A NON-LINEAR SPEAKER ADAPTATION TECHNIQUE USING KERNEL RIDGE REGRESSION

George Saon

IBM T. J. Watson Research Center, Yorktown Heights, NY, 10598
e-mail: gsaon@us.ibm.com

ABSTRACT

We propose a non-linear model space transformation for speaker or environment adaptation based on weighted kernel ridge regression (KRR). The transformation is given by a generalized least squares linear regression in a kernel-induced feature space operating on Gaussian mixture model means and having as targets the adaptation frames. Using the "kernel trick", the solution to the optimization problem is obtained by solving a system of linear equations involving the Gram matrix of the input variables. We show that MLLR is a special case of KRR when a linear kernel is employed. Furthermore, we study an efficient low-rank approximation to the kernel matrix termed "rectangle method", where the regressors are chosen to be a small set of clustered adaptation frames. Experiments conducted on the EARS database (English conversational telephone speech) indicate that KRR with a Gaussian RBF kernel outperforms standard regression class-based MLLR.

1. INTRODUCTION

In recent years, there has been a surge of interest in the study of non-linear transformations for automatic speech recognition. These transformations are used to either alter the features as in [9], or the acoustic model parameters as in [12], in order to improve the discriminability among phonetic classes. A second common application of non-linear transforms is in the context of speaker adaptation where the goal is to warp the test data to match the characteristics of the training data [3, 11, 14]. Here, we distinguish between parametric and non-parametric techniques. One example of a non-parametric technique was introduced in [3] and consists in matching the overall cumulative distribution function (CDF) of the adaptation data to the CDF of the training data on a per dimension basis. This idea has been further developed in [11], where the CDFs for every training and test speakers are warped to the same Gaussian distribution. Among the parametric techniques for adaptation, we can mention the work of [14], where the authors compute a linear projection from a high dimensional Gaussian posterior space to the normal feature space similar to the fMPE transformation [9]. Unlike fMPE however, the projection is estimated using maximum likelihood.

The common denominator of these non-linear parametric methods, whether for discriminative acoustic model training or for adaptation, is the use of gradient descent procedures to solve the various optimization problems. While this may be feasible for training, where the transformation has to be estimated only once, it becomes costly in the case of speaker adaptation. One appealing property of maximum likelihood linear regression (MLLR) [6] is the simplicity of the solution. Indeed, each row of the MLLR matrix can be obtained by solving a single system of linear equations. The question that we address in this paper is the following: can we extend MLLR to a non-linear transformation while keeping the desirable property of having a closed-form solution ?

The observation that we are exploiting in this paper is that, for MLLR, both the optimization problem and the regression function can be expressed solely in terms of dot products between vectors (Gaussian means). Indeed, by writing one row of the transform as a linear combination of Gaussian means, the optimization can be written using linear combinations of inner products between vectors. We then can use a very popular technique in kernel-based machine learning called the "kernel trick" which consists in replacing those dot products with evaluations of an arbitrary kernel function. The result is a generalized least squares linear regression in a kernel-induced feature space and is termed kernel ridge regression or KRR [2, 10].

One potential drawback of KRR is the size of the optimization problem: we turn an n-dimensional problem with n being the dimension of the feature space into an ℓ-dimensional problem with ℓ being the number of samples (i.e. adaptation frames). This entails the use of regularization to avoid overfitting. In addition, we will have to employ approximation techniques for the kernel matrix.

The application of kernel methods to speech recognition is not new. In [7], the authors use kernel PCA for improved feature extraction. The work described in [4] applies kernel discriminant analysis for a similar purpose. A more closely related work to this paper is kernel eigenspace-based MLLR proposed in [8]. There, the authors use kernel PCA to derive a set of eigenmatrices from speaker-dependent MLLR matrices in the kernel-induced feature space. The main difference here is that we employ kernels in the transformations themselves as opposed to "kernelizing" the space of MLLR transformations.

The paper is organized as follows: in section 2, we outline the derivation of kernel ridge regression and highlight the connection with MLLR. In section 3, we present some experimental evidence of its utility followed by some concluding remarks in section 4.

2. KERNEL RIDGE REGRESSION

We are given a training set $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$, consisting of ℓ independent identically distributed samples drawn from some unknown joint probability distribution.

We consider the n-dimensional regression problem, where the ℓ training examples satisfy $\mathbf{x}_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$, for all i. Our goal is to learn a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ that approximates the training examples and that will generalize well on new examples. One way to measure the performance of f on S is to define a smooth loss function $L(y, f(\mathbf{x}))$ and to compute the empirical risk

$$L[f] = \sum_{i=1}^{\ell} L(y_i, f(\mathbf{x}_i)) \quad (1)$$

The problem of finding f which minimizes $L[f]$ is ill-defined because we have not specified the set of allowable functions. If we restrict the solution to lie in a bounded convex subset of a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ defined by a positive definite kernel function K, the following regularized problem is well-defined:

$$\min_{f \in \mathcal{H}} L[f] + \lambda ||f||_K^2 \tag{2}$$

where $||.||_K$ represents the norm of a function in $\mathcal{H}$ and λ is a regularization parameter. Recall that a positive definite kernel function $K : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ implements a dot product in some feature space i.e. there exists $\Phi : \mathbb{R}^n \rightarrow \Omega$ such that

$$K(\mathbf{x}, \mathbf{y}) = < \Phi(\mathbf{x}), \Phi(\mathbf{y}) > \tag{3}$$

with $< \cdot, \cdot >$ denoting the standard dot product in the kernel induced feature space Ω. Moreover, $\mathcal{H}$ is an RKHS if

$$< K(\cdot, \mathbf{x}), f >= f(\mathbf{x}), \qquad \forall f \in \mathcal{H} \tag{4}$$

In fact, $\mathcal{H}$ is a vector space containing all linear combinations of the functions $K(\mathbf{x}, \cdot)$ [5]

$$\mathcal{H} = \{f | f(\cdot) = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_i, \cdot)\} \tag{5}$$

Back to the problem at hand, the representer theorem[1] (Kimeldorf and Wahba, 1971) states that, if the loss function L is only pointwise dependent on f, the solution to (2) has the form

$$f^*(\mathbf{x}) = \sum_{i=1}^{\ell} c_i K(\mathbf{x}, \mathbf{x}_i) \tag{6}$$

Using (4) and (6), let us now derive the norm for functions in the above form

$$\begin{aligned} ||f||_K^2 &= < f(\cdot), f(\cdot) > \\ &= < \sum_{i=1}^{\ell} c_i K(\mathbf{x}_i, \cdot), f(\cdot) > \\ &= \sum_{i=1}^{\ell} c_i < K(\mathbf{x}_i, \cdot), f(\cdot) > \\ &= \sum_{i=1}^{\ell} c_i f(\mathbf{x}_i) \\ &= \sum_{i=1}^{\ell} c_i (\sum_{j=1}^{\ell} c_j K(\mathbf{x}_i, \mathbf{x}_j)) \\ &= \mathbf{c}^T \mathbf{K} \mathbf{c} \end{aligned} \tag{7}$$

where $\mathbf{K}$ now denotes the $\ell \times \ell$ Gram matrix whose (i, j)th entry is $K(\mathbf{x}_i, \mathbf{x}_j)$. We can rewrite (2) now as

$$\min_{\mathbf{c} \in \mathbb{R}^\ell} \sum_{i=1}^{\ell} L(y_i, \sum_{j=1}^{\ell} c_j K(\mathbf{x}_i, \mathbf{x}_j)) + \lambda \mathbf{c}^T \mathbf{K} \mathbf{c} \tag{8}$$

In this paper, we concentrate on the simple square loss function

$$L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2 \tag{9}$$

[1] A nice proof of which can be found in [5].

By converting everything to vector notation, (8) becomes

$$\min_{\mathbf{c} \in \mathbb{R}^\ell} (\mathbf{y} - \mathbf{K}\mathbf{c})^T (\mathbf{y} - \mathbf{K}\mathbf{c}) + \lambda \mathbf{c}^T \mathbf{K} \mathbf{c} \tag{10}$$

This is a convex differentiable function, so we can find the minimum simply by taking the derivative with respect to $\mathbf{c}$ and setting it to zero

$$\begin{aligned} & (\mathbf{y} - \mathbf{K}\mathbf{c})^T (-\mathbf{K}) + \lambda \mathbf{c}^T \mathbf{K} &=& \ 0 \\ \Leftrightarrow \quad & (\mathbf{K}\mathbf{c} - \mathbf{y}) + \lambda \mathbf{c} &=& \ 0 \\ \Leftrightarrow \quad & (\mathbf{K} + \lambda \mathbf{I}_\ell)\mathbf{c} &=& \ \mathbf{y} \end{aligned} \tag{11}$$

where we have made the tacit assumption that $\mathbf{K}$ is invertible. We see that the KRR problem can be solved by solving a single system of linear equations. Herein lies the beauty of this method: one can perform powerful non-linear least squares regression and still have a closed form solution.

2.1. Rectangle approximation method for KRR

Solving (11) has a complexity of $\mathcal{O}(\ell^3)$ which becomes rapidly intractable for a large sample size ℓ. One way around this is to limit the expansion in (6) to a much smaller number of non-zero coefficients, i.e.

$$f^*(\mathbf{x}) \approx \sum_{i=1}^{m} c_i K(\mathbf{x}, \mathbf{x}_i) \tag{12}$$

with $m \ll \ell$. However, we still compute the contribution of all the samples to the objective function. More precisely, we look for $\mathbf{c} \in \mathbb{R}^m$ which minimizes

$$\min_{\mathbf{c} \in \mathbb{R}^m} (\mathbf{y} - \mathbf{K}_{\ell m}\mathbf{c})^T (\mathbf{y} - \mathbf{K}_{\ell m}\mathbf{c}) + \lambda \mathbf{c}^T \mathbf{K}_{mm} \mathbf{c} \tag{13}$$

where $\mathbf{K}_{\ell m}$ and $\mathbf{K}_{mm}$ are respectively, $\ell \times m$ and $m \times m$ kernel matrices. A derivation similar to (11) leads us to the equation

$$(\mathbf{K}_{m\ell}\mathbf{K}_{\ell m} + \lambda \mathbf{K}_{mm})\mathbf{c} = \mathbf{K}_{m\ell}\mathbf{y} \tag{14}$$

which can be solved now in $\mathcal{O}(m^3)$. So far, we have not said anything about the choice of the $\mathbf{x}_i$'s in (12). The standard practice in kernel techniques which use the rectangle approximation method appears to be to choose a subset of size m of the original training set. We believe that a more judicious way would be to cluster the training set to m cluster centers and use those as $\mathbf{x}_i$'s. This opens up an interesting perspective: it is possible to opt for a set of regressors which are completely unrelated to the training samples, say $\mathbf{z}_1, \ldots, \mathbf{z}_m$, $\mathbf{z}_i \in \mathbb{R}^n$ and perform KRR on those. In particular, we use in our experiments regressors given by clustered adaptation frames even though the transform is applied to the Gaussian means.

2.2. Weighted KRR

It is sometimes helpful to give different weights to the contributions of different samples to the objective function. This is particularly pertinent for speaker adaptation where there is an inherent uncertainty in the alignment between adaptation frames and Gaussian means. Using matrix notation, weighted (or generalized) KRR can be formulated as

$$\min_{\mathbf{c}\in\mathbb{R}^\ell} (\mathbf{y}-\mathbf{K}\mathbf{c})^T\mathbf{W}(\mathbf{y}-\mathbf{K}\mathbf{c}) + \lambda \mathbf{c}^T\mathbf{K}\mathbf{c} \quad (15)$$

where $\mathbf{W} = diag(w_1, \ldots, w_\ell)$ is a diagonal weight matrix. Similarly, the rectangle method for weighted KRR leads to

$$\min_{\mathbf{c}\in\mathbb{R}^m} (\mathbf{y}-\mathbf{K}_{\ell m}\mathbf{c})^T\mathbf{W}(\mathbf{y}-\mathbf{K}_{\ell m}\mathbf{c}) + \lambda \mathbf{c}^T\mathbf{K}_{mm}\mathbf{c} \quad (16)$$

with the solution satisfying

$$(\mathbf{K}_{m\ell}\mathbf{W}\mathbf{K}_{\ell m} + \lambda \mathbf{K}_{mm})\mathbf{c} = \mathbf{K}_{m\ell}\mathbf{W}\mathbf{y} \quad (17)$$

Here we assumed that $\mathbf{K}_{m\ell}\mathbf{W}\mathbf{K}_{\ell m} + \lambda\mathbf{K}_{mm}$ is positive definite for the solution to be a minimum which imposes certain constraints on $\mathbf{W}$ (such as positive weights).

2.3. Connection with MLLR

MLLR is a form of generalized least squares linear regression. For a specific dimension d, the objective function for MLLR can be formulated as follows:

$$\min_{\mathbf{a}_d\in\mathbb{R}^n} \sum_{t=1}^{T}\sum_{j=1}^{N} \frac{\gamma_t(j)}{\sigma_{jd}^2}(o_{td} - < \mathbf{a}_d, \mu_j >)^2 \quad (18)$$

where $\mathbf{a}_d \in \mathbb{R}^n$ is row d of the MLLR matrix, $\mathbf{o}_1, \ldots, \mathbf{o}_T$ are adaptation frames, $(\mu_j, \mathbf{\Sigma}_j)$ are respectively, the mean and diagonal covariance matrix of Gaussian j, and $\gamma_t(j)$ is the posterior probability of mixture component j at time t. Defining

- $w_i = \frac{\gamma_t(j)}{\sigma_{jd}^2}$,
- $\mathbf{x}_i = \mu_j$,
- $y_i = o_{td}, i = (1,1)\ldots(t,j)\ldots(T,N)$

leads us to the familiar weighted regression equation

$$\min_{\mathbf{a}_d\in\mathbb{R}^n} \sum_{i=1}^{TN} w_i(y_i - < \mathbf{a}_d, \mathbf{x}_i >)^2 \quad (19)$$

Now, if $rank(\mathbf{x}_1, \ldots, \mathbf{x}_{TN}) = n$, for every $\mathbf{a}_d \in \mathbb{R}^n$ there exist $c_1, \ldots, c_{TN}$ such that $\mathbf{a}_d = \sum_{i=1}^{TN} c_i\mathbf{x}_i$. Plugging this back into (19), we have

$$\begin{aligned}
&\min_{\mathbf{a}_d\in\mathbb{R}^n} \sum_{i=1}^{TN} w_i(y_i - < \mathbf{a}_d, \mathbf{x}_i >)^2 &= \\
&\min_{\mathbf{c}\in\mathbb{R}^{TN}} \sum_{i=1}^{TN} w_i(y_i - < \sum_{j=1}^{TN} c_j\mathbf{x}_j, \mathbf{x}_i >)^2 &= \\
&\min_{\mathbf{c}\in\mathbb{R}^{TN}} \sum_{i=1}^{TN} w_i(y_i - \sum_{j=1}^{TN} c_j < \mathbf{x}_j, \mathbf{x}_i >)^2 &= \\
&\min_{\mathbf{c}\in\mathbb{R}^{TN}} \sum_{i=1}^{TN} w_i(y_i - \sum_{j=1}^{TN} c_j K(\mathbf{x}_j, \mathbf{x}_i))^2
\end{aligned} \quad (20)$$

where $K(\mathbf{x}_i, \mathbf{x}_j) =< \mathbf{x}_i, \mathbf{x}_j >$ is the standard dot product in $\mathbb{R}^n$. It follows that MLLR is a special case of weighted KRR with a linear kernel function and no regularization.

Two observations can be made. The first is that the complexity of KRR appears to be $\mathcal{O}((TN)^3)$. In practice however, only few Gaussian components have non-zero weights (i.e. posteriors) at a given time which reduces the complexity to $\mathcal{O}(T^3)$. In addition, the rectangle approximation method can lower the complexity to $\mathcal{O}(m^2T)$ for matrix multiplications and $\mathcal{O}(m^3)$ for system solving.

The second observation has to do with the regularization term. From a Bayesian perspective, the quantity $\mathbf{c}^T\mathbf{K}\mathbf{c}$ acts as a Gaussian prior on the expansion coefficients, that is

$$\mathbf{c} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}^{-1})$$

This is similar to maximum a posteriori linear regression (or MAPLR) [1] with the main difference being that, in MAPLR, the prior is applied directly to the rows of the transform.

3. EXPERIMENTS AND RESULTS

The experiments were conducted on the EARS database (English conversational telephone speech). The training data consists of 2300 hours of telephone conversations between two strangers on a preassigned topic. We based the experiments on our first-pass decoding setup from the RT'04 evaluation. The acoustic model uses a pentaphone context decision tree and comprises 149K 40-dimensional Gaussians which are discriminatively trained with MPE. The acoustic features are obtained by transforming every 9 consecutive 13-dimensional PLP cepstral frames through LDA and MLLT to a 40-dimensional space. More details about the system can be found in [13].

The test set consists of 36 two-channel conversations (72 speakers) totaling 3 hours of speech and 37.8K words released by NIST as the DEV'04 set. The amount of adaptation data per speaker is roughly 150 seconds suggesting a size of $\ell = 15000 \times 149000$ in the regression problems. In practice, only two Gaussians per frame are active on average because we only keep the pairs of frames and Gaussian means for which the posterior probability exceeds 0.1 (leading to $\ell = 27600$). The complexity is further reduced using the rectangle approximation method.

For MLLR, we use a regression tree obtained by a top-down clustering of the Gaussians to a depth of 5. The number of transforms is controlled by a minimum count threshold of 4000 frames per transform. For KRR, we cluster the adaptation frames using EM to a variable number of clusters per speaker. The number of clusters (m in the rectangle approximation method) is controlled by a minimum count threshold of 500 frames per cluster. Finally, we use equation (17) to compute the KRR transforms for various kernel functions. Similarly to fMPE [9], we don't actually update the means directly; instead we compute offsets to the means so that a null transform will leave the original means unaffected, i.e.

$$\hat{\mu} = \mu + \begin{bmatrix} f_1(\mu) \\ \cdots \\ f_n(\mu) \end{bmatrix} = \mu + \begin{bmatrix} \sum_{j=1}^m c_{1j}K(\mu, \mathbf{x}_j) \\ \cdots \\ \sum_{j=1}^m c_{nj}K(\mu, \mathbf{x}_j) \end{bmatrix} \quad (21)$$

In Table 1, we give a comparison of the various adaptation techniques. For the Gaussian RBF kernel, we set $\sigma = 100$ and $\lambda = 0.1$ for all the kernels. In the last line of Table 1, we used two regression classes for KRR (speech and non-speech). Interestingly, KRR with a quadratic kernel exhibits the same performance as linear KRR (and single-transform MLLR), whereas the exponential kernel outperforms the other methods. These results warrant studying the RBF kernel in more detail. This kernel function

Baseline		23.0%
MLLR	single transform	21.5%
MLLR	multiple transforms	21.3%
KRR	$K(\mathbf{x},\mathbf{y}) = \mathbf{x}\cdot\mathbf{y}$	21.5%
KRR	$K(\mathbf{x},\mathbf{y}) = (\mathbf{x}\cdot\mathbf{y}+1)^2$	21.5%
KRR	$K(\mathbf{x},\mathbf{y}) = (\mathbf{x}\cdot\mathbf{y}+1)^3$	21.7%
KRR	$K(\mathbf{x},\mathbf{y}) = exp(-\|\|\mathbf{x}-\mathbf{y}\|\|^2/\sigma)$	21.1%
KRR	same with 2 transforms (speech,silence)	20.8%

Table 1: Word error rates for MLLR and KRR on the DEV'04 test set.

implements a similarity measure between vectors: vectors close together will have a higher kernel value than those which are far apart. The degree of similarity is controlled by the kernel width σ. For very small widths, the kernel acts as a δ function outputting zero if the arguments are different. KRR with such a kernel will tend to leave the Gaussian means unchanged so we expect a performance closer to the baseline. On the other hand, a very flat kernel will behave like a linear kernel and KRR should exhibit a performance which is closer to MLLR. This intuitive behavior can be verified in Figure 1, where we plot the word error rate as a function of σ.

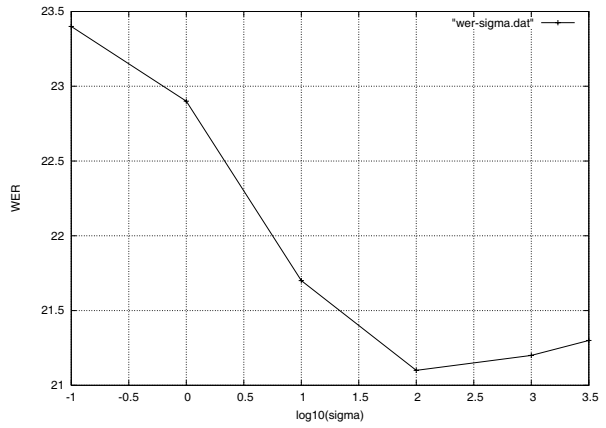

Figure 1: Word error rate as a function of $\log_{10}(\sigma)$ for KRR with a Gaussian RBF kernel.

4. CONCLUSION

In this paper, we discussed the applicability of kernel ridge regression to the problem of transform-based speaker adaptation. We started from the observation that MLLR is a form of generalized least squares linear regression and that both the function and the optimization can be expressed entirely in terms of dot products of the Gaussian means. We then applied the "kernel trick" and turned those dot products into arbitrary kernel function evaluations. This has the effect of embedding the input features into a kernel-induced feature space and performing linear regression in that space which, in turn, is equivalent to performing non-linear weighted least squares regression in the original space.

Experimental results on the EARS database suggest that KRR with an exponential kernel outperforms KRR with polynomial kernels and MLLR. Like for MLLR, the result can be further improved by using multiple regression classes and transforms for KRR. An open question which needs to be addressed is the same which plagues many other kernel-based machine learning techniques namely, the choice of the kernel. Indeed, so far we have not attempted to optimize the regressors or the kernel function. This remains the subject of future research.

5. REFERENCES

[1] C. Chesta, O. Siohan, and C.-H. Lee. Maximum a posteriori linear regression for hidden Markov model adaptation. In Proc. Eurospeech'99, Budapest, 1999.

[2] N. Cristianini and J.-S. Taylor. Support vector machines and other kernel-based learning methods. Cambridge University Press, 2000.

[3] S. Dharanipragada, M. Padmanabhan. A non-linear unsupervised adaptation technique for speech recognition. In Proc. ICSLP'00, Beijing, 2000.

[4] H. Erdogan. Subspace kernel discriminant analysis for speech recognition. ITRW on Robustness Issues in Conversational Interaction, Norwich, UK, 2004.

[5] M. I. Jordan. Advanced topics in learning and decision making. Course CS281B/Stat241B, U. Berkeley, 2004.

[6] C. J. Leggetter and P. C. Woodland. Speaker adaptation of HMMs using linear regression. Technical Report CUED/F-INFENG, Cambridge University Engineering Department, 1994.

[7] A. Lima, H. Zen, Y. Nankaku, C. Miyajima, K. Tokuda and T. Kitamura. On the use of KPCA for feature extraction in speech recognition. In Eurospeech'03, Martigny, Switzerland, 2003.

[8] B. Mak and R. Hsiao. Improving eigenspace-based MLLR adaptation by kernel PCA. In Interspeech'04, Jeju, Korea, 2004.

[9] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig. fMPE: Discriminatively trained features for speech recognition. In ICASSP'05, Philadelphia, PA, 2005.

[10] R. Rifkin, G. Yeo and T. Poggio. Regularized least-squares classification. Chapter 7 of Advances in learning theory: methods, models and applications, NATO Science series 3, IOS Press Amsterdam, 2003.

[11] G. Saon, S. Dharanipragada and D. Povey. Feature space Gaussianization. In ICASSP'04, Montreal, 2004.

[12] K.-C. Sim and M. Gales. Temporally varying model parameters for large vocabulary continuous speech recognition In Proc. Interspeech'05, Lisbon, 2005.

[13] H. Soltau, B. Kingsbury, L. Mangu, D. Povey, G. Saon, and G. Zweig. The IBM 2004 conversational telephony system for rich transcription. In ICASSP'05, Philadelphia, PA, 2005.

[14] K. Visweswariah and P. Olsen. Feature adaptation using projection of Gaussian posteriors. In Proc. Interspeech'05, Lisbon, 2005.