Improving the Arabic Pronunciation Dictionary for Phone and Word Recognition with Linguistically-Based Pronunciation Rules

Fadi Biadsy∗ and **Nizar Habash**† and **Julia Hirschberg**∗
∗Department of Computer Science, Columbia University, New York, USA
{fadi,julia}@cs.columbia.edu
†Center for Computational Learning Systems, Columbia University, New York, USA
habash@ccls.columbia.edu

Abstract

In this paper, we show that linguistically motivated pronunciation rules can improve phone and word recognition results for Modern Standard Arabic (MSA). Using these rules and the MADA morphological analysis and disambiguation tool, multiple pronunciations per word are automatically generated to build two pronunciation dictionaries; one for training and another for decoding. We demonstrate that the use of these rules can significantly improve both MSA phone recognition and MSA word recognition accuracies over a baseline system using pronunciation rules typically employed in previous work on MSA Automatic Speech Recognition (ASR). We obtain a significant improvement in absolute accuracy in phone recognition of 3.77%–7.29% and a significant improvement of 4.1% in absolute accuracy in ASR.

1 Introduction

The correspondence between orthography and pronunciation in Modern Standard Arabic (MSA) falls somewhere between that of languages such as Spanish and Finnish, which have an almost one-to-one mapping between letters and sounds, and languages such as English and French, which exhibit a more complex letter-to-sound mapping (El-Imam, 2004). The more complex this mapping is, the more difficult the language is for Automatic Speech Recognition (ASR).

An essential component of an ASR system is its pronunciation dictionary (lexicon), which maps the orthographic representation of words to their phonetic or phonemic pronunciation variants. For languages with complex letter-to-sound mappings, such dictionaries are typically written by hand. However, for morphologically rich languages, such as MSA,[1] pronunciation dictionaries are difficult to create by hand, because of the large number of word forms, each of which has a large number of possible pronunciations. Fortunately, the relationship between orthography and pronunciation is relatively regular and well understood for MSA. Moreover, recent automatic techniques for morphological analysis and disambiguation (MADA) can also be useful in automating part of the dictionary creation process (Habash and Rambow, 2005; Habash and Rambow, 2007) Nonetheless, most documented Arabic ASR systems appear to handle only a subset of Arabic phonetic phenomena; very few use morphological disambiguation tools.

In Section 2, we briefly describe related work, including the baseline system we use. In Section 3, we outline the linguistic phenomena we believe are critical to improving MSA pronunciation dictionaries. In Section 4, we describe the pronunciation rules we have developed based upon these linguistic phenomena. In Section 5, we describe how these rules are used, together with MADA, to build our pronunciation dictionaries for training and decoding automatically. In Section 6, we present results of our evaluations of our phone- and word-recognition systems (XPR and XWR) on MSA comparing these systems to two baseline systems, BASEPR and BASEWR.

[1] MSA words have fourteen features: part-of-speech, person, number, gender, voice, aspect, determiner proclitic, conjunctive proclitic, particle proclitic, pronominal enclitic, nominal case, nunation, idafa (possessed), and mood. MSA features are realized using both concatenative (affixes and stems) and templatic (root and patterns) morphology with a variety of morphological and phonological adjustments that appear in word orthography and interact with orthographic variations.

We conclude in Section 7 and identify directions for future research.

2 Related Work

Most recent work on ASR for MSA uses a single pronunciation dictionary constructed by mapping every undiacritized word in the training corpus to all of the diacritized Buckwalter analyses and the diacritized versions of this word in the Arabic Treebank (Maamouri et al., 2003; Afify et al., 2005; Messaoudi et al., 2006; Soltau et al., 2007). In these papers, each diacritized word is converted to a single pronunciation with a one-to-one mapping using "very few" unspecified rules. None of these systems use morphological disambiguation to determine the most likely pronunciation of the word given its context. Vergyri et al. (2008) *do* use morphological information to predict word pronunciation. They select the top choice from the MADA (Morphological Analysis and Disambiguation for Arabic) system for each word to train their acoustic models. For the test lexicon they used the undiacritized orthography, as well as all diacritizations found for each word in the training data as possible pronunciation variants. We use this system as our baseline for comparison.

3 Arabic Orthography and Pronunciation

MSA is written in a morpho-phonemic orthographic representation using the *Arabic script*, an alphabet accented with optional diacritical marks.[2] MSA has 34 phonemes (28 consonants, 3 long vowels and 3 short vowels). The Arabic script has 36 basic letters (ignoring ligatures) and 9 diacritics. Most Arabic letters have a one-to-one mapping to an MSA phoneme; however, there are a small number of common exceptions (Habash et al., 2007; El-Imam, 2004) which we summarize next.

3.1 Optional Diacritics

Arabic script commonly uses nine optional diacritics: (a) three short-vowel diacritics representing the vowels /a/, /u/ and /i/; (b) one long-vowel diacritic (Dagger Alif ‘) representing the long vowel /A/ in a small number of words; (c) three *nunation* diacritics (F /an/, N /un/, K /in/) representing a combination of a short vowel and the nominal indefiniteness marker /n/ in MSA; (d) one consonant lengthening diacritic (called Shadda $\sim$) which repeats/elongates the previous consonant (e.g., $kat{\sim}ab$ is pronounced /kattab/); and (e) one diacritic for marking when there is no diacritic (called Sukun o).

Arabic diacritics can only appear *after* a letter. Word-initial diacritics (in practice, only short vowels) are handled by adding an extra Alif ٱ A (also called Hamzat-Wasl) at the beginning of the word. Sentence/utterance initial Hamzat-Wasl is pronounced like a glottal stop preceding the short vowel; however, the sentence medial Hamzat-Wasl is silent except for the short vowel. For example, *Ainkataba kitAbN* is /Ginkataba kitAbun/ but *kitAbN Ainkataba* is /kitAbun inkataba/. A 'real' Hamza (glottal stop) is always pronounced as a glottal stop. The Hamzat-Wasl appears most commonly as the Alif of the definite article *Al*. It also appears in specific words and word classes such as relative pronouns (e.g., *Aly* 'who' and verbs in pattern VII (Ain1a2a3).

Arabic short vowel diacritics are used together with the glide consonant letters w and y to denote the long vowels /U/ (as uw) and /I/ (iy). This makes these two letters ambiguous.

Diacritics are largely restricted to religious texts and Arabic language school textbooks. In other texts, fewer than 1.5% of words contain a diacritic. Some diacritics are lexical (where word meaning varies) and others are inflectional (where nominal case or verbal mood varies). Inflectional diacritics are typically word final. Since nominal case, verbal mood and nunation have all disappeared in spoken dialectal Arabic, Arabic speakers do not always produce these inflections correctly or at all.

Much work has been done on automatic Arabic diacritization (Vergyri and Kirchhoff, 2004; Ananthakrishnan et al., 2005; Zitouni et al., 2006; Habash and Rambow, 2007). In this paper, we use the MADA (Morphological Analysis and Disambiguation for Arabic) system to diacritize Arabic (Habash and Rambow, 2005; Habash and Rambow, 2007). MADA, which uses the Buckwalter Arabic morphological Analyzer databases (Buckwalter, 2004), provides the necessary information to determine Hamzat-Wasl through morphologically tagging the definite article; in most other cases it outputs the special symbol "{" for Hamzat-Wasl.

[2] We provide Arabic script orthographic transliteration in the Buckwalter transliteration scheme (Buckwalter, 2004). For Modern Standard Arabic phonological transcription, we use a variant of the Buckwalter transliteration with the following exceptions: glottal stops are represented as /G/ and long vowels as /A/, /U/ and /I/. All Arabic script diacritics are phonologically spelled out.

3.2 Hamza Spelling

The consonant Hamza (glottal stop /G/) has multiple forms in Arabic script: ء $'$, أ $>$, إ $<$, ؤ $\&$, ئ $\}$, آ $|$. There are complex rules for Hamza spelling that primarily depend on its vocalic context. For example, ئ $\}$ is used word medially and finally when preceded or followed by an /i/ vowel. Similarly, the Hamza form آ $|$ is used when the Hamza is followed by the long vowel /A/.

Hamza spelling is further complicated by the fact that Arabic writers often replace hamzated letters with the un-hamzated form (أ $> \rightarrow$ ا A) or use a two-letter spelling, e.g. ئ $\} \rightarrow$ ىء Y'. Due to this variation, the un-hamzated forms (particularly for أ $>$ and إ $<$) are ignored in Arabic ASR evaluation. The MADA system regularizes most of these spelling variations as part of its analysis.

3.3 Morpho-phonemic Spelling

Arabic script includes a small number of morphemic/lexical phenomena, some very common:

- **Ta-Marbuta** The Ta-Marbuta (p) is typically a feminine ending. It appears word-finally, optionally followed by a diacritic. In MSA it is pronounced as /t/ when followed by a diacritic; otherwise it is silent. For example, $maktabapN$ ‘a library’ is pronounced / maktabatun/.

- **Alif-Maqsura** The Alif-Maqsura (Y) is a silent derivational marker, which always follows a short vowel /a/ at the end of a word. For example, $rawaY$ ‘to tell a story’ is pronounced /rawa/.

- **Definite Article** The Arabic definite article is a proclitic that assimilates to the first consonant in the noun it modifies if this consonant is alveolar or dental (except for j). These are the so-called Sun Letters: *t, v, d, *, r, z, s, $, S, D, T, Z, l,* and *n*. For example, the word *Al$ams* ‘the sun’ is pronounced /a$$ams/ not */al$ams/. The definite article does not assimilate to the other consonants, the Moon Letters. For example, the word Alqamar ‘the moon’ is pronounced /alqamar/ not */aqqamar/.

- **Silent Letters** A silent Alif appears in the morpheme $+uwA$ /U/ which indicates masculine plural conjugation in verbs. Another silent Alif appears after some nunated nouns, e.g., kitaAbAF /kitAban/. In some poetic readings, this Alif can be produced as the long vowel /A/: /kitAbA/. Finally, a common odd spelling is that of the proper name $Eamrw$ /Eamr/ ‘Amr’where the final w is silent.

4 Pronunciation Rules

As noted in Section 3, diacritization alone does not predict actual pronunciation in MSA. In this section we describe a set of rules based on MSA phonology which will extend a diacritized word to a set of possible pronunciations. It should be noted that even MSA-trained speakers, such as broadcast news anchors, may not follow the “proper” pronunciation according to Arabic syntax and phonology. So we attempt to accommodate these pronunciation variants in our pronunciation dictionary.

The following rules are applied on each diacritized word.[3] These rules are divided into four categories: (I) a shared set of rules used in all systems compared (BASEPR, BASEWR, XPR and XWR);[4] (II) a set of rules in BASEPR and BASEWR which we modified for XPR and XWR; (III) a first set of new rules devised for our systems XPR and XWR; and (IV) a second set of new rules that generate additional pronunciation variants. Below we indicate, for each rule, how many words in the training corpus (335,324 words) had their pronunciation affected by the rule.

I. Shared Pronunciation Rules

1. **Dagger Alif:** ‘ → /A/
   (e.g., h‘*A → hA*A) (This rule affected 1.8% of all the words in our training data)

2. **Madda:** | → /G A/
   (e.g., Al|n → AlGAn) (affected 1.9%)

3. **Nunation:** AF → /a n/, F → /a n/, /K/ → /i n/, N → /u n/
   (e.g., kutubAF → kutuban) (affected 9.7%)

4. **Hamza:** All Hamza forms: $', \}, \&, <, >$ → /G/
   (e.g., >kala → Gakala) (affected 21.3%)

[3] Our script that generates the pronunciation dictionaries from MADA output can be downloaded from *www.cs.columbia.edu/speech/software.cgi*.

[4] We have attempted to replicate the baseline pronunciation rules for (Vergyri et al., 2008) based on published work and personal communications with the authors.

5. **Ta-Marbuta:** p → /t/
(e.g., madrasapa → madrasata) (affected 15.3%)

II. Modified Pronunciation Rules

1. **Alif-Maqsura:** Y → /a/
(e.g., salomY → saloma) (affected 4.2%) *(Baseline: Y → /A/)*

2. **Shadda:** Shadda is always removed
(e.g., ba$∼ara → ba$ara) (affected 23.8%) *(Baseline: the consonant was doubled)*

3. **U and I:** uwo → /U/, iyo → /I/
(e.g., makotuwob → makotUb) (affected 25.07%) *(Baseline: same rule but it inaccurately interacted with the baseline Shadda rule)*

III. New Pronunciation Rules

1. **Waw Al-jamaa:** suffixes uwoA → /U/
(e.g., katabuwoA → katabU) (affected 0.4%)

2. **Definite Article:** Al → /a l/ (if tagged as Al+ by MADA)
(e.g., wAlkitAba → walkitAba) (affected 30.0%)

3. **Hamzat-Wasl:** { is always removed.
(affected 3.0%)

4. **"Al" in relative pronouns:** Al → /a l/
(affected 1.3%)

5. **Sun letters:** if the definite article (Al) is followed by a sun letter, remove the *l*.
(e.g., Al$amsu → A$amsu) (affected 8.1%)

IV. New Pronunciation Rules Generating Additional Variants

- **Ta-Marbuta:** if a word ends with Ta-Marbuta (p) followed by any diacritic, remove the Ta-Marbuta and its diacritic. Apply the rules above (I-III) on the modified word and add the output pronunciation.
(e.g., marbwTapF → marbwTa) (affected 15.3%)

- **Case ending:** if a word ends with a short vowel (a, u, i), remove the short vowel. Apply rules (I-III) on the modified word, and add the output pronunciation
(e.g., yaktubu → yaktub (affected 60.9%)

As a post-processing step in all systems, we remove the Sukun diacritic and convert every letter X to phoneme /X/. In XPR and XWR, we also remove short vowels that precede or succeed a long vowel.

5 Building the Pronunciation Dictionaries

As noted above, pronunciation dictionaries map words to one or more phonetically expressed pronunciation variants. These dictionaries are used for training and decoding in ASR systems. Typically, most data available to train large vocabulary ASR systems is orthographically (not phonetically) transcribed. There are two well-known alternatives for training acoustic models in ASR: (1) bootstrap training, when some phonetically annotated data is available, and (2) flat-start, when such data is not available (Young et al., 2006). In flat-start training, for example, the pronunciation dictionary is used to map the orthographic transcription of the training data to a sequence of phonetic labels to train the initial monophone models. Next, the dictionary is employed again to produce networks of possible pronunciations which can be used in forced alignment to obtain the most likely phone sequence that matches the acoustic data. Finally, the monophone acoustic models are re-estimated. In our work, we refer to this dictionary as the **training pronunciation dictionary**. The second usage of the pronunciation dictionary is to generate the pronunciation models while decoding. We refer to this dictionary as the **decoding pronunciation dictionary**.

For languages like English, no distinction between decoding and training pronunciation dictionaries is necessary. However, as noted in Section 3, short vowels and other diacritic markers are typically not orthographically represented in MSA texts. Thus ASR systems typically do not output fully diacritized transcripts. Diacritization is generally not necessary to make the transcript readable by Arabic-literate readers. Therefore, entries in the decoding pronunciation dictionary consist of undiacritized words that are mapped to a set of phonetically-represented diacritizations. However, every entry in the training pronunciation dictionary is a fully diacritized word mapped to a set of possible context-dependent pronunciations. Particularly in the training step, contextual information for each word is available from the transcript, so, for our work, we can use the MADA morphological tagger to obtain the most likely diacritics. As a result, the speech signal is mapped to a more accurate representation of the training transcript, which we hypothesize will lead to a better estimation of the acoustic models.

As noted in Section 1, pronunciation dictionaries for ASR systems are usually written by hand. However, Arabic's morphological richness makes it difficult to create a pronunciation dictionary by hand since there are a very large number of word forms, each of which has a large number of possible pronunciations. The relatively regular relationship between orthography and pronunciation and tools for morphological analysis and disambiguation such as MADA, however, make it possible to create such dictionaries automatically with some success.[5]

5.1 Training Pronunciation Dictionary

In this section, we describe an automatic approach to building a pronunciation dictionary for MSA that covers all words in the orthographic transcripts of the training data. First, for each word in each utterance, we run MADA to disambiguate the word based on its context in the transcript. MADA outputs all possible fully-diacritized morphological analyses, ranked by their likelihood, the MADA confidence score.[6] We thus obtain a fully-diacritized orthographic transcription for training. Second, we map the highest-ranked diacritization of each word to a set of pronunciations, which we obtain from the pronunciation rules described in Section 4. Since MADA may not always rank the best analysis as its top choice, we also run the pronunciation rules on the **second** best choice returned by MADA, when the difference between the top two choices is less than a threshold determined empirically (in our implementation we chose 0.2). In Figure 1, the training pronunciation dictionary maps the 2^{nd} column (the entry keys) to the 3^{rd} column.

We generate the baseline training pronunciation dictionary using only the baseline rules from Section 4. This dictionary also makes use of MADA, but it maps the MADA-diacritized word to only one pronunciation. The baseline training dictionary maps the 2^{nd} column (the entry keys) to **only** one pronunciation in the 3^{rd} column in Figure 1.

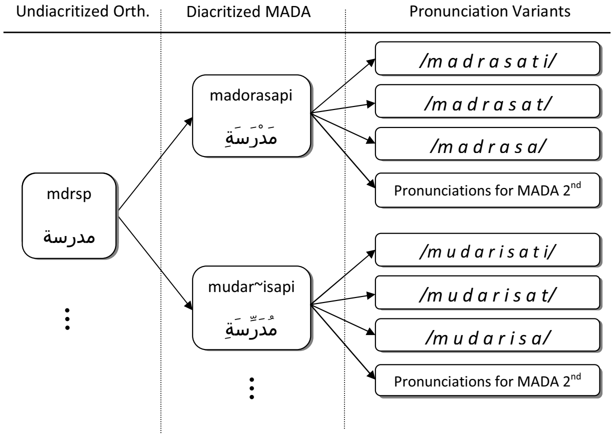

Figure 1: Mapping an undiacritized word to MADA outputs to possible pronunciations.

5.2 Decoding Pronunciation Dictionary

The decoding pronunciation dictionary is used in ASR to build the pronunciation models while decoding. Since, as noted above, it is standard to produce unvocalized transcripts when recognizing MSA, we must map word pronunciations to unvocalized orthographic output. Therefore, for each diacritized word in our training pronunciation dictionary, we remove diacritic markers and replace Hamzat-Wasl ({), <, and > by the letter 'A', and then map the modified word to the set of pronunciations for that word. For example, in Figure 1 the undiacritized word *mdrsp* in the 1^{st} column is mapped to the pronunciations in the 3^{rd} column. The baseline decoding pronunciation dictionary is constructed similarly from the baseline training pronunciation dictionary.

6 Evaluation

To determine whether our pronunciation rules are useful in speech processing applications, we evaluated their impact on two tasks, automatic phone recognition and ASR. For our experiments, we used the broadcast news TDT4 corpus (Arabic Set 1), divided into 47.61 hours of speech (89 news shows) for training and 5.18 hours (11 shows); test and training shows were selected at random. Both training and test data were segmented based on silence and non-speech segments and down-sampled to 8Khz.[7] This segmentation produced 20,707 speech segments for our training data and 2,255 segments for testing.

[5] The MADA system (Habash and Rambow, 2005; Habash and Rambow, 2007) reports 4.8% diacritic error rate (DER) on all diacritics and 2.2% (DER) when ignoring the last (inflectional) diacritic.

[6] In our training data, only about 1% of all words are not diacritized because of lack of coverage in the morphological analysis component.

[7] One of our goals is phone recognition telephone conversation for Arabic dialect identifaction, hence the down-sampling.

6.1 Acoustic Models

Our monophone acoustic models are built using 3-state continuous HMMs without state-skipping with a mixture of 12 Gaussians per state. We extract standard MFCC (Mel Frequency Cepstral Coefficients) features from 25 ms frames, with a frame shift of 10 ms. Each feature vector is 39D: 13 features (12 cepstral features plus energy), 13 deltas, and 13 double-deltas. The features are normalized using cepstral mean normalization. For our ASR experiments, tied context-dependent cross-word triphone HMMs are created with the same settings as monophones. The acoustic models are speaker- and gender-independent, trained using ML (maximum likelihood) with flat-start.[8] We build our framework using the HMM Toolkit (HTK) (Young et al., 2006).

6.2 Phone Recognition Evaluation

We hypothesize that improved pronunciation rules will have a profound impact on phone recognition accuracy. To compare our phone recognition (XPR) system with the baseline (BASEPR), we train two phone recognizers using HTK. The BASEPR recognizer uses the training-pronunciation dictionary generated using the baseline rules; the XPR system uses a pronunciation dictionary generated using these rules plus our modified and new rules (cf. Section 5). The two systems are identical except for their pronunciation dictionaries.

We evaluate the two systems under two conditions: (1) phone recognition with a bigram phone language model (LM)[9] and (2) phone recognition with an open-loop phone recognizer, such that any phoneme can follow any other phoneme with a uniform distribution. Results of this evaluation are presented in Table 1.

Ideally, we would like to compare the performance of these systems against a common MSA phonetically-transcribed gold standard. Unfortunately, to our knowledge, such a data set does not exist. So we approximate such a gold standard on a blind test set through forced alignment, using the trained acoustic models and pronunciation dictionaries. Since our choice of acoustic model (of BASEPR or XPR) and pronunciation dictionary (again of BASEPR or XPR) can bias our results, we consider four *gold* variants (GV) with different combinations of acoustic model and pronunciation dictionary, to set expected lower and upper bounds. These combinations are represented in Table 1 as GV1–4, where the source of acoustic models is BASEPR or XPR and source of pronunciation rules are BASEPR, XPR or XPR and BASEPR combined. These GV are described in more detail below, as we describe our results.

Since BASEPR system uses a pronunciation dictionary with a one-to-one mapping of orthography to phones, the GV1 phone sequence for any test utterance's orthographical transcript according to BASEPR can be obtained directly from the orthographic transcript. Note that if, in fact, GV1 does represent the true gold standard (i.e., the correct phone sequence for the test utterances) then if XPR obtains a lower phone error rate using this gold standard than BASEPR does, we can conclude that in fact XPR's acoustic models are better estimated. This is in fact the case. In Table 1, first line, we see that XPR under both conditions (open-loop and bigram LM) significantly (p-value $< 2.2e-16$) outperforms the corresponding BASEPR phone recognizer using GV1.[10]

If GV1 does *not* accurately represent the phone sequences of the test data, then there must be some phones in the GV1 sequences that should be deleted, inserted, or substituted. On the hypothesis that our training-pronunciation dictionary might improve the BASEPR assignments, we enrich the baseline pronunciation dictionary with XPR's dictionary. Now, we force-align the orthographic transcript using this extended pronunciation dictionary, still using BASEPR's acoustic models, with the acoustic signal. We denote the output phone sequences as GV2. If a pronunciation generated using the BASEPR dictionary was already correct (in GV1) according to the acoustic signal, this forced alignment process still has the option of choosing it. We hypothesize that the result, GV2, is a more accurate representation of the true phone sequences in the test data, since it should be able to model the acoustic signal more accurately. On GV2, as on GV1, we see that XPR, under both conditions, significantly (p-

[8] Since our focus is a comparison of different approaches to pronunciation modeling on Arabic recognition tasks, we have not experimented with different features, parameters, and different machine learning approaches (such as discriminative training and/or the combination of both).

[9] The bigram phoneme LM of each phone recognizer is trained on the phonemes obtained from forced aligning the training transcript to the speech data using that recognizer's training pronunciation dictionary and acoustic models.

[10] Throughout this discussion we use paired t-tests to measure significant difference, where the sample values are the phone recognizer accuracies on the utterances.

Gold Variants			*Open-loop (Accuracy)*		*Bigram Phone LM (Accuracy)*	
GV	*Acoustic Model of*	*Pron. Dict. of*	BASEPR	XPR	BASEPR	XPR
1	BASEPR	BASEPR	37.40	39.21	41.56	45.17
2	BASEPR	BASEPR+XPR	38.64	42.41	43.44	50.73
3	XPR	XPR	37.06	42.38	42.21	51.41
4	XPR	BASEPR+XPR	37.47	42.74	42.59	51.51

Table 1: Comparing the effect of BASEPR and XPR pronunciation rules, alone and in combination, using 4 Gold Variants under two conditions (Open-loop and LM)

value $< 2.2e - 16$) outperforms the corresponding BASEPR phone recognizers (see Table 1, second line).

We also compared the performance of the two systems using upper bound variants. For GV3 we used the forced alignment of the orthographic transcription using only XPR's pronuncation dictionary with XPR's acoustic models. In GV4 we combine the pronunciation dictionary of XPR with BASEPR dictionary and use XPR's acoustic models. Unsurprisingly, we find that the XPR recognizer significantly (p-value $<2.2e - 16$) outperforms BASEPR when using these two variants under both conditions (see Table 1, third and fourth lines).

The results presented in Table 1 compare the robustness of the acoustic models as well as the pronunciation components of the two systems. We also want to evaluate the accuracy of our pronunciation predictions in representing the actual acoustic signal. One way to do this is to see how often the forced alignment process choose phone sequences using the BASEPR pronunciation dictionary as opposed to XPR's. We forced aligned the test transcript — using the XPR acoustic models and only the XPR pronunciation dictionary — with the acoustic signal. We then compare the output sequences to the output of the forced alignment process where the **combined** pronunciations from BASEPR+XPR and the XPR acoustic models were used. We find that the difference between the two is only 1.03% (with 246,376 phones, 557 deletions, 1696 substitutions, and 277 insertions). Thus, adding the BASEPR rules to XPR does not appear to contribute a great deal to the representation chosen by forced alignment. In a similar experiment, we use the BASEPR acoustic models instead of the XPR models and compare the results of using BASEPR-pronunciation dictionary with the combination of XPR+BASEPR's dictionaries for forced alignment. Interestingly, in this experiment we *do* find a significantly larger difference between the two outputs 17.04% (with 233,787 phones, 1404 deletions, 14013 substitutions, and 27040 insertions). We can hypothesize from these experiments that the baseline pronunciation dictionary alone is not sufficient to represent the acoustic signal accurately, since large numbers of phonemes are edited when adding the XPR pronunciations. In contrast, adding the BASEPR's pronunciation dictionary to XPR's shows a relatively small percentage of edits, which suggests that the XPR pronunciation dictionary extends and covers more accurately the pronunciations already contained in the BASEPR dictionary.

6.3 Speech Recognition Evaluation

We have also conducted an ASR experiment to evaluate the usefulness of our pronunciation rules for this application.[11] We employ the baseline pronunciation rules to generate the baseline training and decoding pronunciation dictionaries. Using these dictionaries, we build the baseline ASR system (BASEWR). Using our extended pronunciation rules, we generate our dictionaries and train our ASR system (XWR). Both systems have the same model settings, as described in Section 6.1. They also share the same language model (LM), a trigram LM trained on the undiacritized transcripts of the training data and a subset of Arabic gigawords (approximately 281 million words, in total), using the SRILM toolkit (Stolcke, 2002).

Table 2 presents the comparison of BASEWR with the XWR system. In Section 5.1, we noted that the top two choices from MADA may be included in the XWR pronunciation dictionary when the difference in MADA confidence scores for these two is less than a given threshold. So we analyze the impact of including this second MADA option in both the training and decoding dictionaries on ASR results. In all cases, whether the second MADA choice

[11] It should be noted that we have not attempted to build a state-of-the-art Arabic speech recognizer; our goal is purely to evaluate our approach to pronunciation modeling for Arabic.

is included or not, XWR significantly (p-values $<$ 8.1e-15) outperforms BASEWR. Our best results are obtained when we include the top first and second MADA option in the decoding pronunciation dictionary but **only** the top MADA choice in the training pronunciation dictionary. The difference between this version of XWR and an XWR version which includes the top second MADA choice in the training dictionary is significant (p-value = 0.017).

To evaluate the impact of the set of rules that generate additional pronunciation variants (described in Section 4 - IV) on word recognition, we built a system, denoted as XWR_I-III, that uses only the first three sets of rules (I–III) and compared its performance to that of both BASEWR and the corresponding XWR system. As shown in Table 2, we observe that XWR_I-III significantly outperforms BASEWR in 2.27 (p-value < 2.2e-16). Also, the corresponding XWR that uses all the rules (including IV set) significantly outperforms XWR_I-III in 1.24 (p-value < 2.2e-16).

The undiacritized vocabulary size used in our experiment was 34,511. We observe that 6.38% of the words in the test data were out of vocabulary (OOV), which may partly explain our low absolute recognition accuracy. The dictionary size statistics (for entries generated from the training data only) used in these experiments are shown in Table 3. We have done some error analysis to understand the reason behind high absolute error rate for both systems. We observe that many of the test utterances are very noisy. We wanted to see whether XWR still outperforms BASEWR if we remove these utterances. Removing all utterances for which BASEWR obtains an accuracy of less than 25%, we are left with 1720/2255 utterances. On these remaining utterances, the BASEWR accuracy is 64.4% and XWR's accuracy is 67.23% — a significant difference despite the bias in favor of BASEWR.

System	Acc	Corr	Del	Sub	Ins
BASEWR	52.78	65.36	360	12297	4598
XWR_I–III (1 TD/DD)	55.05	66.84	324	11791	4308
XWR (1 TD/DD)	56.29	69.06	274	11031	4665
XWR (2 TD, 2 DD)	56.28	69.12	274	11008	4694
XWR (2 TD, 1 DD)	55.53	68.55	285	11206	4759
XWR (1 TD, 2 DD)	56.88	69.42	284	10891	4579

Table 2: Comparing the performance of BASEWR to XWR, where the top 1 or 2 MADA options are included in the training dictionary (TD) and decoding dictionary (DD). XWR I–III uses only the first three sets of pronunciation rules in Section 4. **Acc**uracy = (100 - WER); **Corr**ect is Accuracy without counting insertions (%). Total number of words is 36,538.

Dictionary	# entries	PPW
BASEPR TD	45,117	1
BASEPR DD	44,383	1.3
XPR TD (MADA top 1)	80,200	1.78
XPR TD (MADA top 1 and 2)	128,663	2.85
XWR DD (MADA top 1)	71,853	2.08
XWR DD (MADA top 1 and 2)	105,402	3.05

Table 3: Dictionary sizes generated fom the training data only (PPW: pronunciations per word, TD: Training pronunciation dictionary, DD: Decoding pronunciation dictionary).

7 Conclusion and Future Work

In this paper we have shown that the use of more linguistically motivated pronunciation rules can improve phone recognition and word recognition results for MSA. We have described some of the phonetic, phonological, and morphological features of MSA that are rarely modeled in ASR systems and have developed a set of pronunciation rules that encapsulate these features. We have demonstrated how the use of these rules can significantly improve both MSA phone recognition and MSA word recognition accuracy by a series of experiments comparing our XPR and XWR systems to the corresponding baseline systems BASEPR and BASEWR. We obtain an improvement in absolute accuracy in phone recognition of 3.77%–7.29% and a significant improvement of 4.1% in absolute accuracy in ASR.

In future work, we will address several issues which appear to hurt our recognition accuracy, such as handling the words that MADA fails to analyze. We also will develop a similar approach to handling dialectical Arabic speech using the MAGEAD morphological analyzer (Habash and Rambow, 2006). A larger goal is to employ the MSA and dialectical phone recognizers to aid in spoken Arabic dialect identification using phonotactic modeling (see (Biadsy et al., 2009)).

Acknowledgments

We are very thankful to Dimitra Vergyri for providing us with the details of the baseline pronunciation dictionary and for her useful suggestions. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023 (approved for public release, distribution unlimited). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

References

M. Afify, L. Nguyen, B. Xiang, S. Abdou, and J. Makhoul. 2005. Arabic broadcast news transcription using a one million word vocalized vocabulary. In *Proceedings of Interspeech 2005*, pages 1637–1640.

S. Ananthakrishnan, S. Narayanan, and S. Bangalore. 2005. Automatic diacritization of arabic transcripts for asr. In *Proceedings of ICON*, Kanpur, India.

F. Biadsy, J. Hirschberg, and N. Habash. 2009. Spoken Arabic Dialect Identification Using Phonotactic Modeling. In *Proceedings of EACL 2009 Workshop on Computational Approaches to Semitic Languages*, Athens, Greece.

T. Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Cat alog No.: LDC2004L02, ISBN 1-58563-324-0.

Y. A. El-Imam. 2004. Phonetization of Arabic: rules and algorithms. In *Computer Speech and Language 18*, pages 339–373.

N. Habash and O. Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 573–580.

N. Habash and O. Rambow. 2006. MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 681–688, Sydney, Australia.

N. Habash and O. Rambow. 2007. Arabic Diacritization through Full Morphological Tagging. In *Proceedings of the 8th Meeting of the North American Chapter of the Association for Computational Linguistics/Human Language Technologies Conference (HLT-NAACL07)*.

N. Habash, A. Soudi, and T. Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

M. Maamouri, A. Bies, H. Jin, and T. Buckwalter. 2003. Arabic treebank: Part 1 v 2.0. Distributed by the Linguistic Data Consortium. LDC Catalog No.: LDC2003T06.

A. Messaoudi, J. L. Gauvain, and L. Lamel. 2006. Arabic broadcast news transcription using a one million word vocalized vocabulary. In *Proceedings of ICASP 2006*, volume 1, pages 1093–1096.

H. Soltau, G. Saon, D. Povey, L. Mangu, B. Kingsbury, J. Kuo, M. Omar, and G. Zweig. 2007. The IBM 2006 GALE Arabic ASR System. In *Proceedings of ICASP 2007*.

S. Stolcke. 2002. Tokenization, Morphological Analysis, and Part-of-Speech Tagging for Arabic in One Fell Swoop. In *Proceedings of ICSLP 2002*, volume 2, pages 901–904.

D. Vergyri and K. Kirchhoff. 2004. Automatic Diacritization of Arabic for Acoustic Modeling in Speech Recognition. In Ali Farghaly and Karine Megerdoomian, editors, *COLING 2004 Workshop on Computational Approaches to Arabic Script-based Languages*, pages 66–73, Geneva, Switzerland.

D. Vergyri, A. Mandal, W. Wang, A. Stolcke, J. Zheng, M. Graciarena, D. Rybach, C. Gollan, R. Schluter, K. Kirchhoff, A. Faria, and N. Morgan. 2008. Development of the SRI/Nightingale Arabic ASR system. In *Proceedings of Interspeech 2008*, pages 1437–1440.

S. Young, G. Evermann, M. Gales, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. 2006. The HTK Book, version 3.4: htk.eng.cam.ac.uk. Cambridge University Engineering Department.

I. Zitouni, J. S. Sorensen, and R. Sarikaya. 2006. Maximum Entropy Based Restoration of Arabic Diacritics. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 577–584, Sydney, Australia.