A Vector Space Approach to Environment Modeling for Robust Speech Recognition

Yu Tsao and Chin-Hui Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0250, USA
{yutsao, chl}@ece.gatech.edu

10.21437/Interspeech.2006-271

Abstract

We propose a vector space approach to characterizing environments for robust speech recognition. We represent a given environment by a super-vector formed by concatenating all the mean vectors of the Gaussian mixture components of the state observation densities of all hidden Markov models trained in the particular environment. New environment super-vectors can now be obtained either by an interpolation method with a collection of super-vectors trained from many real or simulated environments or by a transformation performed on an anchor super-vector for a specific environment, such as a clean condition. At a 5dB signal-to-noise (SNR) level, both interpolation- and transformation-based approaches achieve a significant error rate reduction of close to 47% from a baseline system with cepstral mean subtraction (CMS) with only two adaptation utterances. When incorporating *N*-best information to perform unsupervised adaptation at 5dB SNR with the same two utterances, we achieve a relative error reduction of about 40%, close to that achieved in the supervised mode.

Index Terms: acoustic modeling, environment adaptation

1. Introduction

Automatic speech recognition (ASR) systems had been largely improved since statistical hidden Markov model (HMM) was established as a fundamental tool to represent speech signals. However, HMMs do not generalize well from the training to testing mismatch conditions. Many robustness techniques have been developed to reduce such mismatch conditions. Among them, the CDCN algorithm [1] performs feature compensation with a correction vector, which is estimated with a VQ codeword, indicating the gap between the training and testing environments. Stochastic matching [2] is also an effective way to estimate the mismatch factor in a maximum likelihood, self-adaptation manner.

It is clear that if an unknown environment can be accurately characterized the performance of speech recognition systems in adverse conditions can usually be effectively improved. In this study we propose a vector space approach to environment modeling that we model each environment of interest by a super-vector consisting of the entire set of mean vectors from all Gaussian components of a set of HMMs intended to be used for the particular environment. If we have available a large collection of such vectors covering the environment space, we can determine the vector for an unknown condition, and use them to construct HMMs for the testing environment.

To estimate the super-vector for an unknown environment we propose two methods. The first technique interpolates the unknown vector with a large collection of environment vectors obtained from the sets of HMMs trained with speech data collected in their corresponding environments. We call this method *interpolation-based environment modeling* (IEM). The second technique is based on estimating the super-vector by performing a transformation over an anchor super-vector. This transformation matrix is often interpreted as a correlation between a noisy super-vector and the anchor super-vector, and can be obtained by another large collection of transformation matrices, each corresponding to the transformation required for a particular known environment. We call this method *transformation-based environment modeling* (TEM).

Two key issues are worth mentioning. The first is about the availability of a large collection of super-vectors to provide a good coverage of the environment space. The second is the amount of data needed to estimate the interpolation weights in IEM, and the transformation matrix in TEM. The latter can be addressed by principle component analysis (PCA) [3]. We will discuss the specific techniques involved here in Sections 2 and 3 when we present the IEM and TEM approaches.

For obtaining a large set of super-vectors, we can collect speech data from a large population of talkers if we intend to characterize the speaker space. For modeling the environment with the combination of adverse conditions and noise levels is too prohibitive that we decide to employ Monte Carlo (MC) [4] technique to simulate a wide range of conditions. The MC method enables us to quantitatively and qualitatively analyze the properties and the coverage of the environment space.

The TIMIT corpus [5] is used as a domain-independent, non-digit training database to obtain phone HMMs in clean conditions. We test the proposed approach on the Aurora 2 connected digit recognition task [6] in diverse conditions. And the noise sources needed to perform Monte Carlo simulation are extracted from the wide selection of noise types in the NOISEX-92 database [7]. The proposed IEM and TEM approaches achieve a close to 50% error reduction over a conventional cepstral mean subtraction method when a few utterances are used for adapting HMMs to the testing environment. Moreover, in an unsupervised adaptation mode significant improvements in performance over the baseline system have also been observed.

2. Interpolation-based environment modeling

There are two steps in the IEM approach. In the offline step, the entire set of mean vectors of a HMM model set for one environment p is concatenated into a super-vector $\mathbf{X}_p$, where p=1, 2…P for P different environments. The dimension for each

super-vector is $G \times D_V$, where G is the number of Gaussian mixtures for one environment and D_V is the dimension for each mean vector. The ensemble of these super-vectors forms an environment space with dimension P. In this paper, this set is referred to as an I-environment space, or simply I-space.

In the online step, with a small amount of speech segments, a super-vector $\mathbf{X}_{test}$, for the unknown testing environment, is estimated from the available set of P environment vectors. Three methods can be used to estimate the super-vector.

2.1. Best first

A best first method can be used to determine $\mathbf{X}_{test}$ by locating the most matched super-vector in the I-space as:

$$\mathbf{X}_{test} = arg \max_{p} L(\mathbf{O}_{test} \mid \mathbf{X}_p) \quad , \quad p=1, 2\ldots P, \tag{1}$$

where L is the likelihood function and $\mathbf{O}_{test}$ is the set of feature vectors corresponding to the adaptation speech data.

2.2. Full space linear combination

An interpolation method is developed to improve the performance. It generates the super-vector $\mathbf{X}_{test}$ by a linear combination of super-vectors in the I-space with a set of weight coefficients $\hat{w}_p$, p=1, 2…P, i.e.:

$$\mathbf{X}_{test} = \sum_{p=1}^{P} \hat{w}_p \mathbf{X}_p \quad . \tag{2}$$

The estimation of the weight coefficients is performed in the online step according to some optimization criteria. Here we use a maximum likelihood (ML) algorithm:

$$\hat{w}_p = \arg\max_{w_p} L(\mathbf{O}_{test} \mid \sum_{p=1}^{P} w_p \mathbf{X}_p) . \tag{3}$$

2.3. Reduced PCA space linear combination

When the amount of adaptation data is very limited, dimension reduction techniques will be needed to properly reduce the number of weight coefficients to be estimated. We used a PCA method on the I-space, while keeping the K_I eigenvectors with the highest singular values, a principle component I-space with a reduced dimension K_I is thus constructed. In the online step, a super-vector for the unknown environment is estimated as:

$$\mathbf{X}_{test} = \sum_{k=1}^{K_I} \hat{v}_k \, \mathbf{e}_{\mathbf{X}k} , \tag{4}$$

where $\mathbf{e}_{\mathbf{X}k}$ is the k-th principle eigenvector in the I-space, and $\hat{v}_k$, k=1, 2…K_I, are the corresponding weight coefficients calculated based on the following ML algorithm:

$$\hat{v}_k = arg \max_{v_k} L(\mathbf{O}_{test} \mid \sum_{k=1}^{K_I} v_k \, \mathbf{e}_{\mathbf{X}k}) . \tag{5}$$

3. Transformation-based environment modeling

For the TEM approach, an environment is described by a transformation characterizing its correlation with the training environment. The super-vector $\mathbf{X}_{test}$ for the testing environment is computed as:

$$\mathbf{X}_{test} = \mathbf{W}_{test} \mathbf{X}_{train} \tag{6}$$

where $\mathbf{X}_{train}$ is an anchor (reference) super-vector for the training environment, and $\mathbf{W}_{test}$ is the transformation for the testing environment. The implementation can also be divided into the online and offline steps. In the offline phase, P sets of transformations, $\mathbf{W}_1$, $\mathbf{W}_2$… $\mathbf{W}_P$, corresponding to P different environments are calculated as follows:

$$\mathbf{W}_p = arg \max_{\mathbf{W}} L(\mathbf{O}_p \mid \mathbf{W}\mathbf{X}_{train}) , p=1, 2\ldots P, \tag{7}$$

where $\mathbf{O}_p$ is all the training data from the p-th environment. Parameters in one transformation are then vectorized into a super-vector, and with these P super-vectors, a transformation-based environment (T-environment) space, or T-space, with dimension P, is constructed. On the other hand in the online phase, the transformation $\mathbf{W}_{test}$ is estimated with some adaptation speech data from the unknown testing environment.

3.1. Best first

The best first method is still the most intuitive solution:

$$\mathbf{W}_{test} = arg \max_{p} L(\mathbf{O}_{test} \mid \mathbf{W}_p \mathbf{X}_{train}) \; p=1, 2\ldots P. \tag{8}$$

3.2. Full space linear combination

Next we consider the linear combination method:

$$\mathbf{W}_{test} = \sum_{p=1}^{P} \tilde{w}_p \, \mathbf{W}_p , \tag{9}$$

where the coefficients $\tilde{w}_p$ are estimated from the ML criterion:

$$\tilde{w}_p = arg \max_{w_p} L(\mathbf{O}_{test} \mid (\sum_{p=1}^{P} w_p \mathbf{W}_p)\mathbf{X}_{train}) . \tag{10}$$

3.3. Reduced PCA space linear combination

PCA can also be utilized to construct a principle component T-environment space with a reduced dimension K_T. Then the super-vector for the testing environment can be estimated as:

$$\mathbf{W}_{test} = \sum_{k=1}^{K_T} \tilde{v}_k \, \mathbf{e}_{\mathbf{W}k} , \tag{11}$$

where $\mathbf{e}_{\mathbf{W}k}$ is the k-th eigen-vector of the super-vector space formed by the all the transformation matrices in the T-space, and coefficients $\hat{v}_k$ are computed based on the ML criterion:

$$\tilde{v}_k = arg \max_{v_k} L(\mathbf{O}_{test} \mid (\sum_{k=1}^{K_T} v_k \, \mathbf{e}_{\mathbf{W}k})\mathbf{X}_{train}) . \tag{12}$$

After $\mathbf{W}_{test}$ is computed, the super-vector $\mathbf{X}_{test}$ for the testing environment can be obtained with the formulation in Eq. (6).

4. Experimental setup and result analysis

Two different corpora, TIMIT [5] and Aurora 2 [6], were used as training and testing sets in all the experiments. Fifteen different types of noise sources were selected from the NOISEX-92 database [7]. Monte Carlo method [4] can now be employed to simulate the noise data at different SNR levels with various noise types and then be added to the TIMIT data to obtain new artificial training data as a particular point in the environment space. When there are S different noise sources with L different SNR levels, P (P=S×L) noisy environments can be constructed to simulate the environment space. Moreover,

with different combinations of noise sources and SNR levels, we may qualitatively and quantitatively analyze the characteristics of the environment space.

For compatibility in sampling rates, all the speech and noise data were down-sampled to 8 KHz before performing feature extraction. We used a commonly adopted feature vector of 39 elements, consisted of 13 MFCC parameters plus their first and second order time derivatives. An utterance-level cepstral mean subtraction (CMS) was performed for normalization.

For each environment the entire training set with 3696 utterances in the TIMIT or simulated TIMIT database was used to train 45 English phone HMMs. All models have 3 states with each state characterized by 16 Gaussian mixture components. The set A in Aurora 2 database was used as the testing set. The utterances in set A are based on speech data from TIDIDIGITS corpus [8] that pass through a linear filter and/or contaminated by four types of noise (subway, babble, car and exhibition) with different SNR levels. Conventional digit-specific trained HMMs produced about 6% and 28% digit error rates averaging over 4 SNR levels at 5dB to 20 dB, in multi-condition and clean training respectively [9]. We expected much lower recognition rates in our experiments since there is no digit knowledge involved in estimating HMMs, nor in building the environment spaces. Here the digit error rate achieved by the TIMIT trained HMMs with per-utterance CMS is 5.27% for the clean condition in set A. From our preliminary experiments we observed that increasing the number of noise types often does not improve recognition performance accordingly, and some representative noises can be used to effectively build the environment space. On the other hand, the coverage of SNR levels is more related to the performance in environment modeling. Instead of using all environments, we select 48 (with white, pink and car noises at 16 SNR levels between 0dB to 40dB) representative conditions to build the environment space in the following experiments.

4.1. Comparison of IEM methods

In Figure 1 we plot the digit error rates, for the IEM approach with 2, 4, 6, 8 and 10 adaptation utterances, averaging over 4 SNR levels at 5, 10, 15, 20 dB and 4 different noise types of set A from the Aurora 2 database. Digit error rate achieved with per-utterance CMS was 38.81%, and it served as the baseline. Curves (a) and (b) are best first and linear combination methods without any dimensionality reduction, i.e., P=48. While curves (c) and (d) are PCA methods for reduced-dimension I-space, with K_I=5, and 10, respectively. Clearly IEM outperformed CMS. For K_I=10 with 2 adaptation utterances we observed an error rate reduction of about 43% from 38.81% to 21.5%.

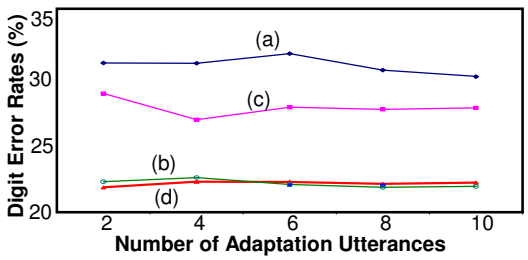

Figure 1: *Comparison of IEM approaches with different online processes. Curves (a), (b), (c) and (d) are for the best first, linear combination and PCA with K_I=5 and K_I=10.*

It is noted that the best first method can not achieve as good performance as the others. For the results obtained with PCA performance will degrade if the dimension is over-reduced, e.g. K_I=5, while performance did not change much for K_I>10. When the amount of adaptation data was limited, the full space linear combination method with all 48 environments gave a slightly worse error rate than the ones when K_I>10. All the performance levels became similar when more data were available. Furthermore the difference in performance between K_I=10 and K_I=25 was very small, and therefore we only reported the results with K_I=10 in Figure 1. It is also noted that with 10 adaptation utterances the linear combination method in (b) gave the best error rate reduction when all 48 environments were used.

4.2. Comparison of TEM methods

Similar to the above analysis in Figure 1 we plot curves (e) (f) (g) and (h) for the best first, linear combination in T-space with P=48, and PCA T-space with K_T=5 and 10, in Figure 2. Again we observe the same trend that the best first method can not give good performance. The linear combination method gave a slightly worse performance than the case with K_T=10 when the amount of adaptation data was limited to 2 utterances.

It is observed that because some noise types in the testing conditions are not used in simulating the environment space, the best first method can not identify the correct noise type, but has the potential to locate the noisy environment at the correct SNR level. Best first method still gave a slight error rate reduction with an average digit error rate of 29.76% with 10 adaptation utterances, as compared to the baseline error rate of 38.81%.

4.3. Comparison with supervised MLLR adaptation

Table 1 shows experiments conducted with conventional MLLR [10] to do environment adaptation, using 6 sets of MLLR matrices corresponding to 6 different manners of articulation, namely vowel, fricative, stop, nasal, approximant and silence. The transformation matrices were estimated directly with the adaptation utterances, and used to adapt HMM parameters for the testing environment. Taking the results obtained with the IEM and TEM approaches, with K_I and K_T set to 10, we listed in Table 1 the digit error rates across 4 SNR levels, from 5dB to 20dB, for supervised MLLR, curve (d) for IEM in Figure 1, and curve (h) for TEM in Figure 2, all with 2 adaptation utterances. The CMS results were also listed for comparison.

It can be noted that both IEM and TEM approaches achieved better performance than the baseline and MLLR, and clearer

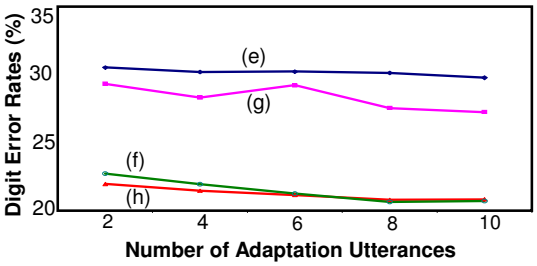

Figure 2: *Comparison of TEM approaches with different online processes. Curves (e), (f), (g) and (h) are for the best first, linear combination and PCA with K_T=5 and K_T=10.*

improvements were observed in lower SNR conditions. For example, when SNR=5dB, the error rate reductions from CMS to IEM and TEM were 46.27% and 46.76%, respectively. And the error rate reductions from MLLR to IEM and TEM were in the range of 10-11%. It is also noted that to characterize the testing environment the TEM method here uses 6 sets of matrices, and each matrix has the same dimension as those used in MLLR. Nonetheless TEM only estimated 6 sets of coefficients to determine the transformation matrices. The number of free parameters for MLLR was 6×(39+39)=468, and only 6×10=60 when K_T=10 for TEM. The realization of this data reduction is made by the *a priori* environment information when building the T-environment space.

Table1: *Digit error rates (in %) across different SNR levels.*

SNR (dB)	CMS	MLLR	IEM	TEM
5	77.63	46.38	41.71	41.33
10	46.73	33.39	22.70	21.89
15	20.13	17.76	14.04	15.17
20	10.76	13.97	9.83	10.46

4.4. Unsupervised adaptation

In the previous sections, the super-vector for an unknown testing environment for either IEM or TEM is estimated in a supervised mode. Since *N*-best information has already been shown beneficial to accomplish unsupervised adaptation of speakers [11], we try to incorporate this *N*-best information to realize unsupervised adaptation with the TEM approach which produces nearly the best performance among all supervised adaptation experiments. Here the adaptation utterances are still provided but the corresponding labels are multiple *N*-best strings. When these *N* strings are integrated, the *N*-best TEM formulation with PCA can be modified to the following:

$$\tilde{v}_{k,Nbest} = \arg\max_{v_k} \sum_{n=1}^{N} L(\mathbf{O}_{test}, q_n \mid (\sum_{k=1}^{K_T} v_k \, \mathbf{e}_{\mathbf{W}k}) \mathbf{X}_{clean}), \quad (13)$$

where $\tilde{v}_{k,Nbest}$ is the weighting coefficient for the *k*-th eigen-vector and q_n is the decoded label string for the *n*-th best list.

It is noted that when compared with TEM in a supervised mode with 2 adaptation utterances, the unsupervised solution achieved similar performance among different SNR levels. At 5dB SNR, the unsupervised adaptation TEM has a higher digit error rate of 47.16%, as compared to 41.33% for supervised TEM. It is interesting to note when SNR=15dB, unsupervised TEM gave a 14.20% digit error rate, which is slightly better than 15.17% achieved in supervised TEM.

We also compared TEM and MLLR in an unsupervised MLLR mode. With *N*-best list from 2 decoding utterances, the unsupervised TEM produced a better error rate than MLLR with a relative digit error reduction of 16% (from 56.56% to 47.16%) and 46% (from 26.28% to 14.20%) in 5dB and 15dB SNR conditions, respectively. When there were 10 decoding utterances, unsupervised TEM and MLLR by incorporating *N*-best lists had similar error rates in all the different conditions.

It should be noted that although both unsupervised TEM and MLLR achieved similar performance, the performance of unsupervised MLLR was seriously degraded with not enough *N*-best decoding utterances. TEM, on the other hand, efficiently utilizes *N*-best information in performing rapid adaptation. It can be concluded that either in a supervised or unsupervised mode, the TEM algorithm is very effective and can be used in many adverse conditions to improve performance.

5. Summary

We propose a new vector space approach to environment modeling. It showed good properties in characterizing the environment spaces of interest. IEM and TEM approaches have been adopted to estimate unknown testing environments in both supervised and unsupervised modes. They have reduced recognition errors significantly in adverse conditions. Improvements are more pronounced in lower SNR conditions. At 5dB SNR, TEM in both supervised and unsupervised modes achieves error rate reduction of more than 40% from the CMS baselines. When compared to a conventional MLLR, 10-11% error rate reductions are achieved with 2 adaptation utterances. The proposed unsupervised IEM and TEM approaches can be adopted because they achieve comparable improvement with those obtained supervised scenarios.

6. Acknowledgements

This work was partially supported under the Texas Instrument Leadership University grant.

7. References

[1] Acero, A., "Acoustical and environmental robustness in Automatic Speech Recognition," *Ph.D. Dissertation*, ECE, Department, CMU, Sept. 1990.

[2] Sankar, A. and Lee, C.-H., "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. on Speech and Audio Proc*., Vol. 4, pp.190-202, May.1996.

[3] Jolliffe, I. T., *Principal Component Analysis*. Berlin, Germany: Springer-Verlag, 1986.

[4] Metropolis, N. and Ulam, S., "The Monte Carlo method," *JASA*, Vol. 44, pp.335- 341, Sept. 1949.

[5] Garofolo, J. S. *et al*., "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus," U.S. Dept. of Commerce, NIST, Gaithersburg, MD, February 1993.

[6] ETSI standard document, "Speech processing, transmission and quality aspects (STQ); Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithm", *ETSI* ES 201 108 v1.1.2 (2000-04). 2000.

[7] Varga, A. P., Steeneken, H. J. M., Tomlinson, M., and Jones, D., "The NOISEX-92 study on the effect of additive noise on automatic speech recognition," *Tech. Rep*., 1992.

[8] Leonard, R. G., "A Database for Speaker-Independent Digit Recognition," *Proc. ICASSP*, 1984.

[9] Hirsch, H. G. and Pearce, D., "The Aurora experimental framework for the performance evaluations of speech recognition systems under noisy conditions," *ISCA ITRW ASR2000,* Sept. 2000.

[10] Leggetter, C. and Woodland, P., "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Compute. Speech Language*, vol. 9, pp.171-185, 1995.

[11] Nguyen, P., Gelin, P., Junqua, J.-C. and Chien, J.-T., "N-best based supervised and unsupervised adaptation for native and non-native speakers in cars," *Proc. ICASSP '99*, Vol. 1, pp.173-176, March 1999.