ROBUST SPEECH RECOGNITION BASED ON STRUCTURED MODELING, IRRELEVANT VARIABILITY NORMALIZATION AND UNSUPERVISED ONLINE ADAPTATION

Qiang HUO[1], *Donglai ZHU*[2]

[1]Microsoft Research Asia, Beijing, China
[2]Institute for Infocomm Research, Singapore
(E-mails: qianghuo@microsoft.com, dzhu@i2r.a-star.edu.sg)

ABSTRACT

We present a new approach to robust speech recognition based on structured modeling, irrelevant variability normalization (IVN) and unsupervised online adaptation (OLA). In offline training stage, a set of generic HMMs for basic speech units relevant to phonetic classification is trained along with several sets of feature transforms with different degrees of freedom by using a maximum likelihood (ML) IVN-based training strategy. In recognition stage, after a first-pass recognition, the most appropriate set of feature transforms is identified and adapted under ML criterion by using the unknown utterance itself, which is recognized again to achieve better performance by using the adapted feature transforms and the pre-trained generic HMMs. The effectiveness of the proposed approach is confirmed by evaluation experiments on Finnish Aurora3 database.

Index Terms— robust speech recognition, feature transformation, irrelevant variability normalization, online adaptation.

1. INTRODUCTION

In the past several years, we've been studying several approaches to robust automatic speech recognition (ASR) based on three key concepts, namely *structured modeling*, *irrelevant variability normalization* (IVN) and *unsupervised online adaptation* (OLA) (e.g., [9, 10, 11, 7, 6, 14, 15]). In structured modeling of a basic speech unit, "hidden" speech information (denoted as a feature vector sequence $X = \{x_t\}$) *relevant* to phonetic classification is modeled by a traditional Gaussian-mixture continuous density hidden Markov model (CDHMM) (referred to as *generic* CDHMM hereinafter), while factors *irrelevant* to phonetic classification are taken care of by an *auxiliary* module. More specifically, given an utterance with observed feature vector sequence $Y = \{y_t\}$, a specific label q_t can be identified for each D-dimensional feature vector y_t by an appropriate labeling procedure. Given q_t, one of the possible mapping functions (a.k.a. feature transforms), $g^{(q_t)}(\cdot)$, is selected, which characterizes the relationship between x_t and y_t, and handles possible "distortions" caused by factors *irrelevant* to phonetic classification. The set of feature transforms (FTs), $\{g^{(f)}(\cdot); f = 1, \cdots, F\}$, is shared by all the basic speech units, which enables efficient and effective unsupervised OLA in recognition stage. Let $\Lambda = \{\lambda\}$ denote the set of generic CDHMMs as well as their model parameters, and Θ denote the set of parameters for F FTs. An IVN-based training procedure can then be designed to estimate Λ and Θ from a large amount of *diversified* training data, $\mathcal{Y} = \{Y_i\}_{i=1}^{I}$, where Y_i is a sequence of feature vectors of the ith training utterance. In recognition stage, the parameters of the auxiliary module, Θ, can be updated via unsupervised OLA by using the unknown utterance itself, which is recognized again to achieve better performance by using the compensated models composed from the generic CDHMMs and the adapted auxiliary module.

Over the years, we have studied several forms of feature transformation with different degrees of flexibility. Some of them (e.g., [10, 11, 6, 14]) can be implemented in a pure *feature-compensation* mode during recognition stage without change of decoder, therefore are quite efficient even for large vocabulary continuous speech recognition (LVCSR) tasks; while approaches in e.g., [9, 7] work in a pure *model-compensation* mode during recognition stage, therefore are computationally more expensive for LVCSR tasks. Yet the approach in [15] can be implemented in a *hybrid* mode, where each frame of feature vector is still subject to a linear transformation, while the decoder need be changed slightly by including an appropriate "Jacobian" term when evaluating the probability density function (PDF) value for each Gaussian component in CDHMMs, therefore its computational complexity lies between the above two sets of approaches. In terms of the effectiveness of the above approaches, we observed that none of them prevails in all possible training-testing conditions as we evaluated on Finnish Aurora3 task [1]. It is therefore well-motivated to develop new approaches which can take advantage of the strengths of the above different approaches. In this paper we propose several such new approaches to robust ASR.

The rest of the paper is organized as follows. In Section 2, we present our new approaches. In Section 3, we report experimental results. Finally, we conclude the paper in Section 4.

2. OUR NEW APPROACHES

2.1. Multiple Types of Feature Transforms

Given the set of training data $\mathcal{Y}$, suppose that they can be partitioned into E "acoustic conditions", each characterized by a Gaussian-mixture model (GMM):

$$p(y|e) = \sum_{k=1}^{K} p(k|e)p(y|k,e) = \sum_{k=1}^{K} p(k|e)\mathcal{N}(y; \xi_k^{(e)}, R_k^{(e)})$$

where e is the index of acoustic condition class, $\mathcal{N}(\cdot; \xi, R)$ is a normal distribution with D-dimensional mean vector ξ and diagonal covariance matrix R. Readers are referred to [12] for the approach we used for the automatic clustering of acoustic conditions from training data $\mathcal{Y}$, the labeling of an utterance Y (in both training and recognition) to a specific acoustic condition, and the estimation of the above model parameters.

In this paper, we study the following three forms of FT functions. The first one (referred to as FT3) is defined as follows [6]:

$$\hat{x} \triangleq \mathcal{F}_3(y; \Theta^{(e)}) = A^{(e)}y + b^{(e)} \ , \qquad (1)$$

where $\Theta^{(e)} = \{A^{(e)}, b^{(e)}\}$ represents the trainable parameters of the transformation, and e denotes the corresponding acoustic condition to which y belongs. In this case, $q_t = e$, $F = E$, and $x_t = g^{(q_t)}(y_t) = A^{(e)}y_t + b^{(e)}$.

The second FT function (referred to as FT5) is defined as [6]

$$\hat{x} \triangleq \mathcal{F}_5(y; \Theta^{(e)}) = A^{(e)}y + b_k^{(e)} , \quad (2)$$

where, for the acoustic condition e which y belongs to,

$$k = \arg \max_{l=1,\ldots,K} p(l|y, e) \quad (3)$$

with

$$p(l|y, e) = \frac{p(l|e)p(y|l, e)}{\sum_{j=1}^{K} p(j|e)p(y|j, e)} ,$$

and $\Theta^{(e)} = \{A^{(e)}, b_l^{(e)}, l = 1, \ldots, K\}$. In this case, $q_t = (e, k)$, $F = E \times K$, and $x_t = g^{(q_t)}(y_t) = A^{(e)}y_t + b_k^{(e)}$.

The third FT function (referred to as FT6) is defined as [15]:

$$\hat{x} \triangleq \mathcal{F}_6(y; \Theta^{(e)}) = A_k^{(e)}y + b_k^{(e)} , \quad (4)$$

where $\Theta^{(e)} = \{A_l^{(e)}, b_l^{(e)}; l = 1, \ldots, K\}$, and k is calculated by using Eq. (3). In this case, $q_t = (e, k)$, $F = E \times K$, and $x_t = g^{(q_t)}(y_t) = A_k^{(e)}y_t + b_k^{(e)}$.

In the above equations, $A^{(e)}$ or $A_k^{(e)}$ is a nonsingular $D \times D$ matrix, and $b^{(e)}$ or $b_k^{(e)}$ is a D-dimensional vector. In the following three subsections, we present three new approaches, respectively.

2.2. Approach-1: Parallel Decoding with Multiple Systems

In training stage, do ML-IVN training using each type of the above mentioned transforms, and obtain

- the set of FT3 transforms and the corresponding set of FT3-based generic CDHMMs [6];
- the set of FT5 transforms and the corresponding set of FT5-based generic CDHMMs [6];
- the set of FT6 transforms and the corresponding set of FT6-based generic CDHMMs [15].

Let $\mu_{sm} = [\mu_{sm1}, \cdots, \mu_{smD}]^T$ denote the D-dimensional mean vector, and $\Sigma_{sm} = diag\{\sigma_{sm1}^2, \cdots, \sigma_{smD}^2\}$ denote the diagonal covariance matrix for Gaussian component m in state s of generic CDHMMs.

The recognition procedure is as follows:

- **Step 1:** Given an unknown utterance Y, do parallel decoding by using FT3, FT5, and FT6 (as trained above) and the corresponding sets of generic CDHMMs to obtain the "first-pass" recognition results, R1(FT3), R1(FT5), R1(FT6), respectively. As a by-product, we also have the corresponding likelihood scores, L1(FT3), L1(FT5), L1(FT6), respectively. Pick up the hypothesis which gives the highest likelihood, and determine which form of transforms and which set of generic CDHMMs are used in OLA.
- **Step 2:** Given the recognition result, the form of transforms, and the set of generic CDHMMs, do OLA to update $b^{(e)}$ or $b_k^{(e)}$ using the corresponding IVN-trained transforms as initial values. The updating formula of $b^{(e)}$ for FT3 is as follows:

$$b_d^{(e)} = \frac{\sum_{t,s,m} \delta[e, q_t]\zeta_t(s, m)(\mu_{smd} - A_d^{(e)} \cdot y_t)/\sigma_{smd}^2}{\sum_{t,s,m} \delta[e, q_t]\zeta_t(s, m)/\sigma_{smd}^2} , \quad (5)$$

where $\delta[\cdot, \cdot]$ is a Kronecker delta function, $A_d^{(e)}$ is the dth row of $A^{(e)}$, and $\zeta_t(s, m)$ is the occupation probability of Gaussian component m in state s of CDHMMs, at time t of the current compensated observation $\hat{x}_t = \mathcal{F}_3(y_t; \Theta^{(e)})$.

The updating formula of $b_k^{(e)}$ for FT5 is as follows:

$$b_{kd}^{(e)} = \frac{\sum_{t,s,m} \delta[(e, k), q_t]\zeta_t(s, m)(\mu_{smd} - A_d^{(e)} \cdot y_t)/\sigma_{smd}^2}{\sum_{t,s,m} \delta[(e, k), q_t]\zeta_t(s, m)/\sigma_{smd}^2} , \quad (6)$$

where $A_d^{(e)}$ is the dth row of $A^{(e)}$, and $\zeta_t(s, m)$ is the occupation probability of Gaussian component m in state s of CDHMMs, at time t of the current compensated observation $\hat{x}_t = \mathcal{F}_5(y_t; \Theta^{(e)})$.

The updating formula of $b_k^{(e)}$ for FT6 is as follows:

$$b_{kd}^{(e)} = \frac{\sum_{t,s,m} \delta[(e, k), q_t]\zeta_t(s, m)(\mu_{smd} - A_{kd}^{(e)} \cdot y_t)/\sigma_{smd}^2}{\sum_{t,s,m} \delta[(e, k), q_t]\zeta_t(s, m)/\sigma_{smd}^2} , \quad (7)$$

where $A_{kd}^{(e)}$ is the dth row of $A_k^{(e)}$. Note that $\zeta_t(s, m)$ is calculated by a forward-backward procedure with the PDF value for each frame of observation evaluated as follows:

$$p(y|\Lambda, \Theta) = \mathcal{N}(\mathcal{F}_6(y; \Theta^{(e)}); \mu_{sm}, \Sigma_{sm})|\det(A_k^{(e)})| , \quad (8)$$

where $\det(A_k^{(e)})$ is the determinant of matrix $A_k^{(e)}$.

- **Step 3:** Do new decoding using the updated transforms and the set of generic CDHMMs to obtain new recognition result.
- **Step 4:** Steps 2 and 3 can be repeated until a pre-specified criterion is satisfied (e.g., a fixed number of cycles).

2.3. Approach-2: Parallel Decoding with Multiple Types of Feature Transforms and Single Set of Generic CDHMMs

The training procedure of this approach is as follows:

- **Step 1:** Do ML-IVN training using FT6 as described in [15], and obtain a set of FT6 transforms, and a set of FT6-based generic CDHMMs.
- **Step 2:** Fix FT6-based generic CDHMMs, estimate the set of FT3 transforms and the set of FT5 transforms using ML-IVN training as follows:
  - **Step 2-1:** Use FT6 to transform feature vectors, calculate the corresponding occupation probabilities conditioned on FT6-based generic CDHMMs; Initialize $A^{(e)}$ as identity matrix and $b^{(e)}$ (for FT3) or $b_k^{(e)}$ (for FT5) as zero bias vector; Re-estimate $A^{(e)}$ and $b^{(e)}$ (for FT3) or $b_k^{(e)}$ (for FT5) as described in [6].
  - **Step 2-2:** Given the current estimate of $A^{(e)}$ and $b^{(e)}$ (for FT3) or $b_k^{(e)}$ (for FT5), and the FT6-based generic CDHMMs, do standard updating as described in [6].
  - **Step 2-3:** Repeat Step 2-2 if necessary (in our experiments, we skipped this step).

The recognition procedure of this approach is as follows:

- **Step 1:** Given an unknown utterance Y, do parallel decoding by using FT3, FT5, and FT6 (as trained above) and the common set of FT6-based generic CDHMMs to obtain the "first-pass" recognition results, R1(FT3), R1(FT5), R1(FT6),

respectively. As a by-product, we also have the corresponding likelihood scores, L1(FT3), L1(FT5), L1(FT6), respectively. Pick up the hypothesis which gives the highest likelihood, and determine which form of transforms is used in OLA.

- **Step 2:** Given the recognition result and the form of transforms, do OLA to update $b^{(e)}$ (for FT3) or $b_k^{(e)}$ (for FT5 and FT6) using the corresponding IVN-trained transforms as initial values. The updating formulas are the same as Eqs. (5) $\sim$ (7).
- **Step 3:** Do new decoding using the updated transforms and the set of FT6-based generic CDHMMs to obtain new recognition result.
- **Step 4:** Steps 2 and 3 can be repeated until a pre-specified criterion is satisfied (e.g., a fixed number of cycles).

2.4. Approach-3: Speeding Up Approach-2 via Lattice Re-scoring

The training procedure of this approach is the same as Approach-2. In order to reduce recognition time, a new recognition procedure is proposed as follows:

- **Step 1:** Given an unknown utterance Y,
  - Do decoding by using FT5 (as trained above) and the common set of FT6-based generic CDHMMs to obtain the "first-pass" recognition results, R1(FT5), the corresponding likelihood score, L1(FT5), and a lattice of recognition results, Lattice(FT5).
  - Given Lattice(FT5), do lattice re-scoring by using FT3 and FT6 (as trained above) to obtain the "first-pass" re-scoring results, R1(FT3) and R1(FT6), respectively. As a by-product, we also have the corresponding likelihood scores, L1(FT3) and L1(FT6), respectively.
  - Pick up the hypothesis which gives the highest likelihood, and determine which form of transforms is used in OLA.
- **Step 2:** The same as Step 2 of the recognition procedure of Approach-2.
- **Step 3:** Given Lattice(FT5), do new decoding via lattice re-scoring using the updated transforms and the set of FT6-based generic CDHMMs to obtain new recognition result.
- **Step 4:** Steps 2 and 3 can be repeated until a pre-specified criterion is satisfied (e.g., a fixed number of cycles).

3. EXPERIMENTS AND RESULTS

3.1. Experimental Setup

We use Finnish Aurora3 database [1] to verify our algorithms. Aurora3 contains utterances of connected digits that were recorded by using both close-talking (CT) and hands-free (HF) microphones in cars under several driving conditions to reflect some realistic scenarios for typical in-vehicle ASR applications. There are roughly three conditions: *quiet*, *low noise*, and *high noise*. The database is divided into following three subsets according to matching degree between training data and test data:

- **Well-Matched (WM) condition**: Both training and testing data include utterances recorded by both CT and HF microphones from all conditions;

Table 1. Number of testing utterances that use each of three forms of feature transform for Approach-1 and Approach-2 respectively.

Approaches	Conditions	FT3	FT5	FT6
Approach-1	WM	0	8	1312
	MM	0	27	221
	HM	7	425	64
Approach-2	WM	0	8	1312
	MM	1	34	213
	HM	9	473	14

- **Medium-Mismatched (MM) condition**: Training data includes utterances recorded by HF microphone in the *quiet* and *low noise* conditions. Testing data includes utterances recorded by HF microphone in the *high noise* condition;
- **High-Mismatched (HM) condition**: Training data includes utterances recorded by CT microphone from all conditions. Testing data includes utterances recorded by HF microphone in the *low noise* and *high noise* conditions.

Therefore, the MM condition simulates mainly the mismatch caused by a noisy environment due to different driving speeds and possible background music. The HM condition simulates mainly the mismatch caused by different transducers.

In our experiments, the ETSI Advanced Front-End (AFE) as described in [2] is used for feature extraction from a speech utterance. A feature vector sequence is extracted from the input speech utterance via a sequence of processing modules that include noise reduction, waveform processing, cepstrum calculation, blind equalization, and "server feature processing". Each frame of feature vector has 39 features that consists of 12 MFCCs (C_1 to C_{12}), a combined log energy and C_0 term, and their first and second order derivatives. Although all the feature vectors are computed from a given speech utterance, the feature vectors that are sent to the speech recognizer and the training module are those corresponding to speech frames, as detected by a VAD module described in Annex A of [2]. In FT-based experiments, all the training data are clustered into 8 different acoustic conditions (i.e. $E = 8$), of which each is modeled by a GMM consisting of 32 Gaussian components (i.e. $K = 32$).

Each digit is modeled as a whole-word left-to-right CDHMM with 16 emitting states, 3 Gaussian mixture components with diagonal covariance matrices per state. Besides, two pause models, "sil" and "sp", are created to model the silence before/after the digit string and the short pause between any two digits, respectively. The "sil" model is a 3-emitting state CDHMM with a flexible transition structure as described in [5]. Each state is modeled by a mixture of 6 Gaussian components with diagonal covariance matrices. The "sp" model consists of 2 dummy states and a single emitting state which is tied with the middle state of "sil". During recognition, an utterance can be modeled by any sequence of digits with the possibility of a "sil" model at the beginning and at the end and a "sp" model between any two digits. For lattice generation in Step 1 of Approach-3, 5 tokens per CDHMM state are used [13].

3.2. Experimental Results

Table 2 summarizes a comparison of word error rates (WERs in %) of following 12 systems:

- **CDHMM Baseline:** a system trained from multi-condition training data using ETSI-AFE;
- **Stochastic-Matching:** baseline system plus feature-space stochastic matching (SM) [8];

Table 2. Comparison of word error rates (in %) of a CDHMM-based baseline system, a system with feature-space stochastic matching, a system with unsupervised CMLLR adaptation, several FT-based IVN-trained systems without and with unsupervised online adaptation (OLA), and systems with three proposed new approaches.

Testing Conditions	CDHMM Baseline	Stochastic Matching	CMLLR OLA	FT3 IVN	FT5 IVN	FT6 IVN	FT3 OLA	FT5 OLA	FT6 OLA	New Approaches		
										1	2	3
WM($\times 40\%$)	3.95	3.65	3.60	3.08	2.92	2.46	2.62	2.74	2.26	2.26	2.05	2.08
MM($\times 35\%$)	19.70	16.48	14.36	16.48	16.48	15.05	12.52	14.57	13.68	13.68	12.31	12.31
HM($\times 25\%$)	14.28	11.27	10.39	15.37	16.61	21.70	12.16	12.72	16.50	12.72	9.93	10.57
Average	12.05	10.05	9.06	10.84	11.09	11.68	8.47	9.38	9.82	8.87	7.61	7.78

- **CMLLR-OLA:** baseline system plus unsupervised CMLLR adaptation (e.g. [3, 4]);
- **FT3/5/6-IVN:** three IVN-trained systems based on different FTs and without unsupervised OLA [6, 15];
- **FT3/5/6-OLA:** three IVN-trained systems based on different FTs and with unsupervised OLA [14];
- **Approach-1/2/3:** three systems based on three proposed approaches, respectively.

For each testing utterance, a global diagonal transformation matrix and a bias vector are estimated in CMLLR adaptation, while a bias vector is estimated in SM approach. In all the unsupervised OLA experiments, two adaptation cycles are performed. From results in Table 2, we make the following observations and discussions:

- When representative yet diversified training data (i.e. in WM condition) are available, all FT-based IVN training approaches help reduce recognition errors. More flexible the FT function, better the performance. This also explains why we use FT6 in Step 1 of Approach-2.
- When there exists big mismatch between training and testing conditions (i.e. in HM condition), FT-based IVN training without OLA may not work, simply because FTs learned from training data cannot be generalized to testing condition. More flexible the FT function, worse the performance.
- After the unsupervised OLA of FT parameters, "IVN+OLA" performs better than "IVN only" for all FT functions and all "training-testing" conditions. However, no single FT function prevails in all possible "training-testing" conditions: FT6 performs best for "WM" condition, but worst for "HM" condition; while FT3 performs best for "HM" condition.
- By taking advantage of the strengths of different types of FTs, Approach-2 is indeed an "all-round" approach which achieves the best overall performance among the approaches compared, while Approach-3 achieves slightly degraded WERs due to the use of lattice re-scoring, but is computationally more efficient than Approach-2. Table 1 summarizes the distribution of the number of testing utterances that use each of three FT forms for Approach-1 and Approach-2 respectively. It is indeed the case that different transforms are selected for different testing utterances under different conditions. Using FT5 in first-pass decoding of Approach-3 to generate the lattice for later re-scoring is a good tradeoff between lattice accuracy and computational complexity.

4. SUMMARY

In this paper, we have studied three new approaches to robust ASR. The best-performing Approach-2 achieves word error rates of 2.05%, 12.31%, 9.93% for WM, MM and HM conditions on Finnish Aurora3 task respectively. In comparison with the CDHMM-based baseline system using ETSI Advanced Front-End, this represents a relative word error rate reduction of 48.1%, 37.5% and 30.5% respectively. As a future work, we will study how effective the above approaches are for LVCSR tasks.

5. REFERENCES

[1] Aurora document AU/217/99, "Availability of Finnish speechdat-car database for ETSI STQ WI008 front-end standardisation," Nokia, Nov. 1999.

[2] ETSI standard document, "Speech processing, transmission and quality aspects (STQ); distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," ETSI ES 202 050 v1.1.1 (2002-10), 2002.

[3] V. V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Trans. on Speech and Audio Processing*, Vol. 3, No. 5, pp.357-366, 1995.

[4] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, Vol. 12, pp.75-98, 1998.

[5] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy conditions," *ISCA ITRW ASR-2000*, Paris, France, pp.181-188.

[6] Q. Huo and D. Zhu, "A maximum likelihood training approach to irrelevant variability compensation based on piecewise linear transformations," *Proc. Interspeech-2006*, pp.1129-1132.

[7] Q. Huo, D. Zhu and J. Wu, "Unsupervised online adaptation of segmental switching linear Gaussian hidden Markov models for robust speech recognition," *Proc. ICASSP-2006*, pp.I-1125-1128.

[8] A. Sankar and C.-H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. on Speech and Audio Processing*, Vol. 4, No. 3, pp.190-202, 1996.

[9] J. Wu and Q. Huo, "A switching linear Gaussian hidden Markov model and its application to nonstationary noise compensation for robust speech recognition," *Proc. Interspeech-2003*, pp.977-980.

[10] J. Wu and Q. Huo, "An environment compensated minimum classification error training approach based on stochastic vector mapping," *IEEE Trans. on Audio, Speech and Language Processing*, Vol. 14, No. 6, pp.2147-2155, 2006.

[11] J. Wu, Q. Huo, and D. Zhu, "An environment compensated maximum likelihood training approach based on stochastic vector mapping," *Proc. ICASSP-2005*, pp. 429-432.

[12] J. Wu, D. Zhu, and Q. Huo, "A study of minimum classification error training for segmental switching linear Gaussian hidden Markov models," *Proc. ICSLP-2004*, pp.2813-2816.

[13] S. J. Young, *et al.*, *The HTK Book* (for HTK Version 3.3), 2005.

[14] D. Zhu and Q. Huo, "A maximum likelihood approach to unsupervised online adaptation of stochastic vector mapping function for robust speech recognition," *Proc. ICASSP-2007*, pp.IV-773-776.

[15] D. Zhu and Q. Huo, "Irrelevant variability normalization based HMM training using MAP estimation of feature transforms for robust speech recognition," *Proc. ICASSP-2008*, pp.4717-4720.