MLLR Transforms as Features in Speaker Recognition

Andreas Stolcke *Luciana Ferrer** *Sachin Kajarekar* *Elizabeth Shriberg* *Anand Venkataraman*

Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA
*Department of Electrical Engineering, Stanford University, Stanford, CA, USA
{stolcke,lferrer,sachin,ees,anand}@speech.sri.com

Abstract

We explore the use of adaptation transforms employed in speech recognition systems as features for speaker recognition. This approach is attractive because, unlike standard frame-based cepstral speaker recognition models, it normalizes for the choice of spoken words in text-independent speaker verification. Affine transforms are computed for the Gaussian means of the acoustic models used in a recognizer, using maximum likelihood linear regression (MLLR). The high-dimensional vectors formed by the transform coefficients are then modeled as speaker features using support vector machines (SVMs). The resulting speaker verification system is competitive, and in some cases significantly more accurate, than state-of-the-art cepstral gaussian mixture and SVM systems. Further improvements are obtained by combining baseline and MLLR-based systems.

1. Introduction

Current speaker recognition systems employ a combination of knowledge sources, but the basis of most state-of-the-art systems is still the modeling of cepstral features extracted over short time spans (a few tens of milliseconds) and modeled as an unordered set of independent samples. The modeling is typically carried out in terms of log-likelihood ratios of Gaussian mixtures [1], or discriminatively using support vector machines (SVMs) [2]. There are two fundamental problems with this approach. First, it ignores longer-term and higher-order structure in the speech, such as is best described at the level of phones, syllables, words, and whole utterances. Consequently, there have been numerous recent developments to characterize speaker idiosyncrasies at those levels, and state-of-the-art systems now typically employ a combination of long-term and short-term features [3, 4].

The second fundamental problem with short-term cepstral modeling is that the overall cepstral distribution conflates speaker characteristics with other factors, principally, channel properties and the choice of words spoken. Standard signal processing and feature-level normalization methods can alleviate some of the channel effects, and score-level normalization techniques such as HNORM [1] and TNORM [5] partially compensate for both sources of extraneous variability. Phone-conditioned (see [6] for an overview) and word-specific [7] cepstral models are a direct attempt to make models invariant to the choice of words (since words by and large determine the phone sequence). However, these approaches have the drawback of fragmenting the data and requiring sufficiently accurate speech recognition. Other recent work has also tried to explicitly decompose cepstral variability by source and design filters that are optimized for the factors that are desirable for a given task (e.g., speaker versus speech recognition) [8].

Although the speaker modeling approach proposed here is also based on cepstral features, it was motivated and enabled by our work on higher-level stylistic features, which typically require the use of large-vocabulary word recognition systems. Such systems use elaborate forms of adaptation to turn the speaker-independent recognition models into more accurate speaker-dependent models. Instead of modeling cepstral observations directly, we can model the "difference" between the speaker-dependent and the speaker-independent models. This difference is embodied by the coefficients of an affine transform of the Gaussian means in the recognition models. These transforms apply to models that are specific not only to phones, but to context-dependent phones (triphones). Thus, to the extent that the triphone-conditioned recognition models are independent of the choice of words, so are the speaker-specific transforms. Because the transforms themselves are shared among triphones (and to some extent also between phones), we avoid the problem of data fragmentation. We can thus represent the cepstral observations in a feature space of fixed, and relatively low, dimensionality. Furthermore, as we will show, the transform features lend themselves quite well to discriminative modeling with SVMs.

In the remainder of the paper we describe the details of our approach and explore several variants that arise in its implementation. We test the method on several speech databases, including the 2004 NIST speaker recognition evaluation (SRE) dataset, and compare its performance to that of standard cepstral models. Finally, we give results for combinations of the various models.

2. Method

2.1. Recognition system

Our speech recognition system is a fast, two-stage version of SRI's conversational telephone speech (CTS) system, as originally developed for the 2003 DARPA Rich Transcription evaluation [9] and later modified for the NIST 2004 speaker recognition evaluation [3]. The system performs a first decoding using Mel frequency cepstral coefficient (MFCC) acoustic models and a bigram language model (LM), generating lattices which are then rescored with a higher-order LM. The resulting hypotheses are used to adapt a second set of models based on perceptual linear prediction (PLP) acoustic features. The adapted models are used in a second decoding pass that is constrained by trigram lattices, which generates N-best lists. These are then rescored by a 4-gram LM and prosodic models to arrive at the final word hypotheses. The whole system runs in about 3 times real time on a hyperthreaded 3.4 GHz Intel Xeon processor.

2.2. Speaker adaptation transforms

In maximum likelihood linear regression (MLLR) [10], an affine transform (A, b) is applied to the Gaussian mean vectors to map from speaker-independent (μ) to speaker-dependent (μ') means: $\mu' = A\mu + b$. In unsupervised adaptation mode, the transform parameters (coefficients) are estimated so as to maximize the likelihood of the recognized speech under a preliminary recognition hypothesis. For a more detailed adaptation, the set of phone models can be partitioned or clustered by similarity, and a separate transform is applied to each cluster.

In our system, MLLR is applied in both recognition passes. The first pass is based on a phone-loop model as reference, and uses three transforms, for nonspeech, obstruent, and nonobstruent phones, respectively. The second decoding pass uses a more detailed MLLR scheme, based on word references generated by the first pass, and nine different transforms corresponding to phone classes for nonspeech, voiced/unvoiced stops, voiced/unvoiced fricatives, high/low vowels, retroflex phones, and nasals.

2.3. Speaker-adaptive training

A variant of MLLR estimates transforms that apply to both Gaussian means and variances (constrained MLLR or CMLLR) [11]. The advantage of this approach that is it can be equivalently carried out by transforming the input features, rather than the model parameters. This makes it easier to apply the transforms on both training and test data, thus yielding models that normalize out training speaker variability, an approach known as feature-space MLLR (fMLLR) or speaker-adaptive training (SAT) [12]. The SRI system uses a single such transform (applied to all frames/phones) in the second decoding pass. This feature-space transform applies before the more detailed model-space transforms described above.

2.4. Feature extraction and SVM modeling

The coefficients from one or more adaptation transforms are concatenated into a single feature vector and modeled using support vector machines. The data used is from conversational telephone speech, and each conversation side is processed as a unit by the speech recognition system. Consequently, each conversation side produces a single set of adaptation transforms pertaining to the same speaker, and hence a single feature vector. Since our acoustic features (after dimensionality reduction) contain 39 components, the number of SVM feature components will equal the number of transforms $\times 39 \times 40$. In cases where the adaptation scheme uses a separate transform for nonspeech models, that transform is left out of the feature vector, since it is not expected to help in speaker recognition.

An SVM is trained for each target speaker using the feature vectors from a background training set as negative examples (of which there are many, typically in the thousands), and the target speaker training data as positive examples (of which there are few, typically 1 or 8). To compensate for the severe imbalance between the target and background data, we adopted a cost model [13] to weight the positive examples 500-fold with respect to the negative examples. Throughout, a linear inner-product kernel function was used for SVM training.

We also found it advantageous to normalize the dynamic ranges of the feature vector components. This is necessary because the SVM kernel function is sensitive to the magnitude of the feature values, and hence to the relative weighting of feature dimensions. In the absence of prior information, a normalization procedure that roughly equates the dynamic ranges of feature components seems appropriate. We have had good success with two simple normalization methods. One is Z-normalization, which subtracts the means and divides by the standard deviations along each feature dimension. Another method is rank normalization, which replaces each feature value by its rank (normalized to the interval $[0, 1]$, i.e., the percentile) in the background distribution. Rank normalization performs an adaptive rescaling of the features to obtain an approximately uniform distribution. Rank normalization is computationally more expensive, but was found to work best in general; it was used in all reported experiments unless noted otherwise.

Table 1: Data sets used in experiments

Test set	SWB-II		Fisher	SRE-04	
Training	1-side	8-side	1-side	1-side	8-side
Conv. sides	3642	3058	734	1384	2695
Models	578	546	734	479	225
Trials	9765	4911	16578	15317	7336

Table 2: Speaker verification results using MLLR features. The top number (in italics) in each table cell is the EER (%). The bottom number is the minimum DCF value. The 1st-stage MLLR system uses Z-normalization on features, all other systems use rank-normalization.

	SWB-II		Fisher	SRE-04	
Features	1-side	8-side	1-side	1-side	8-side
1st stage MLLR	*6.85*	*1.87*	*6.37*	*12.38*	*6.12*
(2 transforms)	.25060	.07429	.09934	.41594	.19934
2nd stage MLLR	*4.75*	*1.23*	*5.57*	*9.49*	*4.96*
(8 transforms)	.1544	.04619	.08281	.33182	.18249
1st + 2nd MLLR	*4.33*	*1.28*	*5.50*	*8.92*	*4.52*
(10 transforms)	.14318	.04288	.07818	.30949	.14987

3. Experiments and Results

3.1. Datasets

We tested our baseline and MLLR-based systems on three databases: a subset of the NIST SRE-03 (Switchboard-II phase 2 and 3) data set, a selection of the Fisher collection conversations, and the NIST SRE-04 (Mixer) data. For Switchboard-II and SRE-04, two data sets were available, for training on 1 and 8 conversation sides, respectively. Table 1 summarizes the statistics of these data sets. The Switchboard-II trials were a subset of those used in the NIST SRE-03 evaluation, but had difficulty comparable to the full evaluation set, as measured by the performance of our baseline system.

The background training set consisted of 1553 conversation sides from Switchboard-II and Fisher that did not occur in (and did not share speakers with) any of the test sets, and that had duplicate speakers removed.

All data was processed identically by SRI's speech recognition system as described above. None of the test or background data were used in training or tuning of the recognition system.

In addition to feature-level normalization, we performed TNORM score-level normalization [5] in all experiments, including for the baseline systems.

3.2. MLLR system results

We first tested systems based solely on the model adaptation transforms employed in the first and second recognition stages of our systems. The first stage uses two speech transforms, yielding a 3120-dimensional feature vector. The second stage

Table 3: Speaker verification results using baseline, MLLR, and combined systems. The MLLR SVM system uses 10 transforms (same as last row in Table 2).

	SWB-II		Fisher	SRE-04	
System	1-side	8-side	1-side	1-side	8-side
MFCC GMM	*4.63* .17857	*1.92* .08353	*4.57* .10259	*7.77* .31126	*4.95* .21146
MFCC SVM	*5.82* .22088	*1.49* .05821	*5.43* .13693	*9.48* .38951	*4.22* .16748
MLLR SVM	*4.33* .14318	*1.28* .04288	*5.50* .07818	*8.92* .30949	*4.52* .14987
MFCC GMM +MLLR SVM				*6.04* .26537	*3.64* .13088
MFCC SVM +MLLR SVM				*6.89* .28271	*3.64* .11739
MFCC GMM +MFCC SVM				*7.17* .32338	*4.23* .17173

uses eight speech transforms, yielding a 12480-dimensional feature vector. We can also concatenate both these sets of transforms into a single 15600-dimensional feature. Table 2 summarizes the results in terms of both minimum detection cost function (DCF) and equal error rate (EER). DCF is the Bayesian risk function defined by NIST with $P_{\text{target}} = 0.1$, $C_{\text{fa}} = 1$, and $C_{\text{miss}} = 10$.

The results with 8-transform MLLR features are competitive with the best reported results for cepstral systems (cf. results in next section). Surprisingly good results are achieved by the 2-transform MLLR system, which uses only a simple phone-loop reference hypothesis, i.e., it does not rely on word recognition search. Finally, a consistent improvement over the 8-transform system is obtainable by concatenating the two feature vectors (using 10 transforms per speaker), showing that the two features are not entirely redundant. This may be in part because the two recognition stages (and corresponding MLLR) are based on different front-end features (MFCC versus PLP).

We also tried to optimize the number of transforms used in the second adaptation stage, since initially 8 just happened to be the value that was found to work best for speech recognition. However, no further improvement was obtained by either collapsing or refining the phone classes, indicating that the optimal choices for speech recognition and speaker recognition must be quite similar.

3.3. Baseline system combination

We compared the 10-transform MLLR system to two state-of-the-art cepstral systems. The first baseline system is a Gaussian mixture model (GMM) with universal background model (UBM) [1], based on 13 MFCCs (without C0) and first-, second-, and third-order difference features. The features are mean-subtracted and modeled by 2048 mixture components. Gender-handset models are adapted from this model and used for feature transformation [14]. The final features are mean and variance normalized at the utterance level. The detection score is the target/UBM likelihood ratio after TNORM.

The second baseline system is also based on MFCCs (with first- and second-order differences), followed by the same feature transformation and normalization steps. The final features are then modeled with SVMs utilizing the polynomial sequence kernel proposed by [2]. This baseline system shares with the MLLR system the advantages of discriminative training and classification afforded by the SVM framework, but uses essentially the same features as the more traditional GMM-UBM system.[1]

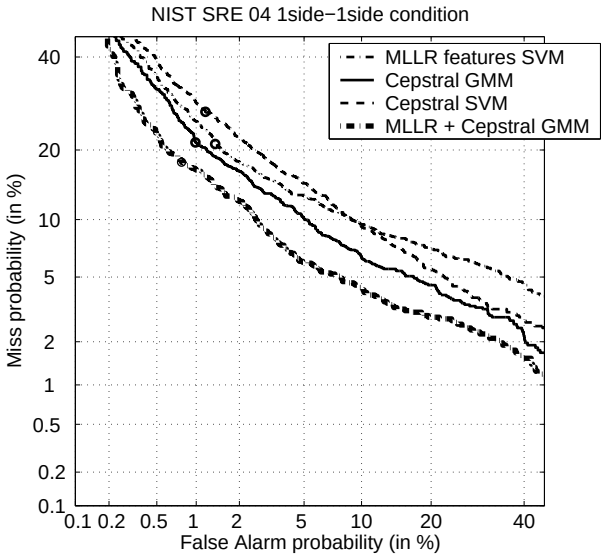

Figure 1: Detection error tradeoff (DET) curves for baseline, MLLR, and combined systems.

Finally, we performed a pairwise score-level combination of all three systems (baseline and MLLR), using a neural network trained to minimize DCF. The results are summarized in Table 3, and the detection error tradeoffs for a subset of the systems are plotted in Figure 1. The combination systems are evaluated only on the SRE-04 data sets since the other sets had been used to tune the combiner itself.

The results show that the MLLR-based system is consistently better than the cepstral systems on the DCF metric. The EER of the MLLR system is competitive, but on some test sets somewhat higher than the better of the cepstral systems. Furthermore, combination of one of the baseline systems with the MLLR system yields DCFs that are between 15% and 38% lower than the corresponding baseline results. EERs are reduced by 14% to 27% relative in the combination. By contrast, a combination of the two baseline systems yields a much smaller error reduction over the individual baselines, showing that system combination *per se* is not sufficient to obtain optimal results, and that the MLLR system contributes information that complements the baselines.

3.4. SAT transform features

One adaptation transform we have ignored so far is the feature (speaker-adaptive training, SAT) transforms employed in the second stage of our recognition system. SAT uses a single transform that operates on speech and nonspeech frames alike, so it is not clear what its role in speaker modeling should be. To answer this question, we built three variants of our 8-transform system. The first system uses the approach followed so far, i.e., the 8 model transforms apply after the features have been SAT-normalized, and the information in the SAT transforms is ignored. The second system estimates model transforms on features that have not been transformed (no SAT in recognition). This might be better if the SAT transform removes variability that is useful for speaker identification. The final variant sys-

[1] Note the cepstral SVM system used here is a standard one [2], not the enhanced version used by SRI in the 2005 NIST SRE.

Table 4: Speaker verification results on Fisher data, using SAT feature transforms.

Features	Fisher 1-side
8-transform MLLR after SAT norm	*5.50* .08150
8-transform MLLR without SAT norm	*5.64* .08483
8-transform MLLR + 1-transform SAT	*5.50* .08020

Table 5: Speaker verification results using MLLR and SAT transforms.

	Fisher	SRE-04	
Features	1-side	1-side	8-side
MLLR (10 transforms)	*5.50* .07818	*8.92* .30949	*4.52* .14987
MLLR + SAT (11 transforms)	*5.50* .07686	*9.07* .31287	*4.96* .17790

tem uses the SAT transform-normalized features, but concatenates the feature transform coefficients and the 8 standard model transform features into one SVM feature vector. (This diagnostic experiment was carried out using only Fisher background and test data, so the numbers are not directly comparable to the results reported earlier.)

Table 4 summarizes the results. Comparing the first two systems, we can observe that removal of SAT feature normalization from the MLLR-only system does *not* improve the speaker modeling, or conversely, that the SAT normalization does not remove information from the MLLR system that is useful in speaker modeling. Furthermore, comparison of the first and last systems shows that explicit inclusion of SAT transforms as feature vectors gives only marginally improved speaker models.

A final result that clarifies the role of SAT in relation to MLLR appears in Table 5. Here we added the SAT transform (as a feature vector) to the best MLLR system, thus obtaining a total of 11 transforms. We see that SAT features improve the system on Fisher data (only in DCF), but degrade accuracy on the SRE-04 data set. Recall that SRE-04 data does not appear in the background training set, whereas Fisher data does.

The tentative conclusion we can draw from these diagnostic experiments is that SAT features are largely determined by channel and corpus mismatch between the reference models and the test data. They are thus not generally useful for speaker modeling purposes, for which most of the relevant information is found in the MLLR transforms. This is also consistent with the earlier result that SAT normalization prior to MLLR improves speaker verification accuracy.

4. Conclusions and Future Work

We have proposed a speaker recognition approach based on SVM modeling of the speaker adaptation transforms found in modern speech recognition systems. By combining MLLR transforms for multiple recognition stages and phone classes we obtain a system that rivals or exceeds the accuracy of state-of-art speaker verification with frame-cepstrum features and GMM or SVM modeling. Furthermore, the MLLR system gives additional gains in combination with cepstral systems. We also found that the feature-level normalization in the recognizer seems to be helpful in removing variability due to source and channel.

We have yet to optimize the recognizer as a feature extractor for speaker recognition purposes. The present good results are achieved with features that are by-products of a system that was tuned for word recognition accuracy. It is quite possible that some of the other normalizations used (such as for vocal tract length) are in fact detrimental to speaker recognition.

5. Acknowledgments

We thank our colleagues Kemal Sonmez, for useful discussions, and Ramana Gadde, for clarifications of the MLLR implementation. This work was funded by a DoD KDD award via NSF IRI-9619921. The views herein are those of the authors and do not reflect the views of the funding agencies.

6. References

[1] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[2] W. M. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition," in *Proc. ICASSP*, Orlando, FL, May 2002, vol. 1, pp. 161–164.

[3] S. S. Kajarekar, L. Ferrer, E. S. K. Sonmez, A. Stolcke, A. Venkataraman, and J. Zheng, "SRI's 2004 NIST speaker recognition evaluation system," in *Proc. ICASSP*, Philadelphia, Mar. 2005, vol. 1, pp. 173–176.

[4] D. Reynolds, W. Campbell, T. Gleason, C. Quillen, D. Sturim, P. Torres-Carrasquillo, and A. Adami, "The 2004 MIT Lincoln Laboratory speaker recognition system," in *Proc. ICASSP*, Philadelphia, Mar. 2005.

[5] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, no. 1-3, Jan. 2000.

[6] A. Park and T. J. Hazen, "ASR dependent techniques for speaker identification," in *Proc. ICSLP*, J. H. L. Hansen and B. Pellom, Eds., Denver, Sept. 2002, pp. 1337–1340.

[7] K. Boakye and B. Peskin, "Text-constrained speaker recognition on a text-independent task," in *Proceedings Odyssey-04 Speaker and Language Recognition Workshop*, Toledo, Spain, May 2004.

[8] S. Kajarekar, N. Malayath, and H. Hermansky, "Analysis of speaker and channel variability in speech," in *Proceedings IEEE Workshop on Speech Recognition and Understanding*, Keystone, Colo., Dec. 1999.

[9] A. Stolcke, H. Franco, R. Gadde, M. Graciarena, K. Precoda, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, Y. Huang, B. Peskin, I. Bulyko, M. Ostendorf, and K. Kirchhoff, "Speech-to-text research at SRI-ICSI-UW," in *DARPA RT-03 Workshop*, Boston, May 2003, http://www.nist.gov/speech/tests/rt/rt2003/spring/presentations/sri+-rt03-stt.pdf.

[10] C. Leggetter and P. Woodland, "Maximum likelihood linear regress ion for speaker adaptation of HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

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

[12] H. Jin, S. Matsoukas, R. Schwartz, and F. Kubala, "Fast robust inverse transform SAT and multi-stage adaptation," in *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, Feb. 1998, pp. 105–109, Morgan Kaufmann.

[13] K. Morik, P. Brockhausen, and T. Joachims, "Combining statistical learning with a knowledge-based approach—a case study in intensive care monitoring," in *Proceedings of the 16th International Conference on Machine Learning*, I. Bratko and S. Dzeroski, Eds., San Francisco, CA, 1999, pp. 268–277, Morgan Kaufmann.

[14] D. A. Reynolds, "Channel robust speaker verification via channel mapping," in *Proc. ICASSP*, Hong Kong, Apr. 2003, vol. 2, pp. 53–56.