Word Boundary Modelling and Full Covariance Gaussians for Arabic Speech-to-Text Systems

F. Diehl, M.J.F. Gales, X. Liu, M. Tomalin, & P.C. Woodland

Engineering Department, Cambridge University, Trumpington St., Cambridge, CB2 1PZ U.K.
{fd257, mjfg, xl207, mt126, pcw}@eng.cam.ac.uk

Abstract

This paper describes recent improvements to the Cambridge Arabic Large Vocabulary Continuous Speech Recognition (LVCSR) Speech-to-Text (STT) system. It is shown that word-boundary context markers provide a powerful method to enhance graphemic systems by implicit phonetic information, improving the modelling capability of graphemic systems. In addition, a robust technique for full covariance Gaussian modelling in the Minimum Phone Error (MPE) training framework is introduced. This reduces the full covariance training to a diagonal covariance training problem, thereby solving related robustness problems. The full system results show that the combined use of these and other techniques within a multi-branch combination framework reduces the Word Error Rate (WER) of the complete system by up to 5.9% relative.

Index Terms: Arabic, STT, Context, Covariances

1. Introduction

In recent years, Arabic-based Automatic Speech Recognition (ASR) systems have improved considerably [1, 2, 3]. The Arabic language is a member of the Semitic language family and it poses many problems for ASR systems since it is morphologically complex and its dialects differ significantly. In addition, Modern Standard Arabic (MSA) is usually written without short vowels. Consequently, graphemic and phonetic Arabic STT systems are widely used: the former use unvowelised transcriptions, while the later use vowelised transcriptions. The complementary phonetic and graphemic output is combined in a system combination framework [2].

This paper describes recent improvements to the Cambridge Arabic STT systems. The techniques discussed improve both the individual branches and the gains obtained by system combination. Section 2 describes the use of word-boundary context models. Such models distinguish between phonemes which appear at different positions in a word (i.e., word-initial, word-medial, word-final). The emphasis falls primarily on graphemic systems, since word-boundary information can help to alleviate the modelling limitations caused by the absence of short vowel information. In section 3, the use of full covariance matrices for acoustic modelling is discussed. The complementarity of the full covariance models compared to other acoustic models is explored, as is their use in a system combination framework. An effective implementation of full covariance models is proposed and its advantages for discriminative acoustic model training are highlighted. Section 4 describes the experimental setup, while section 5 gives detailed results and analysis for the Cambridge Arabic STT systems.

2. Word-boundary context modelling

In MSA, the short vowels (*fatha* /a/, *kasra* /i/, and *damma* /u/) as well as the corresponding word-final nunations (*fathatan* /an/, *kasratan* /in/, and *dammatan* /un/) are not marked in written text. For phonetic systems, this complicates the design of the dictionary since the short vowels present in the spoken utterance need to be inferred. In the work discussed here, vowelised forms are obtained using the Buckwalter Morphological Analyser (version 2.0; henceforth 'Buckwalter').[1] However, a morphological analyser such as Buckwalter cannot produce analyses for all the words in the ASR dictionary. For instance, for a 350k word dictionary, Buckwalter provided only 260k vowelised word forms [4]. A further problem is caused by the resulting dictionary size: although Buckwalter cannot produce vowelised word forms for every entry, it typically generates 4.3 vowelised word forms per graphemic word form [4], and this creates a significant processing overhead during decoding and discriminative training. Graphemic systems overcome these difficulties since they rely on a one-to-one mapping between graphemes and phonemes, thereby simplifying the dictionary generation task. The drawback is that acoustic models used in graphemic systems only model the short vowels *implicitly*. Modelling a short vowel together with its associated consonants results in less precise acoustic units [4].

Consequently, graphemic models are faster in decoding and more robust to changes in the acoustic environment and speaking styles, but they are outperformed by phonetic models [4]. Some of the shortcomings of graphemic models can be overcome by using word-boundary context information [5]. This information is added to the system by marking the phonetic units of a word in the dictionary as word-initial, *I_*, word-medial, *M_*, and word-final, *F_*. For the word `ktAb`[2], the transcription changes as follows:

```
ktAb     /k/ /t/ /A/ /b/
→        /I_k/ /M_t/ /M_A/ /F_b
```

Word-boundary markers distinguish phonemic pronunciation variants due to word position. They also provide indirect information about short vowels and nunation. As mentioned above, nunations may appear at the end of a noun or an adjective but are commonly not marked in written texts. Marking a given word-final acoustic unit by *F_* indicates a possible nunation which distinguishes it from a word-medial version of the same unit. In this way, word-boundary markers provide graphemic systems with more finely granulated information. For phonetic systems, the modelling advantages are smaller since the short vowels are explicitly marked in the dictionary.

[1] Available from the Linguistic Data Consortium (LDC), http://www.ldc.upenn.edu.

[2] Buckwalter transliteration.

The setup described in section 4.1 was used for all tests discussed in this section. Traditionally, one decision tree is built for each phonetic base unit and state position. However, three times more base units are needed when word-boundaries are marked, which leads to an over fragmentation of the states space. Therefore, the word-initial, word-medial, and word-final phone variants of a given phoneme were clustered within one decision tree for this phoneme. Consequently, 'central phone questions' were introduced, in addition to questions asking for one of the three possible word-boundary context variants. Polythetic questions asking for a particular phoneme in one of the three possible contexts were also used.[3] Table 1 contrasts the use of word-boundary information for a graphemic ('Gra') and a phonetic ('Pho') system. The subscript $_{context}$ indicates the use of word-boundary markers.

System	Testset		
	d07	**d08**	**d09sub**
Gra	21.2	24.5	27.0
Gra$_{context}$	18.5	21.6	24.2
Pho	16.3	19.2	23.0
Pho$_{context}$	15.9	19.0	22.2

Table 1: Contrast of graphemic and phonetic models with and without word-boundary information. ML trained acoustic models, 12k tied states, unadapted decoding results, WER in%.

In the phonetic case, reductions in WER of 1.0-3.5% relative are observed. This confirms the hypothesis that position-dependent pronunciation variations for acoustic units can be captured by word-boundary markers. In the graphemic case, the WER reductions of 10.4-12.7% relative are much larger than in the phonetic case. This confirms the hypothesis that short vowel information is at least partly recovered by the word-boundary markers. A similar pattern of an approximate reduction in WER of 10% relative was also found in case of a MADA morpheme based graphemic system.

3. Full Covariance Modelling

For reasons of efficiency and robustness, STT systems usually use diagonal covariances for the individual Gaussian components. Spectral (intra-frame) correlations are taken into account by applying a mixture of Gaussians for each state. This modelling approach has the virtue of being efficient in terms of memory and CPU consumption, but it lacks the modelling capability of applying full covariance matrices. A well-known problem with full covariance models is the large number of parameters which needs to be estimated. For a 39-dimensional feature vector (as used for this work), the number of model parameters required compared to the diagonal case is increased by a factor of 10 when keeping the number of Gaussian components fixed. For model training, the related data sparsity problem is a central issue. Thus, for Maximum Likelihood (ML) training, the covariances are smoothed by the diagonal elements. When calculating the diagonal elements of the covariance matrices, their component occupation counts γ_{jm} (for the j^{th} state and the m^{th} component) are increased by a prior count τ to

$$\gamma_{jm}^{'} = \gamma_{jm} + \tau \tag{1}$$

[3]For Chinese ASR systems, polythetic questions give slight performance gains.

whereas the occupation counts for the off-diagonal elements are kept unchanged. In this work a value of $\tau = 100$ resulted in a stable reestimation procedure.

System		Testset		
Cov	**#Comp**	**d07**	**d08**	**d09sub**
diag	36	21.3	24.8	27.1
diag	72	20.1	23.3	25.7
diag	144	19.5	22.2	24.8
full	4	23.3	26.1	28.8
full	8	20.9	24.1	26.3
full	16	19.6	22.7	25.5
full	36	18.5	21.6	23.9

Table 2: Contrast of diagonal versus full-covariance modelling for different number of Gaussian components. ML trained acoustic models, 9k tied states, unadapted decoding results, WER in%.

Table 2 compares three ML-trained systems with diagonal covariance modelling with four corresponding full covariance systems. In the diagonal and the full covariance cases, the system performance is increased when the number of Gaussian components is incremented. However, the decrements in WER tend to be larger in the full covariance case and the best system performance is obtained for the full covariance system with 36 components per state. This confirms the potential of the proposed smoothing procedure. Though the number of model parameters in the 36 component full covariance case is increased by more than a factor of 2.5 compared to the 144 component diagonal covariance case, the full covariance system copes well with data sparsity. It outperforms the best diagonal covariance system by 0.6-1.0% WER absolute.

It is well known that, in the case of MPE training, the data sparsity issue is even more critical than in the ML case. To cope with this problem, I-smoothing was introduced in [6]. I-smoothing can be regarded as the use of a prior over the parameters of a Gaussian, with the prior being based on the statistics of a more robust estimation procedure. For the CUED MPE training, I-smoothing is a two stage process where the MPE statistics are I-smoothed by MMI statistics which are again I-smoothed by corresponding ML statistics. In the full covariance case, this procedure has the disadvantage that, in addition to the MPE statistics, full covariance MMI and ML statistics also need to be estimated and stored. This requires considerable CPU time, as well as disc space to store the intermediate statistics. In case of full covariance models and a large amounts of acoustic training data, this procedure is impractical.

To overcome these problems, MPE training in transformed feature spaces can be implemented. With the full covariance matrix $\mathbf{\Sigma}_{jm}$, an observation $\boldsymbol{o}$ is modelled at the component level by a Gaussian as $\boldsymbol{o} \sim \mathcal{N}(\boldsymbol{\mu}_{jm}, \mathbf{\Sigma}_{jm})$. When introducing the decorrelating feature transform

$$\boldsymbol{o}^{'} = \boldsymbol{A}_{jm}^{T}\boldsymbol{o} \tag{2}$$

which diagonalises the covariance matrix $\mathbf{\Sigma}_{jm}$ to

$$\mathbf{\Sigma}_{jm}^{'} = \boldsymbol{A}_{jm}^{T}\mathbf{\Sigma}_{jm}\boldsymbol{A}_{jm} \tag{3}$$

the transformed feature space observation $\boldsymbol{o}^{'}$ is modelled by

$$\boldsymbol{o}^{'} \sim \mathcal{N}(\boldsymbol{\mu}_{jm}^{'}, \mathbf{\Sigma}_{jm}^{'}) \tag{4}$$

where $\boldsymbol{\mu}'_{jm} = \boldsymbol{A}^T_{jm}\boldsymbol{\mu}_{jm}$ is the transformed mean. Principle Component Analysis (PCA) is used for feature decorrelation. The elements of the diagonal covariance matrix $\boldsymbol{\Sigma}'_{jm}$ in the transformed space are therefore the eigenvalues of the original covariance matrix $\boldsymbol{\Sigma}_{jm}$. Parameter reestimation, including I-smoothing, is then performed in the transformed, decorrelated domain. Consequently, MPE training involves a few additional steps. The starting point is an ML-trained full covariance model where for each of its Gaussian component PCA is applied and the transformations $\boldsymbol{A}_{jm}$ are calculated. Next, the transformations are applied to the model and the observations $\boldsymbol{o}$, and MPE training is carried out in the decorrelated parameter space. Finally, the reestimated model parameters are transformed back to the original parameter space which gives the resulting full covariance model. The proposed method of implementing an MPE-trained full covariance system solves two problems. In terms of memory and CPU demand, it is comparable to the training of a standard diagonal covariance model. In addition, the data sparsity problem is solved as standard I-soothing techniques are applied. These advantages come with the drawback that only a fraction of all model parameters are trained discriminatively.

Table 3 compares three MPE-trained diagonal covariance systems with a MPE-trained 36-component full covariance system. Increasing the number of components for the diagonal covariance systems results in consistent performance gains. Comparing the best 144-component diagonal covariance system with the 36-component full covariance system, similar performance is observed. However, in the full covariance case, the number of MPE-trained parameters constitutes only a fourth of the MPE-trained parameters in the diagonal covariance case. This confirms the potential of the proposed approach for full covariance modelling.

System		Testset		
Cov	**#Comp**	**d07**	**d08**	**d09sub**
diag	36	15.7	18.3	21.6
diag	72	15.4	17.8	21.2
diag	144	15.0	17.5	20.6
full	36	15.1	17.4	20.4

Table 3: Contrast of diagonal versus full covariance modelling for different number of Gaussian components. MPE-trained acoustic models, unadapted decoding results, WER in%.

4. System description

4.1. Development Systems

For system development, ML- and MPE-trained acoustic models were built. All systems applied PLP-based front-ends with a 39-dimensional feature vector after a HLDA transform. Cross-word decision-tree state-clustered triphones were built using approximately 1850 hours of acoustic training data. All systems referred to in section 2 'word-boundary context modelling' used 12k tied states, while the systems refered to in section 3 'full covariance modelling' use 9k tied states. The larger number of tied states used in the former systems account for the increased logical state space (by a factor of $3^3 = 27$) due to 3 times more phonetic base units in the case of the word-boundary context models. It was found that increasing the number of tied states from 9k to 12k typically reduced the absolute WER by 0.5%.

For development purposes, an unadapted decoding configuration was chosen. For the graphemic systems, a two-pass decoding setup was used which involves lattice-generation with a bi-gram Language Model (LM) followed by lattice rescoring with a tri-gram LM. All LMs and associated dictionaries were based on a 350k wordlist and 1.2G tokens LM training material. For the phonetic systems, acoustic rescoring was used rescoring the lattices of the graphemic system which were not applying word-boundary markers. For phonetic dictionary generation, Buckwalter provided 310k vowelised pronunciations for the 350k words. For the remaining words, pronunciations based on the G2P-method described in [7] were used. On the average this gave 6.7 pronunciations per word. The system performance was evaluated on three development test sets: d07 (2.58 hours), d08 (3.04 hours), and d09sub (2.93 hours) for which the OOV rates were in the range 1-2%.[4]

4.2. Evaluation Systems

The final assessment of the word-boundary context markers and the proposed full covariance modelling was performed using the CUED ASR system developed for the GALE phase 5 system evaluation.[5] This system consists of five branches providing five hypothesises which are combined by ROVER [8]. All branches use the same multi-pass adaptation framework as described in [2], but apply branch specific front-ends, LMs, and acoustic models. For the front-ends, PLP features and TANDEM connected [9] PLP-MLP (Multi-Layer Perceptron), features [10] are used. The MLP applies phonetic targets providing implicit short vowel information to graphemic systems [11]. In addition to standard word-based systems, MADA morpheme-based systems [12] are also used. To further increase the diversity between the system branches, this is combined with graphemic and phonetic modelling, and MPE and boosted MMI (BMMI) [13] training.

The three-stage decoding process consists of a P1-stage which is a fast decoding run with gender-independent (GI) PLP graphemic models. The P2-stage uses speaker-adapted gender-dependent (GD) graphemic models based on the P1 supervision. It generates trigram lattices which are expanded using a 4-gram LM. This is followed by LM rescoring applying a class-based LM and a Neural Network LM (NN-LM) which were both interpolated with the 4-gram [2]. The training material, wordlists, and build procedure used for the n-gram LMs and class-based LMs (1000 classes) are equivalent to the ones described in section 4.1. The NN-LMs follow the build procedure described in [12].

The P3-stage serves for acoustic rescoring and applies different GD models which were adapted using 1-best CMLLR and lattice-MLLR as discussed in [14]. Confusion network decoding was then performed on this output and ROVER [8] was used for system combination. For PLP-system adaptation, full CMLLR and lattice-MLLR transforms were used. For the PLP+MLP-systems, block diagonal transforms were deployed, one block for the PLP features, and one for the MLP features. To reduce the computational cost for the full covariance modelling, the estimation of the linear transformations was based on the leading diagonal of the covariance statistics.

For the P1+P2 stage, three different acoustic models were used. The 'G1' graphemic word-based model, and the two 'G2' and 'G3' graphemic morpheme-based models. G2 and G3 differ in the MADA version used. G2 uses MADA version 2.3, while G3 uses MADA version 1.8. All three models feature 9k tied

[4] d07, d08, d09sub, as well as d10c, and d10d denote the dev07, dev08, dev09sub, dev10c, and dev10d, respectively, development testsets used for system development within the GALE project.

[5] See: http://projects.ldc.upenn.edu/gale/

states, MPE acoustic training, diagonal covariances, PLP features and a dictionary without word-boundary context markers.

For the P3 stage, two more morpheme-based models and two more word-based models were used. The 'V1' phonetic model ('V' for vowelised) uses MADA version 2.3, the PLP+MLP frontend, 12k states, word-boundary context markers and BMMI acoustic training. The 'V2' is similar to V1 though using MADA version 1.8, 9k states, and no word-boundary markers. The V2 model is the best acoustic model developed for the GALE phase 4 evaluation. The remaining models are word-based and MPE-trained. The graphemic G1 model applies the PLP+MLP frontend, 12k states and word-boundary markers, whereas the phonetic V3 model is based upon the standard PLP front-end. Model V3 features 9k states without word-boundary markers and is the only model with full covariance matrices.

For the evaluation systems, the development test sets d10c (6.38 hours), and d10d (18.46 hours) were used along with d09sub. d10c is comparable to d09sub in complexity. By contrast, d10d features acoustic data which is more challenging. This is indicated by the 'd' extension which stands for '**d**ifficult'.

5. Experiments and Results

Table 4 shows the individual branch results and the ROVER outcome for the systems. When comparing the best individual phase 5 and phase 4 systems (i.e. V1 versus V2) improvements of 0.4-0.8% in absolute WER are observed. Most gains are due to the word-boundary context modelling incorporated into the phase 5 model. The graphemic word-based G1 system performs nearly as well as the phonetic morpheme-based V2 system. This emphasises the importance of word-boundary context modelling and the use of MLP features with phonetic targets in grahemic model. Both techniques help to overcome the graphemic system's implicit modelling of short vowels.

Comparing the V1 and V2 branch with the V3 branch, it is clear that the full covariance modelling can not compensate for the lack of TANDEM PLP+MLP features, morphological decomposition, or the use of MPE instead of BMMI for model training. However, even a simple full covariance model such as V3 is more complementary in system combination than any other word-based phonetic system.

Finally, comparing the ROVER result for the GALE phase 5 evaluation with the GALE phase 4 evaluation, a reduction of 0.6-0.9% WER absolute is observed. These improvements are mainly due to the improved acoustic modelling which comes from word-boundary markers and the full covariance modelling in the V3 system. However, other factors include the increased state space for the word-boundary context models, dedicated dictionaries and LMs developed for the phase 5 evaluation, and approximately 300 hours of additional acoustic training data.

6. Conclusion

This paper has described recent improvements to the Cambridge Arabic STT systems. It has been shown that word-boundary context markers provide an efficient way to incorporate phonetic information into graphemic systems. This results in graphemic systems which are closer in performance to phonetic systems, but which feature the reduced complexity associated with graphemic system. Further, a novel approach for full covariance MPE training was introduced. It was shown that it solves the robustness problem of discriminative full covariance modelling. Similar performance to a diagonal covariance system with four times more discriminatively trained model parameters was obtained.

System	Configuration					Testset		
	Ctx	Full	MLP	Mada	BMMI	d09sub	d10c	d10d
V1	√	-	√	√	√	14.8	13.8	25.0
V2	-	-	√	√	√	15.4	14.5	25.8
G1	√	-	√	-	-	15.5	14.6	25.9
V3	-	√	-	-	-	16.4	15.7	27.3
G2	-	-	-	√	-	17.3	16.3	28.2
ROVER GALE phase 5 (V1⊕V2⊕G3⊕G2⊕V4)						13.7	12.8	23.8
ROVER GALE phase 4						14.3	13.6	24.7

Table 4: Individual branch results and ROVER results from the GALE phase 5 and phase 4 system evaluation, WER in %. Nomenclature: 'Ctx' → use of word boundary markers and 12k states; 'Full' → use of full covariances; 'MLP' → use of TANDEM PLP+MLP features; 'Mada' → use of MADA morphological decomposition; 'BMMI' → use of BMMI trained models. If no 'tick' is present, the standard configuration applies: no word-boundary markers, 9k states, diagonal covariances, PLP features, word based tokens and MPE trained models.

Finally, the proposed techniques were investigated within a state-of-the-art 5-way LVCSR system. The combined use of these techniques give WER reductions of 3.6-5.9% relative.

7. References

[1] G. Saon, H. Soltau, U. Chaudhari, S. Chu, B. Kingsbury, H. Kuo, L. Mangu, and D. Povey, "The IBM 2008 GALE Arabic Speech Transcription System," in *Proc. of ICASSP*, 2010.

[2] M. Tomalin, F. Diehl, M.J.F. Gales, J. Park, and P.C. Woodland, "Recent improvements to the Cambridge Arabic Speech-to-Text Systems," in *Proc. of ICASSP*, 2010.

[3] T. Hg, K. Nguyen, and R. Zbib, "Improved Morphological Decomposition for Arabic Broadcast News Transcription," in *Proc. of ICASSP*, 2009.

[4] F. Diehl, M.J.F. Gales, M. Tomalin, and P.C. Woodland, "Phonetic pronunciations for Arabic speech-to-text systems," in *Proc. of ICASSP*, 2008.

[5] D. Vergyri, K. Kirchhoff, R. Gadde, A. Stolcke, and J. Zheng, "Development of a conversational telephone speech recognizer for Levantine Arabic," in *Proc. of Interspeech*, 2005.

[6] D. Povey, "Discriminative Training for Large Vocabulary Speech Recognition," *Ph.D Thesis, University of Cambridge*, 2004.

[7] M. Bisani and H. Ney, "Joint-Sequence Models for Grapheme-to-Phoneme Conversion," *Speech Communication*, vol. 50, pp. 434–451, May 2008.

[8] J.G. Fiscus, "A Post-Processing System to Yield Reduced Word Error Rates: Recogniser Output Voting Error Reduction (ROVER)," in *Proc. of ASRU*, 2006.

[9] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in *Proc. of ICASSP*, 2000.

[10] J. Park, F. Diehl, M.J.F. Gales, M. Tomalin, and P.C. Woodland, "Efficient Generation and Use of MLP Features for Arabic Speech Recognition," in *Proc. of Interspeech*, 2009.

[11] J. Park, F. Diehl, M.J.F. Gales, M. Tomalin, and P.C. Woodland, "Training and adapting MLP features for Arabic speech recognition," in *Proc. of ICASSP*, 2009, pp. 4461–4464.

[12] F. Diehl, M.J.F. Gales, M. Tomalin, and P.C. Woodland, "Morphological Analysis and Decomposition for Arabic Speech-to-Text Systems," in *Proc. of InterSpeech*, 2009.

[13] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for model and feature-space discriminative training," in *Proc. of ICASSP*, 2008.

[14] M.J.F Gales, F. Diehl, C.K. Raut, M. Tomalin, P.C. Woodland, and K. Yu, "Development of a phonetic system for large vocabulary Arabic speech recognition," in *Proc. of ASRU*, 2007.