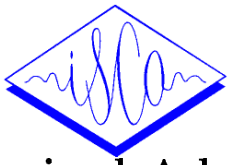

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

A Nonlinear Unsupervised Adaptation Technique for Speech Recognition

S. Dharanipragada *M. Padmanabhan*

IBM T.J. Watson Research Center
PO Box 218, Yorktown Heights, NY, 10598, USA
{dsatya,mukund}@watson.ibm.com

ABSTRACT

This paper describes a computationally inexpensive, nonlinear feature transformation technique for rapid adaptation of a speech recognition system to new acoustic conditions. One of the advantages of the method is that it does not require any initial decoding of the adaptation data for computing the nonlinear transform. This technique performs as well as the more expensive unsupervised MLLR technique. Furthermore, it significantly adds to the improvement when combined with unsupervised MLLR.

1. Introduction

A real-world speech recognition system encounters several acoustic conditions in the course of its application. Currently, it is well known that a system trained only for a particular acoustic condition degrades drastically when it encounters a different acoustic condition. One method of improving recognition accuracy in new conditions during run-time is by identifying a transform that brings the new speech/feature vectors close to the speech/feature vectors seen during training – a technique often called "adaptation via feature space transformation." This paper describes a computationally inexpensive, nonlinear feature transformation technique for rapid adaptation of a speech recognition system to new acoustic conditions. One of the advantages of the method is that it does not require any initial decoding of the adaptation data for computing the nonlinear transform. This technique performs as well as the more expensive unsupervised MLLR technique. Furthermore, it significantly adds to the improvement when combined with unsupervised MLLR.

Earlier techniques to adapt the acoustic models to a specific environment may be roughly classified into "model transformation" and "feature space transformation" techniques. In these techniques, the test utterance is first decoded with a generic speaker independent system (first pass), and the (errorful) transcription is used to compute the extent of the mismatch between the generic model and the specific environment. A specific example of "model transformation"is MLLR [1], which is based on the assumption that the model that is most suitable for transcribing the test speech is related to the generic model by means of a linear transform (i.e. the means and covariances of the gaussians in the transformed model are related to the means and covariances of the gaussians in the generic model by a linear transform). The parameters of the transformation are computed so that the likelihood of the test speech is maximized with the use of the transformed system, and assuming that the first pass transcription is the correct transcription of the test speech. In "feature space transformation" techniques, the feature space of the test utterance is assumed to be related to the generic feature space through a linear transformation, [2] and the linear transformation is computed, as before, to maximize the likelihood of the test speech under the assumption that the first pass transcription is correct. Other techniques to implement "feature space transformations" also exist - for instance, Probabilistic optimum filtering [3] and CDCN [4]. These techniques do not require a first pass decoding, but they do have the computational overhead of vector quantizing the acoustic space, and finding the center that is closest to each test feature vector.

The technique that we propose belongs to the category of "feature transformation." It has the advantage of being computationally much less expensive than the other techniques as it does not require a first pass decoding or a VQ computation. This non-linear map can be efficiently implemented using a simple table-lookup. It also represents a more powerful and flexible transformation as the mapping of the test feature to the space of the training features is not constrained to be linear.

Mathematically, the non-linear transform is obtained by maximizing a penalized likelihood of the transformed adaptation data using the training data density. It can be easily shown that the transformation that achieves this objective transforms the feature-space in such a way that the "cumulative" distribution function of the training acoustic data and test acoustic data are matched.

This paper is organized as follows. In the next section we give a mathematical formulation of the adaptation problem and provide solution to it. Section 3 we explore multiple transforms by combining with the MLLR

technique. Section 4 describes a practical approach to implementing the solution obtained in Section 2. Section 5 gives results.

2. Mathematical Formulation

Let us assume that the training data has a density p_0. Let $y_1, y_2, \cdots, y_N : y_i \in R^d$ be the adaptation vectors. Let p_y denote the density of the adaptation data. Let $f : R^d \rightarrow R^d$ be a transform, possibly nonlinear, and let $z_1, z_2, \cdots, z_N : z_i \in R^d$ be the transformed adaptation data, i.e., $z_i = f(y_i)$. Let p_z denote the density of the transformed data. Consider, the Kullback-Liebler distance between the densities p_z and p_0, which is a function of the transformation f:

$$L(f) = D(p_z|p_0) = \int_z p_z(z) \log p_z(z) dz - \int_z p_z(z) \log p_0(z) dz \quad (1)$$

Our goal is to obtain f^* that minimizes $L(f)$, i.e.,

$$f^* = \arg\min_f L(f). \quad (2)$$

$L(f)$ is minimized when $p_z(z) = p_0(z)$ for all $z \in R^d$, i.e., the transform must be selected in such a way that the density of the transformed data matches the training data density.

Interestingly, $L(f)$ can also be viewed as the negative of the penalized log-likelihood of the transformed data according to the training density. The penalty term, which is equal to the entropy of p_z prevents the optimal solution from being a trivial point mass distribution.

Finding a multidimensional transform that achieves the above is difficult in general. However, if we make the simplifying assumption of independence between the dimensions of the feature vectors then the density matching can be done for each dimension separately. One dimensional density matching is a widely applied technique for equalization and enhancement. In [5], this method was applied directly to the speech samples for the application of speaker identification. The transform is found as follows: Let r and s be two real-valued random variables with probability density functions $p_r(r)$ and $p_s(s)$ respectively. Let

$$h(r) = \int_{-\infty}^{r} p_r(w) dw \quad (3)$$

and

$$g(s) = \int_{-\infty}^{s} p_s(w) dw. \quad (4)$$

Both h and g, being cumulative distribution functions (CDFs), are single-valued and monotonically increasing in the interval $[0 \quad 1]$. If we define the transform $f = g^{-1}h$, then if $y = f(r)$, the probability density function of y, p_y, is equal to p_s.

3. Combining with MLLR

It is relatively straightforward to see that the above formulation and solution can be extended to obtain multiple class dependent nonlinear transformations. Another method to obtain class dependent adjustments is to apply MLLR after a global nonlinear transform is applied to the adaptation data. MLLR is a widely used model transformation technique [1]. In MLLR, the means, μ_j, of the output Gaussian distributions are modified by applying a linear transform, W_j and an offset, v_j, i.e.

$$\hat{\mu}_j = W_j \mu_j + v_j. \quad (5)$$

The transforms W_j and offsets v_j are chosen to maximize the likelihood of the adaptation data. A close examination of the equations determining the optimal linear transforms W_j and v_j in the MLLR technique will reveal that they are linearly related to the adaptation vectors, i.e.,

$$(W_j, v_j) = \mathcal{L}(z_1, z_2, \cdots, z_N) \quad (6)$$

But, since the z_i are related to y_i by a nonlinear transform we now have (W_j, v_j) related to the adaptation vectors in a nonlinear fashion.

4. Practical Implementation

Since we have sufficient training data, we can employ a non-parametric method, such as a histogram method, to determine the training data distribution. Since each dimension of the speech vector is considered independently, subscripts are dropped for ease of notation. First, the maximum and the minimum values across the whole training set, x_{max} and x_{min} are determined for each dimension. The range $[x_{min}\ ,\ x_{max}]$ is divided uniformly into M non-overlapping intervals or bins (usually equally spaced and typically about 10,000 in number); $x_{min} = b_1 < b_2 < \cdots b_{M+1} = x_{max}$ and bin $\mathcal{B}_i = [b_i\ ,\ b_{i+1})$. Next, a histogram is constructed on these bins using the entire training data-set. To do this, the entire training data is scanned and the number of samples that fall in each bin is counted. Let n_i be the number of samples in bin $\mathcal{B}_i$ and N be the total number of samples, i.e. the training data size. The probability of x being in bin $\mathcal{B}_i$ is approximated by :

$$p_0(x \in \mathcal{B}_i) = \frac{n_i}{N}. \quad (7)$$

Furthermore, for any $x \in \mathcal{B}_i$ the cumulative distribution function is approximated by:

$$g(x : x \in \mathcal{B}_i) = \sum_{j=1}^{i} \frac{n_j}{N} \quad (8)$$

which is a piece-wise constant function approximation of the true cumulative distribution function.

When enough adaptation data is available the same non-parametric histogram technique can be used to approximate the probability density of the adaptation data,p_y, along each dimension. The computation of the adaptation CDF, h, is identical to the training CDF computation, but with the adaptation speech vectors.

To efficiently implement the transformation the computations are organized as follows: First two tables (x_k, g_k) and (y_k, h_k) are constructed from g and h respectively in such a way that g_k and h_k take on values in [0 1] in equal increments (typically 1000 increments). Combining these tables gives the map $\{y_k, x_k\}$, which is a piecewise constant approximation of the true transformation. Then for any value y, the closest value in y_k is found by a employing a simple binary search and the corresponding value from $\{x_k\}$ is used as the transformed value of y.

5. Experimental Results

We experimented with this technique to compensate for the mismatch between handset and speaker-phone telephone data. The baseline system was trained using handset data and as expected this performs relatively poorly on the speakerphone data. The training CDF comprises of the CDF of the handset features and the test CDF comprises of the CDF of the speakerphone features. These CDFs can be computed on a speaker dependent basis or a speaker independent basis and we experimented with both techniques. For the speaker dependent case, to ensure that there is no dependence on the phonetic content of the test and training data, no stereo data was used i.e., the training and test CDF's were computed using different sentences from the same speaker recorded either on a speakerphone or handset.

5.1. System Description

All experiments were conducted on the IBM rank-based LVCSR system. The IBM LVCSR system uses context-dependent sub-phone classes which are identified by growing a decision tree using the training data and specifying the terminal nodes of the tree as the relevant instances of these classes [6, 7]. The training feature vectors are poured down this tree and the vectors that collect at each leaf are modeled by a mixture of Gaussian pdf's, with diagonal covariance matrices. Each leaf of the decision tree is modeled by a 1-state Hidden Markov Model with a self loop and a forward transition. Output distributions on the state transitions are expressed in terms of the rank of the leaf instead of in terms of the feature vector and the mixture of Gaussian pdf's modeling the training data at the leaf. The rank of a leaf is obtained by computing the log-likelihood of the acoustic

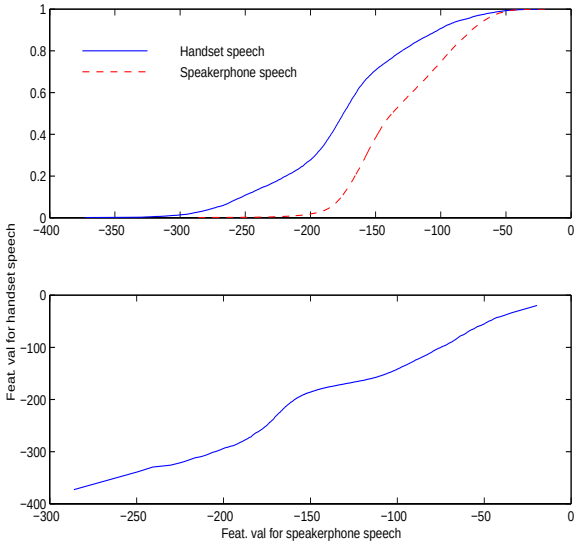

Figure 1: Cdf matching for the first cepstral dimension of handset and speaker phone features

vector using the model at each leaf, and then ranking the leaves on the basis of their log-likelihoods.

Speech was coded into 25 ms frames, with a frame-shift of 10 ms. Each frame was represented by a 39 component vector consisting of 13 MFCCs and their first and second time derivatives. Time derivatives and mean removal were performed after the nonlinear transformation. A set of SI models were trained on several hours of the telephone handset data. Overall, the decision tree had 2615 leaves. Each leaf had 15 Gaussian mixture components for the output distribution.

5.2. Experimental Set-up

The test data comprises of 25 sentences each (from the air travel domain) from 30 speakers recorded on speakerphones. The adaptation data comprises of two components: (1) a different set of 25 sentences from each speaker recorded on speakerphone and (2) another set of 25 sentences from each speaker recorded on a handset. Note that the nonlinear transformation technique uses both set of adaptation data whereas the other adaptation techniques that we benchmarked (unsupervised MLLR [1]) use just the first component.

Table 1 summarizes all the results. We obtain a relative improvement of 32.5% in word error rate with the nonlinear transformation technique using a speaker dependent reference CDF (SD-NL). We also note that this improvement is similar to that obtained with the unsupervised MLLR technique. When MLLR is applied

	Word Error (%)	Rel. Improv (%)
Baseline System	46.6	–
SD-NL	31.0	32.5
SI-NL	36.1	22.5
MLLR	31.6	31.1
SD-NL + MLLR	24.5	47.5
SI-NL + MLLR	26.5	43.1
2pass MLLR	29.8	36.1

Table 1: Word Error Rate comparisons on a speakerphone test set.

after the nonlinear transformation the WER reduction is 47.5%. When a speaker independent handset CDF (SI-NL) was used the improvement is reduced to 43%. One could view SD-NL+MLLR or SI-NL+MLLR as a two pass adaptation strategy. Hence we compare the results of these experiments with a two pass MLLR. It is clear from the table that both SD-NL+MLLR and SI-NL+MLLR give better results than performing a second iteration of MLLR which gives only a marginal improvement over the first pass.

Figure 2 summarizes the performance improvements as a function of the baseline error rate. From the figure it is clear that the MLLR technique is a more stable adaptation process since improves on all speakers, whereas with the nonlinear feature transformation technique, performance on some speakers gets worse.

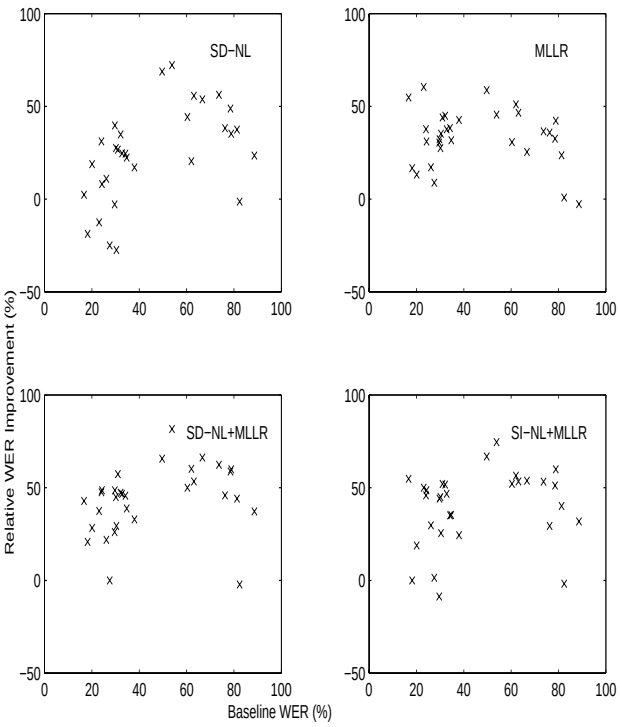

Figure 2: Relative WER improvements at different baseline WERs

6. Conclusion

We described a nonlinear feature transformation technique for adaptation of a speech recognition to new environments that is based on matching the training and adaptation densities. Our experiments show that this technique performs as well as the more expensive unsupervised MLLR technique. Furthermore, it significantly adds to the improvement when combined with the unsupervised MLLR technique.

References

1. C. J. Legetter and P. C. Woodland, "Maximum Likelihood linear regression for speaker adaptation of continuous density HMM's," in Comp. Speech Lang., vol9, pp. 171-186, 1996.
2. A. Sankar and C. H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," IEEE Trans. ASSP, 1995.
3. L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition," *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1994, pp 417-420.
4. F. H. Liu, A. Acero, R. M. Stern, "Efficient joint compensation of speech for the effect of additive noise and linear filtering," *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1992.
5. R. Balachandran and R. Mammone, "Non-parametric estimation and correction of non-linear distortion in speech systems," *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1998.
6. L.R. Bahl and P.V. deSouza and P.S. Gopalakrishnan and D. Nahamoo and M.A. Picheny, "Robust methods for context-dependent features and models in a continuous speech recognizer," in *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1994, pp. I-533-536.
7. P.S. Gopalakrishnan and L.R. Bahl and R. Mercer, "A tree search strategy for large vocabulary continuous speech recognition," in *Proc., Intl Conf. on Acoust., Speech, and Sig. Proc.*, 1995.