THE IBM 2008 GALE ARABIC SPEECH TRANSCRIPTION SYSTEM

George Saon, Hagen Soltau, Upendra Chaudhari, Stephen Chu, Brian Kingsbury, Hong-Kwang Kuo, Lidia Mangu and Daniel Povey[†*]

IBM T. J. Watson Research Center, Yorktown Heights, NY, USA

[†]Microsoft Research, Redmond, WA, USA

ABSTRACT

This paper describes the Arabic broadcast transcription system fielded by IBM in the GALE Phase 3.5 machine translation evaluation. Key advances compared to our Phase 2.5 system include improved discriminative training, the use of Subspace Gaussian Mixture Models (SGMM), neural network acoustic features, variable frame rate decoding, training data partitioning experiments, unpruned n-gram language models and neural network language models. These advances were instrumental in achieving a word error rate of 8.9% on the evaluation test set.

Index Terms— Speech recognition

1. INTRODUCTION

Under the auspices of DARPA's Global Autonomous Language Exploitation (GALE) project, a tremendous amount of work was done by the speech research community toward improving speech recognition performance. The goal of the GALE program is to make foreign language (Arabic and Chinese) speech and text accessible to English-only speakers, particularly in military settings. A core component of GALE is automatic speech recognition, and research in this area spans multiple fields ranging from traditional speech recognition to improving the interfaces between speech recognition, machine translation and information extraction. In this paper, we focus on Arabic broadcast transcription, although many of the techniques we describe have been successfully applied to our Chinese and English broadcast transcription systems.

2. SYSTEM OVERVIEW

The IBM system for Arabic Broadcast News consists of a variety of models which differ in the following aspects: (i) acoustic features (PLP cepstra, MFCCs, and neural network features); (ii) vowelization (unvowelized and vowelized acoustic models); (iii) Gaussian allocation/estimation (standard Gaussian Mixture Models and Subspace Gaussian Mixture Models [1]); (iv) variable and fixed frame rate decoding; and (v) training data selection (models trained on broadcast news and broadcast conversations versus models trained only on broadcast news). The unvowelized models are strictly graphemic whereas the vowelized ones explicitly model diacritics and short vowels. All acoustic models use penta-phone crossword acoustic context and a combination of feature space and model space discriminative training using either MPE [2] or the boosted MMI criterion [3] with some modifications described in [4].

[*]This work was performed while the author was with IBM Research.

2.1. Acoustic training and test data

Arabic broadcast news and conversations are highly variable, coming from many different broadcasters and containing a mixture of Modern Standard Arabic (MSA) and dialectical Arabic. We used the following corpora for acoustic model training in experiments presented here:

- 85 hours of FBIS and TDT-4 audio with transcripts provided by BBN,
- 1500 hours of transcribed GALE data provided by the LDC for the GALE Phase 3.5 (P3.5) evaluation

We report results on a variety of test sets: DEV'07 (2.5 hours), DEV'08 (3 hours) and the unsequestered portion of EVAL'08 (3 hours).

2.2. P3.5 System Architecture

The system architecture has a series of recognition passes, speaker adaptation passes, and language model rescoring and system combination passes. More precisely, the operation of our system follows the steps:

(1) clustering of the speech segments into speaker clusters,
(2) speaker independent (SI) decoding of the speech segments,
(3) estimation of VTLN warp factors based on the SI output,
(4a) estimation of fMLLR [5] and MLLR for the unvowelized acoustic models based on the SI output,
(4b) estimation of fMLLR and MLLR for the vowelized acoustic models based on the SI output,
(5a) speaker-adapted decoding with unvowelized models (U),
(5b) speaker-adapted decoding with vowelized models (V),
(6a) estimation of fMLLR and MLLR for U based on V output,
(6b) estimation of fMLLR and MLLR for U model trained on BN only based on V output,
(6c) estimation of fMLLR and MLLR for U model with MFCC features based on V output,
(6d) estimation of fMLLR and MLLR for V based on U output,
(6e) estimation of fMLLR and MLLR for V SGMM model based on U output,
(6f) estimation of fMLLR and MLLR for V model with NN features based on U output,
(7a) variable frame rate re-decoding with the U model
(7b) variable frame rate re-decoding with the V model
(7c) variable frame rate decoding with the V-SGMM model
(7d) variable frame rate decoding with the V-NN model
(7e) fixed frame rate decoding with the U-BN model

Step	Decoding pass	DEV'07	DEV'08	EVAL'08
(2)	SI	24.5%	27.2%	23.3%
(5a)	U	11.6%	12.8%	11.1%
(5b)	V	11.0%	12.5%	10.5%
(7a)	UxV (vfr)	10.6%	11.5%	9.9%
(7b)	VxU (vfr)	10.0%	11.5%	9.8%
(7c)	V-SGMMxU (vfr)	9.5%	11.0%	9.6%
(7d)	V-NNxU (vfr)	10.1%	11.2%	9.5%
(7e)	U-BNxV	10.5%	11.6%	10.0%
(7f)	U-MFCCxV	10.8%	12.1%	10.1%
(8a)	LM resc. (7a)	10.0%	11.0%	9.3%
(8b)	LM resc. (7b)	9.7%	11.2%	9.2%
(8c)	LM resc. (7c)	9.3%	10.6%	9.1%
(8d)	LM resc. (7d)	9.9%	11.0%	9.2%
(8e)	LM resc. (7e)	9.9%	11.1%	9.4%
(8f)	LM resc. (7f)	10.2%	11.5%	9.3%
(9)	(8c)+(8e)	9.1%	10.3%	8.9%

Table 1. Word error rates of different decoding steps on DEV'07, DEV'08 and EVAL'08 unsequestered part.

(7f) fixed frame rate decoding with the U-MFCC model
(8a) rescoring of the lattices from (7a) with a NN-LM
(8b) rescoring of the lattices from (7b) with a NN-LM
(8c) rescoring of the lattices from (7c) with a NN-LM
(8d) rescoring of the lattices from (7d) with a NN-LM
(8e) rescoring of the lattices from (7e) with a NN-LM
(8f) rescoring of the lattices from (7f) with a NN-LM and
(9) lattice combination of the lattices from (8c) and (8e).

The performance of various decoding steps on DEV'07, DEV'08 and EVAL'08 unsequestered part is summarized in Table 1.

2.3. Acoustic modeling

The input speech is represented by either MFCC or PLP VTL-warped cepstra and a context window of 9 frames. Both sets of features are mean and variance normalized on a per speaker basis. An LDA transform is used to reduce the feature dimensionality to 40. The maximum-likelihood training of the acoustic model is interleaved with estimation of a global semi-tied covariance (STC) transform [6].

Words in the recognition lexicon are represented as sequences of phones, and phones are modeled with 3-state left-to-right HMMs that do not permit state skipping. All models have penta-phone cross-word acoustic context. The numbers of context-dependent states and Gaussian mixture components used in our GALE P3.5 evaluation system are summarized in Table 2 for the various models. Note that all models have 40-dimensional Gaussians except for the neural network features which are of dimension 60.

2.3.1. Improved discriminative training

The speaker-adapted models are discriminatively trained using a variety of techniques operating in both feature and model space. One such technique is a feature-space transformation called feature-space minimum phone error (fMPE) [2]. For model-space training, we experimented with minimum phone error (MPE) and boosted maximum mutual information (BMMI) [3], a new form of discriminative

Model	Nb. leaves	Nb. Gaussians
SI	3.5K	150K
U	5.0K	400K
V	6.0K	400K
SGMM	6.0K	150M*
NN	6.0K	270K

Table 2. GALE P3.5 evaluation system statistics. * The SGMM model uses an efficient subspace representation for the Gaussians.

Training style	Feature space	Model space	WER
–	none	ML	17.1%
Integrated	fMPE	ML	14.3%
Integrated	fMPE	MPE	13.0%
Integrated	fBMMI	ML	14.4%
Integrated	fBMMI	BMMI	13.2%
Two-stage	fMPE	ML	14.3%
Two-stage	fMPE	BMMI	12.4%

Table 3. Discriminative training results for the unvowelized speaker-adapted models on DEV'07.

training based on large margin classification. Additionally, we experimented with separate training of the transform and the models in two stages. In the first stage, we estimate the fMPE transform using the original set of ML-trained models. In the second stage, we rebuild the decision trees and new ML models in the resulting fMPE space, finishing with model-space discriminative training of the new ML models. In Table 3, we show various discriminative training results comparing MPE and BMMI, and single-stage versus two-stage training for unvowelized speaker-adapted models on DEV'07. It appears that fMPE and fBMMI have similar performance (14.3% versus 14.4%) and that two-stage training with alternating criteria (MPE for feature-space and BMMI for model-space) gives the best results.

2.3.2. Subspace Gaussian mixture models

Our Arabic system included a novel type of acoustic model, the Subspace Gaussian Mixture Model. We will summarize it here; more details on this type of model can be found in [1]. The basic idea can be expressed in the following three equations:

$$p(\mathbf{x}|j) = \sum_{i=1}^{I} w_{ji} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{ji}, \boldsymbol{\Sigma}_i) \quad (1)$$

$$\boldsymbol{\mu}_{ji} = \mathbf{M}_i \mathbf{v}_j \quad (2)$$

$$w_{ji} = \frac{\exp \mathbf{w}_i^T \mathbf{v}_j}{\sum_{i'=1}^{I} \exp \mathbf{w}_{i'}{}^T \mathbf{v}_j}, \quad (3)$$

so each state j is a mixture of I Gaussians (typically around 1000) and the full covariances $\boldsymbol{\Sigma}_i$ are shared across states. The means and mixture weights $\boldsymbol{\mu}_{ji}$ and w_{ji} are not parameters of the model but are a function of state-specific parameters $\mathbf{v}_j$ and globally shared parameters $\mathbf{M}_i$ and $\mathbf{w}_i$. The dimension of the vectors $\mathbf{v}_j$ is quite small (50 in experiments reported here). To introduce more parameters in individual states we generalize to a mixture of "substates", introducing a mixture weight c_{jm} and multiple vectors $\mathbf{v}_{jm}$ in each state. Although the expanded form of the model as a GMM

is extremely large (see Table 2), the model is actually quite fast to train and decode with.

We also introduce a form of speaker adaptation where

$$\boldsymbol{\mu}_{jmi} = \mathbf{M}_i \mathbf{v}_{jm} + \mathbf{N}_i \mathbf{v}^{(s)}, \quad (4)$$

where in our experiments, $\mathbf{v}^{(s)}$ is of the same dimension of $\mathbf{v}_{jm}$. The matrices $\mathbf{N}_i$ are another set of shared parameters that must be learned. In experiments reported here we combine this form of adaptation with fMLLR and MLLR. We also implemented discriminative training based on MPE and boosted MMI [3], together with the feature-space versions of both of these [2].

Our Arabic SGMM system was based on a decision tree built for a conventionally structured speaker adaptively trained system, which we also used to obtain the state alignment during training and the fMLLR transforms used during training (since our SGMM system was speaker adaptively trained). We used $I = 750$ Gaussians in the shared GMM structure, phonetic and speaker subspace dimensions of 50, and 200K sub-states allocated in proportion to a small power (0.2) of the state count. We trained a shared full-covariance GMM (the Universal Background Model, or "UBM") for four iterations on our training data irrespective of speech class, starting from a Maximum Likelihood clustering of the diagonal Gaussians in our baseline model. We did in-memory initialization of the model parameters as described in [1] from pruned statistics of all pairs i, j of aligned state j and aligned Gaussian i in the shared GMM. We then trained with MLE for 25 iterations; on each iteration we applied the E-M update to all parameter types (which is not provably correct) and from iteration 11 we reduced the step sizes by a factor of 2/3 in order to control the resulting instability. To speed up model training we trained on half the data (randomly selected on each iteration) in early iterations.

The discriminative training regime applied to the SGMM system was rather *ad-hoc* and and involved 6 iterations of model-space boosted MMI training, lattice regeneration, 4 further passes of boosted MMI, 5 iterations of fMPE and four iterations of MPE. In test time we used a less strong than normal acoustic scale: 1/12, rather than our normal value of around 1/18. The SGMM system used the variable frame rate decoding described later on. We estimated the fMLLR and MLLR transforms based on a transcript from the unvowelized system.

Our WER improvements before discriminative training were larger, at around 1% absolute WER, than our final improvements which were less than 0.5% absolute (see Table 1). This is mainly due to a reduced improvement from feature-space discriminative training; also multi-class MLLR was not implemented in this system so we had a reduced improvement from cross-adaptation.

2.3.3. Neural network features

We also experimented with neural network features, using a bottleneck network [7]. The first layer of weights and logistic nonlinearities in the network maps an input of 9 frames of 13 mean- and variance-normalized VTLN PLP features to a hidden representation of 800 dimensions. A second layer of weights projects 11 frames of this high dimensional hidden representation to a 20-dimensional output representation that is appended to our standard 40-dimensional feature vector. The weights come from an original, 4-layer network that was trained using the cross-entropy criterion to discriminate between 256 context-dependent HMM states. The resulting 60-dim. features are used in a speech recognition system that is trained using a single, global STC transform, speaker-adaptive training using feature-space MLLR, and feature- and model-space discriminative training using the BMMI criterion.

2.3.4. Variable frame rate decoding

To reduce degradation caused by speaking rate variations, a variable frame rate decoding technique is applied. The technique normalizes speaking rate at the utterance level by applying temporal warping in frontend processing. The warping factors are determined from speaking rate estimates derived from timing information given by the fixed frame rate decoding outputs. The detailed description of this method can be found in an accompanying paper. It is shown that the technique gives consistent improvements on both the vowelized and the unvowelized setups, as well as on the UBM system. For instance, on the final pass decoding with the vowelized system, 0.3% and 0.2% absolute WER reduction is achieved on EVAL'07 unsequestered and EVAL'08 unsequestered, respectively.

2.3.5. Training data selection

The GALE P3 training data is initially partitioned into two sets corresponding respectively to broadcast news (BN) and broadcast conversation (BC) data, using the provided labels. Likewise, testing data DEV'08 is split into BN and BC partitions DEV'08-BN and DEV'08-BC. All of the systems are unvowelized. Table 4 shows the performance after discriminative training for three different systems. We observe that the BN-only model performs well on both BN and BC test data. On the other hand, the BC-only model performs worse on both the BN and BC subsets. In the last column, we see the effects of combining the three systems using ROVER with equal weights. Improvements are seen across the board for the entire DEV'08 test data as well as the DEV'08-BN and DEV'08-BC subsets.

Test Data	Training Data			Rover
	ALL	BN only	BC only	
DEV'08	14.4%	14.6%	15.2%	12.5%
DEV'08-BN	10.4%	10.3%	11.5%	9.0%
DEV'08-BC	20.8%	21.5%	21.1%	18.1%

Table 4. Word error rates on DEV'08.

2.4. Recognition vocabulary and vowelization

One challenge in Arabic speech recognition is that the rich morphological structure of Arabic induces a large number of word forms. We dealt with this challenge by simply increasing the size of our recognition vocabulary to 774K words. The selection criterion is word frequency with source dependent cut-off values (placing emphasis on acoustic transcripts and machine translation vocabulary). While this vocabulary is quite large, the search is still manageable as shown in [8] where we compare the performance of both dynamic and static decoding approaches.

Another challenge in Arabic speech recognition is that there is a systematic mismatch between written and spoken Arabic. Short vowels and diacritics are normally omitted in written Arabic. There are two approaches to handling this mismatch between the acoustics and transcripts. In the "unvowelized" approach, words are modeled graphemically, in terms of their letter sequences, and the acoustics corresponding to the unwritten diacritics are implicitly modeled by the Gaussian mixtures in the acoustic model. In the "vowelized"

Models	Unvowelized	Vowelized
ML	17.2%	14.6%
Disc. trained	11.6%	11.0%

Table 5. Comparison of vowelized and unvowelized models before and after discriminative training on the DEV'07 test set.

LM	DEV'07	DEV'08	EVAL'08
Baseline (76M n-grams)	9.5%	11.0%	9.6%
Un-pruned (500M n-grams)	9.5%	10.9%	9.4%
500M n-grams LM + NN	9.3%	10.6%	9.1%

Table 6. Word error rates for different LMs.

approach, words are modeled phonemically, in terms of their sound sequences, and the correct vocalization of transcribed words is inferred during training. Note that even when vowelized models are used the word error rate calculation is based on unvowelized references. The vowelized forms are mapped back to unvowelized forms in scoring (NIST style scoring).

We use the Buckwalter Morphological Analyzer (Version 2.0) [9], and the Arabic Treebank to generate vowelized variants of each word. The pronunciation of each variant is modeled as the sequence of letters in the diacriticized word, including the short vowels. For *shadda* (consonant doubling), an additional consonant is added (same phone symbol) , and for *sukun* (no vowel), nothing is added. Table 5 shows a comparison of vowelized and unvowelized models before and after discriminative training. More details about vowelization can be found in [10].

2.5. Language modeling

In this subsection we describe the baseline language model (LM), and some rescoring experiments with augmented LMs.

In building the baseline LM, the first step involved training 4-gram LMs (with modified Kneser-Ney smoothing) for each source of data, e.g. transcripts of audio data, parts of the Arabic Gigaword corpus, etc. We created a 774K word vocabulary based on all the available corpora, insuring full coverage of the acoustic transcripts. The component 4-gram models were linearly interpolated, with weights optimized to minimize the perplexity on a held-out set. Typically the LMs corresponding to audio transcripts (broadcast news and conversations) have the highest weights because they are best matched to the domain of interest. The interpolated LM was pruned using entropy pruning to about 5M n-grams in order to build a static finite-state decoding graph. The output lattices are then rescored with a larger LM. In the past, we had typically used a pruned LM, for example, one containing 76M n-grams. Due to improved efficiency in our code, we decided to use an un-pruned LM that contains 500M n-grams.

Table 6 shows the results of the best single set of lattices obtained during the 2008 DARPA evaluation. Compared to using a large pruned LM (76M n-grams), using an un-pruned LM (500M n-grams) improves the WER by up to 0.2%. We also built a neural network language model as we had previously done [11]. The neural network used here was built on 11M words of acoustic transcripts. It is a 6-gram model with 100 hidden units and with an output vocabulary limited to the 20K most frequent words. This was the optimal configuration found in [11]. The neural network LM was interpolated with the baseline LM and used for lattice rescoring. The interpolation weights are optimized on the held-out set, the weight for the neural network LM being 0.3, the largest among all the 16 LMs used in the interpolation. It can be seen that significant improvements are obtained for all the test sets.

3. CONCLUSION

In this paper we presented a set of techniques for Arabic broadcast transcription that, taken together, lead to word error rates below 9%. Techniques that contribute to this level of performance include improved discriminative training, SGMM acoustic modeling, neural network acoustic features, variable frame rate decoding, exclusion of conversational training data and the use of unpruned n-grams language models and neural network language models.

4. ACKNOWLEDGMENT

The authors thank Mark Fuhs and Ian Lane from CMU for running the system combination experiments. We acknowledge the support of DARPA under Grant HR0011-06-2-0001 for funding part of this work.

5. REFERENCES

[1] D. Povey, "Subspace Gaussian Mixture Models for Speech Recognition," Tech. Rep. MSR-TR-2009-64, Microsoft Research, 2009.

[2] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively trained features for speech recognition," in *Proc. ICASSP*, 2005.

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

[4] G. Saon and D. Povey, "Penalty function maximization for large margin HMM training," in *Interspeech-08*, 2008.

[5] M.J.F Gales, "Maximum Likelihood Linear Transformations for HMM-based Speech Recognition," *Computer Speech and Language*, vol. 12, 1998.

[6] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Trans.Speech and Audio Processing*, vol. 7, no. 3, May 1999.

[7] František Grézl, Martin Karafiát, Stanislav Kontár, and Jan Černocký, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proc. ICASSP*, 2007.

[8] H. Soltau and G. Saon, "Dynamic network decoding revisited," in *Proc. ASRU*, 2009.

[9] T. Buckwalter, "LDC2004L02: Buckwalter Arabic morphological analyzer version 2.0," 2004.

[10] H. Soltau, G. Saon, B. Kingsbury, J. Kuo, L. Mangu, and A. Emami, "Advances in Arabic speech transcription at IBM under the DARPA GALE program," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 17, no. 5, 2009.

[11] A. Emami and L. Mangu, "Empirical study of neural network language models for Arabic speech recognition," in *Proc. ASRU*, 2007.