DEVELOPMENT OF A PHONETIC SYSTEM FOR LARGE VOCABULARY ARABIC SPEECH RECOGNITION

M.J.F. Gales, F. Diehl, C.K. Raut, M. Tomalin, P.C. Woodland and K. Yu

Cambridge University Engineering Department
Cambridge CB2 1PZ, UK

ABSTRACT

This paper describes the development of an Arabic speech recognition system based on a phonetic dictionary. Though phonetic systems have been previously investigated, this paper makes a number of contributions to the understanding of how to build these systems, as well as describing a complete Arabic speech recognition system. The first issue considered is discriminative training when there are a large number of pronunciation variants for each word. In particular, the loss function associated with Minimum Phone Error (MPE) training is examined. The performance and combination of phonetic and graphemic acoustic models are then compared on both Broadcast News (BN) and Broadcast Conversation (BC) data. The final contribution of the paper is a simple scheme for automatically generating pronunciations for use in training and reducing the phonetic out-of-vocabulary rate. The paper concludes with a description and results from using phonetic and graphemic systems in a multi-pass/combination framework.

Index Terms— Large vocabulary speech recognition, Arabic, discriminative training.

1. INTRODUCTION

In recent years there has been interest in transcribing Arabic Broadcast news [1, 2]. Compared to English there are a number of issues in automatically transcribing Arabic speech [2]. In Arabic texts the short vowels are not normally marked, this is also true for languages such as Farsi and Hebrew. This means that each "word" in the text may have a large number of pronunciations, with the pronunciations being associated with different, but possibly related, meanings. In addition, Arabic is a highly inflected agglutinative language. This results in a large vocabulary, as words are often formed by attaching affixes to triconsonantal roots. Techniques such as morphological analysis may be used to handle this problem [3]. This paper is primarily concerned with the first issue, handling the modelling of short vowels in Arabic.

There are two approaches to handling the lack of vowel-markings in written Arabic. The first is to rely on the acoustic models to implicitly model the vowels. Here the pronunciations can be simply based on the orthographic form of the word. This is referred to as a *graphemic* system. Alternatively a dictionary that explicitly includes the vowels can be constructed. This is a *phonetic* system. In this case it is necessary to generate pronunciations for each orthographic transcription, in a similar fashion to English. This paper considers some of the design issues and options when constructing these phonetic systems. In particular the following issues are addressed: the impact of having large numbers of pronunciations on discriminative training (section 4.1); the interaction/combination of phonetic and graphemic systems (section 4.3); and automatically deriving pronunciations (section 5.1).

This work was supported in part under the GALE program of the Defense Advanced Research Projects Agency, Contract No. HR0011-06-C-0022. Many thanks to BBN for supplying segmentations, lightly supervised and unsupervised transcriptions for the BN03 data and the intial GALE Arabic data releases. Also the authors would like to thank BBN and LIMSI for many helpful discussions about initial pronunciation generation for the phonetic system.

2. GRAPHEMIC AND PHONETIC DICTIONARIES

There are two forms of Arabic system that are commonly constructed; graphemic and phonetic systems. In the graphemic system a dictionary is generated using one-to-one letter-to-sound rules for each word. Note for all this work the Arabic text is *romanised* and the word and letter order swapped to be left-to-right. Thus the dictionary entry for the word كتاب, "book" in Arabic, is

```
ktAb        /k/ /t/ /A/ /b/
```

This scheme yields 28 consonants, four *alif* variants (*madda* and *hamza* above and below), *ya* and *wa* variants (*hamza* above), *tamarbuta* and *hamza*. Thus the total number of graphemes is 36 (excluding silence).

In Arabic the short vowels (*fatha* `/a/`, *kasra* `/i/` and *damma* `/u/`) and diacritics (*shadda*, *sukun*) are commonly not marked in texts. Additionally, *nunation* can result in a word-final *nun* (`/n/`) being added to nouns and adjectives in order to indicate that they are unmarked for definiteness. In graphemic systems the acoustic models are required to model the implied pronunciation variations implicitly. An alternative approach is to use a phonetic system where the pronunciations for each word explicity include the short vowels and nun. Note in this work the other diacritics are not considered and are required to be implicitly modelled. In phonetic systems it is necessary to hypothesise the forms of the pronunciations that can be used. The baseline process used in this work is to use the Buckwalter Morphological Analyser (version 2.0)[1], referred to as Buckwalter in this paper. All initial recognition dictionaries were based on this analysis. However for training data Buckwalter was used in combination with the Treebank and the FBIS pronunciations (similar to the procedure described in [1]). Here the following strategy is used:

$$\text{Buckwalter} \rightarrow \text{Treebank} \rightarrow \text{FBIS pron.}$$

where $\rightarrow$ means if the word is not found in the left dictionary search in the the right dictionary. This expands the coverage for the training data and is not felt to be a major issue as inconsistencies in the dictionaries will minimally impact other words as training is an alignment

[1] Available at http://www.qamus.org/index.html.

process. In contrast for decoding, an inconsistent dictionary may affect both the word in question and the surrounding words. Again taking the example of the dictionary entry for the word كتاب

```
ktAb        /k/ /i/ /t/ /A/ /b/
```

Some simple mapping rules were used to reduce the number of "phones" given the explicit vowel modelling. The four variant forms of *alif* and *hamza* were mapped to the simple *alif*, and *ya* and *wa* variants were both mapped to their respective simple forms. Thus the total number of "phones" is 32, again excluding silence.

3. TRAINING DATA AND TEST SETS

Sys.	FBTD	BN03	GALE-Y1/P2R{1,2,3}			Total
	l/supv	usupv	supv	lsupv	usupv	(Hrs)
G0	100.7	—	—	—	—	100.7
G1	109.6	791.0	—	39.8	287.6	1228.0
G2	109.6	791.0	655.0	39.8	219.8	1815.2
G3	109.6	—	655.0	39.8	219.8	1024.2
V0	101.8	—	—	—	—	101.8
V1	101.8	791.0	—	39.8	287.6	1220.2
V2	101.8	—	439.3	39.8	219.8	800.7
V3	101.8	—	653.7	39.8	219.8	1015.1

Table 1. Training data used for all the system evaluated in this paper, hours of supervised (supv), unsupervised (usupv) and lightly supervised (lsupv) are given for FBIS and TDT4 (FBTD), EARS BN03 (BN03) and the GALE data.

The training data, summarised for each of the systems in Table 1, was used in three distinct stages. The first stage, was used for all the initial phonetic system development in section 4. The data consists of the FBIS data for which detailed transcriptions (including short vowels and diacritic markings) are available and the TDT4 Arabic data which was used in a lightly supervised fashion, where a biased language model is generated and the training data recognised [4]. If a phonetic system is used for the recogniser then it is guaranteed that all pronunciations are available for all the light-supervision data. However, for the TDT4 data a graphemic system was used, as this was the "best" system available. This resulted in approximately 7.8 hours of TDT4 data not being available for the phonetic system. The graphemic G0 system was built on a reduced subset to give approximately the same amount of training data as the initial V0 system.

The second block of training data was used to generate the G1 and V1 acoustic models. This data consisted of the BN03 data (data collected for the EARS programme in 2003) and the GALE-Y1Q{1,2} data releases. None of this data was used in a supervised fashion. 39.8 hours of the data had lightly supervised transcriptions from BBN[2], the remaining data had unsupervised transcriptions also supplied by BBN. As lightly supervised and unsupervised training was used, very few segments did not have phonetic pronunciations available (BBN used slightly different mappings for their phonetic system). This was used to build the G1 and V1 acoustic models. The final block of data consisted of the GALE-Y1Q{3,4} and GALE-P2R{1,2,3} data. Some of the transcriptions available with this data overlapped with the unsupervised transcriptions in the second-block of data. In this case the supervised transcriptions were used. This final block of data was used in a supervised fashions.

Two test sets defined by BBN were used for evaluating the systems. The first, `bnat06`, consists of about 3 hours of Broadcast News (BN) style data collected in November 2005 and January 2006. The second, `bcat06` is about 3 hours of Broadcast Conversation (BC) style data collected in January 2006. It was ensured that there was no overlap between this training and test data. The test data had to be segmented and speaker clustered. The initial segmentation and clustering used will be referred to as the CU segmentation. For a description of the language models and training data see section 5.2.

[2]The segmentation supplied with the transcription was not sufficiently accurate to allow them to be directly used. For further details of the data see the LDC web-site - http://projects.ldc.upenn.edu/gale/data/DataMatrix.html.

4. INITIAL PHONETIC SYSTEM DEVELOPMENT

This section describes the initial phonetic system development. The baseline phonetic (V0) and graphemic systems (G0) were used. Both these systems are state-clustered decision-tree triphone systems with approximately 4K distinct states and an average of 16 Gaussian components per state. The baseline 65K vocabulary-size language models, LM1 for the graphemic system and LM2 for the phonetic system, are used in these initial experiments, for details of these see section 5.2.

4.1. Discriminative Training with Multiple Pronunciations

One interesting aspect of phonetic models for Arabic is that each of the "words" has a large number of pronunciations. The average number of pronunciations, using a 250K word vocabulary and Buckwalter, is 4.3 per word. In contrast, using a 59K word vocabulary English system there are only 1.1 pronunciations per word. In addition to this approximately four-fold increase in the number of pronunciations, the nature of the pronunciations is fundamentally different. In Arabic the multiple pronunciations are really distinct words, but all mapped to the same vocabulary entry. This may be expected to have an impact on discriminative training. Two forms of discriminative training are examined, Maximum Mutual Information (MMI) and Minimum Phone Error (MPE) training.

The training criterion for MMI training can be expressed as

$$\mathcal{F}_{\texttt{mmi}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\log\left(\frac{p(\mathbf{O}^{(r)}|\mathbf{w}_{\texttt{ref}}^{(r)};\boldsymbol{\lambda})P(\mathbf{w}_{\texttt{ref}}^{(r)})}{\sum_{\mathbf{w}}p(\mathbf{O}^{(r)}|\mathbf{w};\boldsymbol{\lambda})P(\mathbf{w})}\right) \quad (1)$$

where $\boldsymbol{\lambda}$ is the set of model parameters, $\mathbf{O}^{(r)}$ is the r^{th} observation sequence and $\mathbf{w}_{\texttt{ref}}^{(r)}$ is the associated word-sequence reference. When using multiple pronunciations the likelihood may be expressed as

$$p(\mathbf{O}^{(r)}|\mathbf{w};\boldsymbol{\lambda}) = \sum_{\mathbf{v}\in\mathbf{V}_{\mathbf{w}}} P(\mathbf{v}|\mathbf{w})p(\mathbf{O}^{(r)}|\mathbf{v};\boldsymbol{\lambda}) \quad (2)$$

where $\mathbf{V}_{\mathbf{w}}$ is the set of all possible pronunciation sequences for the word sequence $\mathbf{w}$ and $P(\mathbf{v}|\mathbf{w})$ is obtained from the pronunciation probabilities. The set of possible word sequences in the denominator is determined from a lattice. Thus there is little to consider as the multiple pronunciations can be directly handled in the criterion.

The basic MPE criterion [5] can be expressed as

$$\mathcal{F}_{\texttt{mpe}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\sum_{\mathbf{w}} P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)}) \quad (3)$$

where $\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)})$ is the loss function of word sequence $\mathbf{w}$ against the reference measured at the phone level. When multiple pronunciations are used there are a number of variations that can be used for

the loss function. The most direct approach is to take the minimum between the phone sequences specified by the words.

$$\mathcal{L}(\mathbf{w}, \mathbf{w}_{\mathtt{ref}}^{(r)}) = \min_{\tilde{\mathbf{v}} \in \mathbf{V}_{\mathtt{ref}}^{(r)}, \mathbf{v} \in \mathbf{V}_{\mathbf{w}}} \{\mathcal{L}(\mathbf{v}, \tilde{\mathbf{v}})\} \quad (4)$$

Here, as the number of pronunciations increases so the phone-accuracy must increase. Alternatively the loss function may be expressed at the phone sequence level. Here

$$\mathcal{F}_{\mathtt{mpe}}(\boldsymbol{\lambda}) = \frac{1}{R} \sum_{r=1}^{R} \sum_{\mathbf{w}} \sum_{\mathbf{v} \in \mathbf{V}_{\mathbf{w}}} P(\mathbf{w}|\mathbf{v}) P(\mathbf{v}|\mathbf{O}^{(r)}; \boldsymbol{\lambda}) \mathcal{L}(\mathbf{v}, \mathbf{w}_{\mathtt{ref}}^{(r)}) \quad (5)$$

where

$$\mathcal{L}(\mathbf{v}, \mathbf{w}_{\mathtt{ref}}^{(r)}) = \min_{\tilde{\mathbf{v}} \in \mathbf{V}_{\mathtt{ref}}^{(r)}} \{\mathcal{L}(\mathbf{v}, \tilde{\mathbf{v}})\} \quad (6)$$

This is the *multiple* pronunciation implementation used in this work. In HTK V3.4 [6] used in this work, the homophone issue, the effect of $P(\mathbf{w}|\mathbf{v}) \neq 1$, does not need to be considered as the pronunciation sequence is linked with the word sequence in the lattices. Note in the HTK implementation there is no check that the pronunciations chosen are consistent with pronunciations selected earlier [5]. An alternative to the use of multiple pronunciations is to select the best single reference pronunciation, given the current model-parameters.

$$\hat{\mathbf{v}} = \arg \max_{\mathbf{v} \in \mathbf{V}_{\mathtt{ref}}^{(r)}} \left\{ P(\mathbf{v}|\mathbf{w}_{\mathtt{ref}}^{(r)}) p(\mathbf{O}^{(r)}|\mathbf{v}; \boldsymbol{\lambda}) \right\} \quad (7)$$

The loss function is then simply $\mathcal{L}(\mathbf{v}, \hat{\mathbf{v}})$. When used with equation 5, this is the *single* pronunciation system in this work. The single pronunciation derived in equation 7 may also be used for systems trained with MMI. This is the single pronunciation MMI system.

An alternative way of addressing multiple pronunciations is to use the expected loss over all the pronunciations, rather than taking the minimum. Thus the loss function is modified to

$$\mathcal{L}_{\mathtt{sum}}(\mathbf{v}, \mathbf{w}_{\mathtt{ref}}^{(r)}) = \sum_{\tilde{\mathbf{v}} \in \mathbf{V}_{\mathtt{ref}}^{(r)}} P(\tilde{\mathbf{v}}|\mathbf{w}_{\mathtt{ref}}^{(r)}) \mathcal{L}(\mathbf{v}, \tilde{\mathbf{v}}) \quad (8)$$

Note by definition

$$\mathcal{L}_{\mathtt{sum}}(\mathbf{v}, \mathbf{w}_{\mathtt{ref}}^{(r)}) \geq \mathcal{L}(\mathbf{v}, \mathbf{w}_{\mathtt{ref}}^{(r)}) \geq \mathcal{L}(\mathbf{v}, \hat{\mathbf{v}}) \quad (9)$$

where the differences increase with the number of pronunciations. This summation form of loss function is not examined further in this work. In this work no pronunciation probabilities, $P(\mathbf{v}|\mathbf{w})$, were used during training. When pronunciation probabilities were used in training there was little difference in recognition performance.

Two forms of the V0 model were built, multiple and single reference pronunciations. The front-end processing was the same as that used for the English systems in [7]. Figure 1 shows the difference in the normalised MPE criterion, $1 - R\mathcal{F}_{\mathtt{mpe}}(\boldsymbol{\lambda})/N_{\mathtt{phone}}$ ($N_{\mathtt{phone}}$ is the total number of phones in the R observation sequences), between MPE training with multiple and single reference pronunciations. As expected the multiple reference pronunciations has a larger MPE score than the single one.

Given the differences in the MPE criterion in figure 1, both systems were evaluated. The MPE training for the multiple pronunciations system, V0$_{\mathtt{m}}$, was stopped after 4 iterations as it was found to degrade performance after that. In contrast the single pronunciations system, V0, 6 iterations were performed. Table 2 shows the performance of both MPE and MMI training. For both MMI and MPE

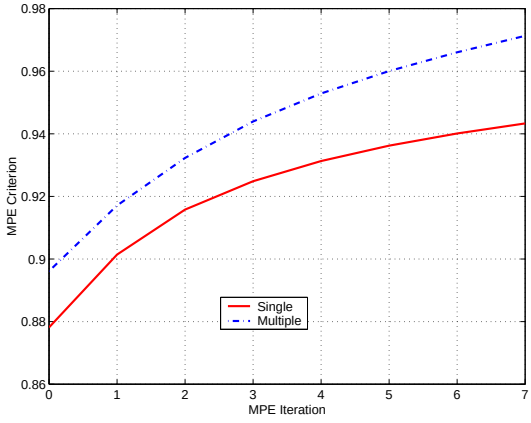

Fig. 1. Training data normalised MPE criterion value for the single pronunciation (V0) and multiple pronunciation (V0$_{\mathtt{m}}$) system.

Criterion	Reference Pron.	Test Set WER (%)	
		bnat06	bcat06
MMI	Multiple	37.5	46.0
	Single	37.6	46.2
MPE	Multiple*	37.2	45.6
	Single†	37.0	45.7

Table 2. %WER of unadapted decoding 16 using the CU segmentation and LM2, † indicates the V0 baseline system, $*$ the V0$_{\mathtt{m}}$ system

training the differences between the single and multiple pronunciation were small. However using the NIST pair-wise significance test (MAPSSWE), the probability that the two MPE systems were the same was only about 0.3 on bnat06. As this is based on the pattern of errors, the two phonetic systems may be useful in combination.

4.2. Phonetic System Combination

As discussed in the previous section, it is interesting to combine the single and multiple pronunciation systems together. The general combination framework described in [7] was used. This is the same as a single segmentation branch of the evaluation style configuration in figure 2. The P1-stage is a fast decoding run with GI models. The P2-stage uses GD models adapted using LSLR and variance scaling using the P1 supervision. The P2-stage generates trigram lattices which are expanded using a 4-gram language model and then rescored in the P3 stage. The P3-stage models are again GD, adapted using 1-best and lattice-MLLR as discussed in [7].

Table 3 shows the combination of the single and multiple pronunciation phonetic systems either with or without pronunciation probabilities. A number of observations can be made. First the use of pronunciation probabilities shows good gains for recognition. For example comparing the V0 phonetic branch with (P2a→P3a) and without (P2a→P3-) pronunciation probabilities, shows that pronunciation probabilities give a 0.4% absolute gain on BN data and 0.9% on BC data. Pronunciation probabilities are therefore used for all subsequent multi-pass/combination experiments. The use of multiple reference pronunciations shows slight gains over the single reference pronunciation. More interestingly cross-adapting the

System		Pron Probs	Test Set WER (%)	
			bnat06	bcat06
P2a	V0	-PP	35.0	43.9
P2b	$V0_m$		35.0	43.9
P2a→P3-	V0	-PP	33.9	43.6
P2a→P3a	V0	+PP	33.5	42.7
P2a→P3c	$V0_m$		32.9	42.2
P2b→P3b	V0	+PP	33.3	42.5
P2b→P3d	$V0_m$		33.1	42.3
P3a+P3c	CNC	—	32.7	42.0
P3a+P3d			32.5	41.8

Table 3. Multi-pass/combination performance of multiple ($V0_m$) and single pronunciation (V0) MPE phonetic system combination with (+PP) and without (-PP) pronunciation probabilities using LM2.

two systems (P2a→P3c) shows gains. However better performance was obtained by using Confusion Network Combination (CNC) to combine the two distinct branches (P3a+P3d) which yields a gain of 0.6% on BN and 0.5% on BC data over the best multiple (P2b→P3d) or single (P2a→P3a) pronunciation system. It is interesting that in other experiments gains were not observed when an MPE system was combined with an MMI system using single pronunciations.

4.3. Phonetic/Graphemic System Combination

Rather than combining two phonetic systems together, it is possible to combine the phonetic system with a graphemic system. This was performed in a cross-adaptation mode in [8]. In this section both cross-adaptation and CNC are examined in the same framework as the previous section. Note, when the graphemic system is used for cross-adaptation the phonetic language model and word-list (LM2) must be used to enable pronunciations to be obtained for all words.

System		Lang. Model	Test Set WER (%)	
			bnat06	bcat06
P2	G0	LM1	36.0	43.4
P2a	V0	LM2	35.0	43.9
P2c	G0	LM2	36.1	43.5
P2 →P3	G0	LM1	34.8	42.7
P2a→P3a	V0	LM2	33.5	42.7
P2c→P3e	V0	LM2	33.1	41.6
P3+P3a			31.8	40.2
P3+P3e	CNC	—	32.1	40.3
P3+P3c			31.6	40.1

Table 4. Multi-pass/combination performance of phonetic (V0) and graphemic (G0) system combination.

Table 4 shows the performance when combining the phonetic (V0) and graphemic (G0) systems. When cross-adaptation is performed for the phonetic system (P2c→P3e) compared to the standard pass 3 performance (P2a→P3a) gains of 0.4% on bnat06 and 1.1% on bcat06 were obtained. This shows the gains of cross-adaptation between graphemic and phonetic (as observed in [8]). It also highlights the difference in performance between the phonetic and graphemic systems on BN and BC data. On BN the phonetic system is better. However on BC the graphemic is comparable. This performance difference is propagated through the cross-adaptation stage. As expected, the implicit modelling of the graphemic system seems to be more robust to the greater variability of BC-style speech. Comparing the graphemic cross-adaptation (P2c→P3e) with the phonetic cross-adaptation from Table 3 (P2a→P3c), shows gains of 0.6% on the BC test set, but a degradation of 0.2% on the BN data.

It is also possible to do CNC between the graphemic P3 branch (P2→P3) and the phonetic P3 branches. Combination of both the straight phonetic system and cross-adaptation system were performed. Though cross-adaptation yielded the best single-branch performance, the best CNC performance was combining the separate branches (P3+P3a). As an additional contrast the phonetic cross-adaptation branch (P2a→P3c from Table 3) was combined with the graphemic system (P3+P3c). This gave additional small gains.

5. PHONETIC SYSTEM REFINEMENT

This section examines the selection and use of additional training data, in particular the use of unsupervised data, and how all this data may be used with a phonetic system. All the acoustic models in this section were state-clustered triphone models with approximately 7K distinct states and an average of 36 Gaussian components per state. Only the single pronunciation discriminative training was used, as this is felt to be slightly more robust to automatically derived pronunciations. This section also discusses schemes for reducing the Out Of Vocabulary (OOV) rates for this basic LMs and the impact of additional LM training data. Note, preliminary experiments using a different number of states for short vowel modelling only yielded small gains, unlike the large gains found in [8], so this was not further investigated.

5.1. Phonetic Acoustic Model

The use of unsupervised training data for Arabic has been shown to yield gains [9]. However the use of unsupervised data for discriminative training when there is a mismatch between the supervised and unsupervised data is less clear [10]. From Table 1 a large amount of unsupervised data (BN03) may be used for building the system. This section examines the selection of training data. Graphemic systems are used as this allows all the data to be used without having to consider how to generate the pronunciations (examined below).

System	#hrs	Test Set WER (%)	
		bnat06	bcat06
G0	100.7	41.8	48.5
G1	1228.0	34.1	40.4
G2	1815.2	33.9	38.6
G3	1024.2	33.5	37.6

Table 5. %WER of unadapted decoding ML 36 components (16 components for G0) , CU segmentation and LM1.

Using the second block of data, the G1 system gave a large reduction in WER compared to the G0 system. In preliminary experiments with this second set of data, the large quantity of unsupervised BN03 data was found to aid performance. This was felt to be because of the small amount of supervised training data. Using all the training data, the G2 system, gave only small gains on the BN data, but 1.8% absolute on the BC data. However removing the unsupervised BN03 data gave further performance gains. This is not surprising as the gains from using unsupervised training are expected to decrease

as the quantity of supervised training data increases, and, as seen here, may degrade performance if the amount of unsupervised data is similar to that of the supervised data.

If all the training data used for the G3 graphemic system is to be used for the phonetic system then pronunciations are required for all words. In contrast to the previous systems with mainly lightly supervised and unsupervised data, there was large amounts of supervised data (655 hours) with only the standard orthographic transcriptions. If the procedure described in the section 2 is used then about 216 hours of data (about a third of the data) has at least one word in a segment that does not have a pronunciation. There are a number of approaches that have been adopted in the literature to deal with this. As previously discussed, one approach is to use the training data in a lightly/unsupervised fashion [9]. Alternatively it is possible to back-off to the graphemic pronunciation and build a combined system [8]. Finally in [2] a series of expert rules are used to derive pronunciations.

In this work rather than using expert derived rules a series of rules were automatically generated from a 250K Buckwalter derived phonetic dictionary. Though this derives many of the standard expert rules, it ensures that the rules were consistent with pronunciations from Buckwalter. The pronunciations were derived in a "right-associative" fashion and the start (_S) and end (_E) of word pronunciations were kept distinct from standard variations (_V) (this also allows inter-word silence to be correctly added to the pronunciations). The pronunciation and derived rules for كتاب are

```
ktAb        /k/ /i/ /t/ /A/ /b/
k_S         /k/
t_V         /i/ /t/
A_V         /A/
b_E         /b/
```

This yielded 889 derived pronunciations which were guaranteed to yield a pronunciation for each word. The vast majority resulted from *nunation* at the end of words.

System	#hrs	Test Set WER (%)	
		bnat06	bcat06
G1	1228.0	30.8	37.2
G3	1024.2	29.2	33.3
V1	1220.2	30.4	38.0
V2	800.7	29.1	34.9
V3	1015.1	28.5	33.9

Table 6. %WER of unadapted decoding MPE 36 components, CU segmentation, and LM2.

Table 6 shows the performance of MPE trained graphemic and phonetic systems. The V1 and G1 systems are comparable. Again the general trend of the graphemic system performing better on the BC data is observed, whereas the phonetic system performs better on the BN data. The V2 system was built using all the training data segments for which pronunciations could be obtained using the standard approach. Though reductions in WER were obtained they were less than for the graphemic system (G3). The automatic pronunciation scheme was then used. This gave gains similar to those seen for the graphemic system, showing the efficacy of this simple approach.

5.2. Language Model Development

The initial language models used in the previous sections, LM1 a 65K graphemic word-list and LM2 using a 65K phonetic word-list, were constructed using 422 million words of LM training data. The graphemic word-list was obtained using weighted frequencies from the training data. The phonetic word-list consisted of the top 65K words, ranked by frequency, that phonetic pronunciations could be obtained using Buckwalter.

Language Model	Vocab Size	OOV Rate (%)	
		bnat06	bcat06
LM1	65K	4.9	6.4
—	130K	2.6	3.8
—	250K	1.6	2.6
LM2	65K	5.3	8.0
—	130K	3.4	6.1
—	210K	2.7	5.4

Table 7. OOV rates for Various Language Model Word-Lists, the 210K phonetic word-list is the subset of the 250K graphemic word-list for which pronunciations could be generated.

Table 7 shows the impact of increasing the vocabulary size with these initial word-lists. As observed in other work [1, 8] increasing the vocabulary size from 65K dramatically reduces the OOV rate. As expected the OOV rate for the phonetic word-list is higher. For the 210K phonetic word-list, which is the subset of the 250K graphemic word-list that Buckwalter could generate pronunciations for, the difference in OOV rate was 2.8% for BC. Irrespective of how large the word-lists are made this difference can only grow.

Language Model	Vocab Size	Pron Source	OOV Rate (%)	
			bnat06	bcat06
LM3	350K	—	1.1	2.0
—	258K	buck	2.5	5.0
—	+1670	map	2.4	4.9
—	+734	auto	2.0	3.5
LM4	+50	hand	1.9	3.2

Table 8. OOV rates for Various Language Model Word-Lists.

As additional data was made available for the new LMs, a new word-list was generated again using weighted word frequencies. The total available training data for the LM3 and LM4 language models was 1013 million words. A 350K graphemic word-list was constructed. Of this 350K word-list, approximately 258K were able to be directly handled by Buckwalter (buck). On the BC data this resulted in an OOV rate about 3.0% absolute greater than the graphemic one. These pronunciations were then augmented in three stages, see Table 8. First, since *alif* and other variants will be mapped to simple forms, all words that only differ by the form of the alif, for example, from a word that Buckwalter could derive a pronunciation were included with the mapped pronunciation (map). Second automatic pronunciations from section 5.1 that occurred more than five times and in the 350K word-list, were then added (auto). Finally the most common 50 words that were still missing were then added by hand (hand). This reduced the OOV rate to about 1.2% worse than the graphemic word-list on BC data and 0.8% worse on BN data.

Table 9 shows the WER using the MPE trained G3 system. As well as having lower OOV rates, LM3 and LM4 yielded significantly lower WERs. Note, the difference in performance between LM3 and LM4 was 0.1% absolute or less, consistent with the differences obtained using LM1 and LM2.

Language Model	Vocab Size	Test Set WER (%)	
		bnat06	bcat06
LM1	65K	28.8	33.0
LM2	65K	29.2	33.3
—	130K	28.4	32.5
—	210K	28.0	32.2
LM3	350K	26.6	30.5
LM4	260K	26.6	30.6

Table 9. %WER of unadapted MPE G3 models, CU segmentation.

6. EVALUATION SYSTEM

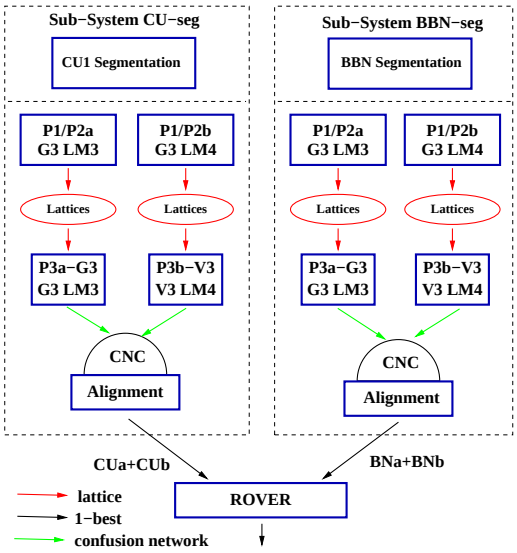

Fig. 2. Evaluation System Configuration.

The phonetic V3 and graphemic G3 acoustic models were used with the LM4 and LM3 language models in a full evaluation framework. This is shown in figure 2. Two segmentations were used. The first CU1 is a revised version of the CU segmentation used earlier (this addressed an issue of deleting significant data from one of the shows). The second segmentation was supplied by BBN. This is similar to the CUED GALE'07 Arabic STT system and is based on the dual segmentations RT04f English evaluation system [7]. The P1/P2 branches were run using the graphemic models, due to the speed/ software issues of using large word-lists with the large number of phonetic pronunciations. The same general trends as the development systems can be observed in the full system results, see Table 10. The phonetic system is better on BN data, the graphemic system is better on BC data, even though cross-adaptation yields gains for the phonetic system. Cross segmentation combination yields small, consistent, gains. As a contrast, combining the single pronunciation, V3, system with the multiple pronunciation, $V3_{\mathtt{m}}$, system using the CU1 segmentation gave 19.2% on BN and 26.3% on BC data. The graphemic and phonetic combination (CUa+CUb) is again better, though there are still gains combining the two phonetic systems.

7. CONCLUSIONS

This paper has described the development of a phonetic system for Arabic speech recognition. A number of issues involved with building these systems have been discussed. First the impact of the large number of pronunciations has on discriminative training, and how the use of multiple and single pronunciation systems may be successfully combined together. The interaction of phonetic and graphemic systems is then described, where it is found that CNC rather than cross-adaptation is better. Also, graphemic systems perform at least as well as phonetic systems on BC data. Finally a simple approach that allows all the acoustic training data to be used is described. Finally the performance in a multi-pass combination framework similar to the one used for the GALE'07 evaluation is given.

System			Seg	Test Set WER (%)	
				bnat06	bcat06
	P2a-G3	LM3	CU1	21.1	27.6
			BBN	20.4	27.3
	P2b-G3	LM4	CU1	21.3	27.8
			BBN	20.6	27.6
CUa	P3a-G3	LM3	CU1	20.4	27.2
BNa			BBN	19.9	26.9
CUb	P3b-V3	LM4	CU1	20.0	27.3
BNb			BBN	19.1	27.1
CUa + CUb		CNC	CU1	18.7	25.3
BNa + BNb			BBN	17.9	25.2
CUa+CUb⊕BNa+BNb			—	17.5	25.0

Table 10. Multi-pass/combination performance using the CU1 and BBN segmentations, "⊕" indicates ROVER combination, "+" CNC.

8. REFERENCES

[1] M. Afify, L. Nguyen, B. Xiang, S. Abdou, and J. Makhoul, "Recent progress in Arabic broadcast news transcription at BBN," in *Proc. InterSpeech*, 2005.

[2] A. Messaoudi, J.-L. Gauvain, and L. Lamel, "Arabic transcription using a one million word vocalized vocabulary," in *Proc. ICASSP*, 2006.

[3] M. Afify, R. Sarikaya, H-K. Kuo, L. Besacier, and Y. Gao, "On the use of morphological analysis for dialectal Arabic speech recognition," in *Proc. InterSpeech*, 2007.

[4] H. Y. Chan and P. C. Woodland, "Improving broadcast news transcription by lightly supervised discriminative training," in *Proc. ICASSP*, 2004.

[5] D. Povey, *Discriminative Training for Large Vocabulary Speech Recognition*, Ph.D. thesis, Cambridge Uni., 2004.

[6] S.J. Young et. al., *The HTK Book (for HTK Version 3.4)*, University of Cambridge, Dec. 2006.

[7] M.J.F. Gales et. al., "Progress in the CU-HTK broadcast news transcription system," *IEEE Transactions Speech and Audio Processing*, September 2006.

[8] H Soltau, G Saon, B Kingsbury, H-K Kuo, L Mangu, D Povey, and G Zweig, "The IBM 2006 GALE Arabic ASR system," in *Proc. ICASSP*, 2007.

[9] J. Ma, S. Matsoukas, O. Kimball, and R. Schwartz, "Unsupervised training on large amount of broadcast news data," in *Proc. ICASSP 2006*, May, 2006, pp. 1056–1059.

[10] L. Wang, M.J.F. Gales, and Woodland P.C., "Unsupervised training for Mandarin broadcast news and conversation transcription," in *Proc. ICASSP*, 2007.