.9% when not modeling geminates.

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.[4]

[4]N.S.R.S.Toolkit, Speech Recognition Scoring Toolkit, 2001.

5.2. Pronunciation Rules

Intuitively, there are many graphemes that may correspond to multiple phonemes, with 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 pronunciation rules from the literature by Ahmed, El-Imam, and Biadsy et al. under a single setup [10, 11, 9].

The lexicon maps the ND vocabulary from the training text to their D 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 [10] 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 phones, and retroflexed vowels. Cross-word rules were not implemented.

Rules II by El-Imam [11] 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, while ignoring geminates.

Rules III by Biadsy et al. [9] 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.

5.2.1. 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 4. We take the baseline, as before, to be the ND graphemic lexicon assessed over the Eval data. All lexicons perform better than the baseline, with two out of the three performing better than the D graphemic lexicon.

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 D graphemic lexicon. This may be due to data sparsity when modeling the acoustics of these phones.

The other two rule-based lexicons fair better. Rules II slightly outperforms the D 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.

Overall, it would seem that it hurts to model phones too finely with the data size 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 $p < 0.004$.

Table 4: *Impact of Lexicon Pronunciation Rules on ASR Performance.*

Lexicon	PPW	# phones	Dev WER(%)	Eval WER(%)	Sig. at $p <$
ND	1	36	24.2	25.1	-
D Grapheme	1.28	72	22.6	23.4	0.001
Rules I - *Ahmed* [10]	1.27	135	22.9	24.0	0.004
Rules II - *El-Imam* [11]	1.27	63	22.4	23.3	0.001
Rules III - *Biadsy et al.* [9]	1.85	34	22.3	22.7	0.001

Table 5: *Impact of Stochastic Lexicon on ASR Performance.*

Phonetic Lexicon	JSM Vocab (PPW)	PMM Lex. PPW	Dev WER (%)	Eval WER (%)	Sig. at $p <$
Baseline - Rules III	-	1.85	22.3	22.7	-
GALE + MADA D - all prons	-	1.57	20.9	22.4	0.254
GALE + MADA D - top prons	61K (1.28)	2.05	20.5	22.3	0.407
GALE + MADA D - all prons	61K (5.42)	2.02	20.5	22.0	0.026
Nemlar + manual D	39K (1.56)	2.05	20.9	22.5	0.159
Nemlar + MADA D - top prons	36K (1.29)	2.02	20.2	22.2	0.689
Quran + manual D	15K (1.18)	1.93	22.6	23.9	0.001

Candidate pronunciations generated using JSM. $M = 5$, $L = R = 1$, $K = 50$, $N = 100$, $T = 0.01$.

5.3. Acoustic Based Pronunciation Modeling

We are interested in evaluating the performance of a lexicon that allows for a less constrained phonetic representation of words.

We start by using the candidate Ds available from MADA+TOKAN Toolkit, to build a list of candidate pronunciations. We then learn a stochastic lexicon where a mixture of weights are assigned to these candidate pronunciations.

From a 61K vocabulary, the word database of the MADA+TOKAN Toolkit contains 330K potential Ds (pronunciations). Any given word in the seed lexicon will have a Pronunciation Per Word (PPW) of 5.42, with any given word in the text having a PPW of 6.73.

We take this a step further, and reduce our reliance on candidate pronunciations available from NLP tools and databases. We generate candidate pronunciation from JSMs that were trained on some Arabic lexicon, which include the GALE, Nemlar, and Quran corpora. Weights were then assigned to these candidates that most closely represent the observed pronunciations in the acoustics of the speech data. This was done using the PMM framework [16].

The acoustic models used for this set of experiments were initially built using the best performing setup from the previous section. This is the lexicon that incorporates pronunciation rules from Biadsy et al., that resulted in a WER of 22.7% on the Eval set. This was also taken as the baseline.

5.3.1. Results

All lexicons performed better than the baseline except one. The PMM framework helped improve performance by producing a stochastic lexicon, while JSM, provided an additional benefit. Results are displayed in Table 5.

Evaluating using a stochastic lexicon that was based on the D database of the MADA+TOKAN toolkit led to an absolute WER improvement of 0.3% over the baseline. An extension of this, generating candidate pronunciations from JSMs, provided an additional absolute WER improvement of 0.7% over the baseline. This seems to indicate that allowing for pronunciations beyond the canonical, better accommodates the pronunciations observed in the acoustics.

Using other sources (except one) to train the JSM, also provided an absolute WER improvement over the baseline ranging between 0.2% and 0.5%. We also observe that there may be a limit as to the type of sources that can be used for lexical modeling. Using the Quran negatively affected performance, which may be due to two reasons, the smaller size of the training lexicon, and the differing nature of the Arabic that exist between the source lexicon, and testing corpora.

Overall, we observe benefits to modeling a lexicon stochastically under the PMM framework, and allowing for broader pronunciation candidates. This indicates the potential for skipping the time consuming step of pre-processing Arabic text using NLP tools to provide diacritics.

6. Conclusions & Future Work

We have presented a systematic framework to evaluate the effects of modeling the different classes of diacritics in the lexicon. We also assessed the impact of different applying pronunciation rules that are available in the literature. We found that modeling short vowels and nunnations helped performance, ignoring geminates. We also observed that simple rules helped boost performance, rather than attempting to model more nuanced phenomena. Building on these results, we proceeded to show the benefits of using a stochastic lexicon that models a mixture of pronunciations based on the acoustics. The best performance came when potential pronunciations were generated by JSMs, and then scored based on the acoustics through the PMM framework, loosening the constraint of using a canonical set of pronunciations for lexical modeling.

This paper establishes a trajectory to further explore deriving pronunciations from speech, specifically, to derive diacritizations of words from the acoustics. This could potentially allow us to discern the semantics in speech, skip the preprocessing of Arabic text with NLP toolkits, and perform more accurate acoustic modeling. This work can also be used to explore lexical modeling of Arabic dialects, where a lexical gold-standard is lacking, and NLP is not an option.

We are currently applying acoustic-based data-driven pronunciation modeling techniques on Arabic data other than MSA. An area which would most benefit, since no NLP information, such as MADA+TOKAN, exists, and where human annotation is not feasible.

7. Acknowledgements

Many thanks to the Al Nokhba Scholarship Program and Abu Dhabi Education Council for their continuous support.

8. References

[1] 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.

[2] 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.

[3] 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.

[4] 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, 2012.

[5] 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.

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

[7] D. Vergyri, A. Mandal, W. Wang, A. Stolcke, J. Zheng, M. Graciarena, D. Rybach, C. Gollan, R. Schlüter, K. Kirchhoff, *et al.*, "Development of the SRI/nightingale Arabic ASR system.," 2008.

[8] 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.

[9] 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.

[10] 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.

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

[12] 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.

[13] 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, IEEE, 2011.

[14] H. Al-Haj, R. Hsiao, I. Lane, A. W. Black, and A. Waibel, "Pronunciation modeling for dialectal Arabic speech recognition," in *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pp. 525–528, IEEE, 2009.

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

[16] I. McGraw, I. Badr, and J. R. Glass, "Learning lexicons from speech using a pronunciation mixture model," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 2, pp. 357–366, 2013.

[17] 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.

[18] 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 LRs in NEMLAR project," in *Proceedings of LREC*, 2006.

[19] 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, no. 4, pp. 357–366, 1980.

[20] M. J. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[21] C. J. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171–185, 1995.

[22] 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.