DIMENSIONAL REDUCTION, COVARIANCE MODELING, AND COMPUTATIONAL COMPLEXITY IN ASR SYSTEMS

Scott Axelrod, Ramesh Gopinath, Peder Olsen, Karthik Visweswariah

IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, USA
$\{$*axelrod,rameshg,pederao,kv1*$\}$*@us.ibm.com*

ABSTRACT

In this paper, we study acoustic modeling for speech recognition using mixtures of exponential models with linear and quadratic features tied across all context dependent states. These models are one version of the SPAM models introduced in [1]. They generalize diagonal covariance, MLLT, EMLLT, and full covariance models. Reduction of the dimension of the acoustic vectors using LDA/HDA projections corresponds to a special case of reducing the exponential model feature space. We see, in one speech recognition task, that SPAM models on an LDA projected space of varying dimensions achieve a significant fraction of the WER improvement in going from MLLT to full covariance modeling, while maintaining the low computational cost of the MLLT models. Further, the feature precomputation cost can be minimized using the hybrid feature technique of [2]; and the number of Gaussians one needs to compute can be greatly reducing using hierarchical clustering of the Gaussians (with fixed feature space). Finally, we show that reducing the quadratic and linear feature spaces separately produces models with better accuracy, but comparable computational complexity, to LDA/HDA based models.

1. INTRODUCTION

In this paper we study acoustic models for speech recognition which are mixtures of exponential models for acoustic vectors $x \in \mathbf{R}^d$ which use features tied across all states s of a context dependent Hidden Markov model. We look at systems with δ linear features ($\delta \leq d$) and D quadratic features ($D \leq d(d+1)/2$). These models were introduced in [1] under the acronym SPAM models because they are Gaussian mixture models with a subspace constraint placed on the model precisions (inverse covariance matrices) and means; although the precise condition on the means was left ambiguous in [1]. Reference [1] focused on the case of unconstrained means, in which the only constraint was that the precision matrices be a linear combination of matrices $\{S_k\}_{k=1}^{D}$ which are shared across Gaussians.

The SPAM models generalize the previously introduced EMLLT models [3, 4], in which the S_k are required to be rank one matrices. The well known maximum likelihood linear transform (MLLT) [5] or semi-tied covariance [6] models are the special case of EMLLT models when $D = d$.

Using the techniques developed in section 3 here and in [1, 2, 3, 4, 7], it is now possible to perform, at least to a good approximation, maximum likelihood training of these models for reasonably large scale systems, in both the completely general case and in a number of interesting subcases.

Our goal is to use these models as a tool to improve word error rates at reasonable computational cost. The time required for evaluating the acoustic model is

$$time = precompute + nGaussEvaled * perGaussian \ . \quad (1)$$

Here $precompute$ is the time required to precompute all of the linear and quadratic features; $nGaussEvaled$ is the actual number of Gaussians evaluated; and $perGauss$ is the amount of time required for each Gaussian evaluation, which, up to constants, is just $D + \delta$. If we were to evaluate all of the Gaussians in a system with very many Gaussians, the term $nGaussEvaled * perGaussian$ would very much dominate over the precomputation time. However, by clustering the Gaussians (preserving the fixed feature space), as discussed in 5, we are able to reduce $nGaussEvaled$ to the point where the precomputation time becomes a significant fraction of the overall computation.

The feature precomputation time can be reduced by either of two techniques. First, as discussed in section 4, one can reduce the effective dimensions of the samples x by generalizing the heteroscedastic discriminant analysis technique of [7]. Second, one can use the *hybrid* technique of [2] which restricts the matrices $\{S_k\}$ to be linear combinations of K rank one matrices, $K << d(d+1)/2$.

2. DEFINITION OF MODEL

The SPAM models have the form:

$$\begin{aligned} p(x|s) &= \sum_{g \in \mathcal{G}(s)} \pi_g p(x|g) \ , && (2) \\ p(x|g) &= \det\left(\frac{P_g}{2\pi}\right)^{1/2} e^{-\frac{1}{2}(x-\mu_g)^T P_g (x-\mu_g)} \ , && (3) \end{aligned}$$

where $\mathcal{G}(s)$ is the set of Gaussians for state s, and the precisions and means are written as

$$\begin{aligned} P_g &= S_0 + \sum_{k=1}^{D} \lambda_g^k S_k \ , && (4) \\ \theta_g &= P_g \mu_g = l_0 + \sum_{a=1}^{\delta} \phi_g^a \, l_a \ . && (5) \end{aligned}$$

The $d \times d$ symmetric matrices $\{S_k\}$ and the vectors $\{l_a\}$ in $\mathbf{R}^d$ are tied across all Gaussians. In the following, we will drop the affine shifts (i.e. set S_0 and l_0 to zero) when they don't need to be emphasized.

The constraints (4) and (5), correspond to restricting to the following linear and quadratic features,

$$\begin{array}{lclclcl} f^{lin}(x) & \in & \mathbf{R}^{\delta}, & \quad & f_a^{lin}(x) & = & l_a^T \, x \ , \\ f^{quad}(x) & \in & \mathbf{R}^{D}, & & f_k^{quad}(x) & = & x^T \, S_k \, x \ . \end{array} \quad (6)$$

Note that $f^{lin} = Lx$, where $L = [l_1 ... l_\delta]^T$. The exponential model version of (3) is:

$$\begin{aligned} p(x|g) &= (2\pi)^{-d/2}\, e^{\frac{1}{2}q(x|g)} \ , && (7) \\ q(x|g) &= C(\lambda_g, \phi_g) - \lambda_g^T\, f^{quad} + 2\phi_g^T\, f^{lin} && (8) \\ C(\lambda, \phi; \{S_k\}, L) &= \log\ \det(P) - \phi^T (LP^{-1}L^T)\phi \ . && (9) \end{aligned}$$

3. PARAMETER ESTIMATION

In this section we consider training all of the model parameters, $\Theta = \{\lambda_g, \phi_g, \{S_k\}, L\}$, so as to maximize, or at least approximately maximize, the total log likelihood of a set of labeled training vectors (x_t, s_t)[1]. This can be accomplished, according to the EM algorithm by iteratively updating the parameters. Given a current set of parameter $\hat{\Theta}$, the E-step of the EM algorithm gives the function $Q(\Theta, \hat{\Theta})$ that we need to maximize over Θ. Letting $N(s)$ be the number of samples associated with state s and $s(g)$ be the HMM state for Gaussian g and solving for the priors in the usual way, we have, as in [1, 2],

$$\begin{aligned} Q(\Theta, \hat{\Theta}) &= \sum_g n_g G(P_g; \Sigma_g), && (10) \\ n_g &= N(s(g))\pi_g, && (11) \\ \pi_g &= \frac{1}{N(s(g))} \sum_{t; s_t = s(g)} \gamma_t(g), && (12) \\ \gamma_t(g) &= \gamma_t(g, \hat{\Theta}) = \frac{\pi_g p(x_t|g, \hat{\Theta})}{p(x_t|s(g), \hat{\Theta})}, && (13) \\ G(P; \Sigma) &= \log(\det(P)) - \mathrm{Tr}(\Sigma P), \text{ and} && (14) \\ \Sigma_g &= \tilde{\Sigma}_g + (\mu_g - \tilde{\mu}_g)(\mu_g - \tilde{\mu}_g)^T. && (15) \end{aligned}$$

Here $\tilde{\mu}_g$ and $\tilde{\Sigma}_g$ are the mean and covariance matrix of the set of samples labeled by $s(g)$, with sample x_t given a weight of $\gamma_t(g)/n_g$. Letting $E_g(F(x))$ denote the expectation value of the function $F(x)$ for this sample distribution, we have

$$\begin{aligned} \tilde{\mu}_g &= E_g(x) && (16) \\ \tilde{\Sigma}_g &= E_g(xx^T) - \tilde{\mu}_g \tilde{\mu}_g^T \ . && (17) \end{aligned}$$

For the case of unconstrained means, fast algorithms for calculating the λ's and approximately calculating the $\{S_k\}$ are given in [1], and exact calculation of the $\{S_k\}$ is performed in [2]. For the general case considered here, section 3.1 describes how to jointly optimize for the tied matrix L defining the constrained linear space and the untied parameters $\{\phi_g\}$ defining the point θ_g in the linear space; and section 3.2 gives an algorithm to optimize for the untied parameters $\{\lambda_g, \phi_g\}$.

3.1. Optimization of the Linear Parameters

The part of the Q function which depends on L and $\{\phi_g\}$ for fixed precision matrices $\{P_g\}$ (i.e. fixed $\{S_k\}$ and $\{\lambda_g\}$) is:

$$\begin{aligned} Q(L, \{\phi_g\}) &= -\sum n_g ||\tilde{\theta}_g - \sum \phi_g^a l_a||^2_{P_g^{-1}} && (18) \\ \tilde{\theta}_g &= P_g \tilde{\mu}_g \ , && (19) \end{aligned}$$

where, for Σ a symmetric matrix,

$$||u||^2_\Sigma = u^T\, \Sigma\, u, \qquad \text{for } u \in \mathbf{R}^d. \qquad (20)$$

To maximize Q we start by letting $L^{(0)}$ be the solution of the total least squares problem obtained when the P_g^{-1} are all replaced by their average $\bar{\Sigma}$:

$$l_a^{(0)} = \bar{\Sigma}^{-1/2}\, \mathrm{eig}_a \left(\sum_g n_g \bar{\Sigma}^{1/2} \tilde{\theta}_g \tilde{\theta}_g^T \bar{\Sigma}^{1/2} \right) \ , \qquad (21)$$

where $\mathrm{eig}_a(Y)$ stands for the eigenvector corresponding to the a'th largest eigenvalue of the symmetric matrix Y.

Then we simply alternate between optimizing the quadratic in $\{\phi_g\}$ obtained from Q by fixing L and the quadratic in L obtained from Q by fixing $\{\phi_g\}$. The linear equations for the updates of ϕ_g and L are, respectively,

$$\begin{aligned} (LP_g^{-1}L^T)\phi_g &= L\tilde{\theta}_g && (22) \\ \sum_g n_g \phi_g \phi_g^T L P_g^{-1} &= \sum_g n_g \phi_g \tilde{\mu}_g \ . && (23) \end{aligned}$$

3.2. Optimization of the Untied Parameters

The function $G(P_g; \Sigma_g)$ above depends implicitly on the untied parameters λ_g and ϕ_g, the tied parameters $\{S_k\}$ and L, and the E-step means $\bar{f}_g^{lin} = E_g(f^{lin}(x))$ and $\bar{f}_g^{quad} = E_g(f^{quad}(x))$ of the linear and quadratic features. In this section, we consider optimizing G with respect to the untied parameters. Dropping the subscript g, we may write

$$\begin{aligned} G(\lambda, \phi) &= G(P; \Sigma) = E(q(x)) \\ &= C(\lambda, \phi; \{S_k\}, L) - \lambda^T\ \bar{f}^{quad} + 2\phi^T\ \bar{f}^{lin} && (24) \end{aligned}$$

$G(\lambda, \phi)$ is a concave function of ϕ and λ. It may be optimized by alternately maximizing with respect to ϕ and λ until convergence.

Maximization with respect to ϕ for fixed λ gives:

$$\phi = (LP^{-1}L^T)^{-1} \bar{f}^{lin} \ . \qquad (25)$$

Note that in the case of unconstrained means (where we take L to be the identity), $\bar{f}^{lin} = \tilde{\mu}$ and $\phi = P^{-1}\tilde{\mu}$, so that the model mean $\mu = P\phi$ equals the data mean $\tilde{\mu}$.

For the case when the means are unconstrained, an efficient technique was given in [1] to maximize G with respect to λ for fixed μ. We use a similar technique here to optimize for λ when ϕ is fixed. Namely, we apply the conjugate gradient algorithm with fast line searches for the maximum of the function $G(\phi, \lambda)$ along the line through the value of λ found at the end of the last conjugate gradient iteration in the direction of the conjugate gradient search direction η. The function to optimize when doing the line search is:

$$\begin{aligned} f(t) &= G(\lambda + t\eta, \phi) - G(\lambda, \phi) + (\text{a constant}) && (26) \\ &= \log\det(P_t\ P_0^{-1}) - \phi^T (L\ P_t^{-1}\ L^T)\phi - t\beta && (27) \\ \beta &= \eta^T\ \bar{f}^{quad} && (28) \\ P_t &= \sum_k (\lambda_k + t\eta_k) S_k = P_0 + tR && (29) \\ R &= \sum_k \eta_k S_k \ . && (30) \end{aligned}$$

[1] The labeling corresponds to a fixed alignment of some speech corpus. More generally, we could of course weight the training vectors using the forward-backward algorithm for HMM's.

Let $\{w_p\}$ be the eigenvalues and $\{y_p\}$ be an orthonormal basis of eigenvectors of $P_0^{-1/2} R P_0^{-1/2}$. The vector $v_p = P_0^{-1/2} y_p$ is called a generalized eigenvector of the pair (R, P_0) because $R v_p = w_p P_0 v_p$. Using

$$P_t{}^{-1} \quad = \quad \sum_{p=1}^{d} \frac{v_p \; v_p^T}{1 + t \; w_p} \ , \tag{31}$$

it is straight forward to verify that

$$\begin{aligned} f(t) &= -t\beta + \sum_{p=1}^{d} \log(1 + t w_p) - \frac{\alpha_p^2}{1 + t w_p} & (32) \\ \alpha_p &= v_p^T \, L \phi \ . & (33) \end{aligned}$$

Since L and ϕ are fixed, the α_p can be precomputed at the start of the line search. The line search is restricted to the values of t for which P_t is positive definite, i.e. we restrict to the interval of t for which $1 + t w_p$ is positive for all p.

4. SPAM-HDA MODELS

Following [7], we define a SPAM-HDA model to be one in which $\mathbf{R}^d$ is broken into two complementary subspaces:

$$x \mapsto \begin{bmatrix} x^1 \\ x^2 \end{bmatrix} = Tx, \quad T = \begin{bmatrix} T^1 \\ T^2 \end{bmatrix}, \quad T^i \text{ is } d_i \times d$$

and the Gaussians are tied along one of the subspaces:

$$T\mu_g = \begin{bmatrix} \mu_g^1 \\ \mu^2 \end{bmatrix}, \qquad P_g = T^T \begin{bmatrix} P_g^{11} & 0 \\ 0 & P^{22} \end{bmatrix} T$$

If P_g^{11} are unrestricted, the model is called a full covariance HDA model, or simply an HDA model. If they are diagonal, it is called a diagonal HDA model. If the P_g^{11} are allowed to be full covariance, but are required to be independent of g, the authors of [7] show that the maximum likelihood projection matrix agrees with the well known Linear Discriminant analysis (LDA) matrix. The more general SPAM-HDA model allows for an arbitrary subspace restriction on P_g^{11}. The feature precomputation cost for the models is $d_1 * d + D * d_1 * (d_1 + 1)/2$, which can be much smaller than the generic precomputation cost of $D * d * (d + 1)/2$.

5. CLUSTERING OF GAUSSIANS

By using Gaussian clustering techniques [8], it is possible to reduce the number of Gaussian one needs to evaluate significantly below the total number of Gaussians in an acoustic model. To apply this idea to a SPAM model, we will find a collection of cluster models $p(x|c)$, $c = 1, ..., nClusters$ (which are exponential models using the same tied features as the SPAM model) and an assignment $c(g)$ of Gaussians g of the SPAM model to clusters. We choose this clustering to maximize

$$\sum_g \pi_g E_g(q(x|c(g))) = \sum_c n_c G(\lambda_c, \phi_c; \bar{f}_c) \ \text{, where} \tag{34}$$

$$\begin{aligned} n_c &= \sum_{g;c(g)=c} \pi_c & (35) \\ \bar{f}_c &= \sum_{g;c(g)=c} \frac{\pi_g}{n_c} E_g((f^{quad}(x), f^{lin}(x))) \ , & (36) \end{aligned}$$

where $E_g(F(x))$ is now the expectation of $F(x)$ under the model $p(x|g)$, and the function G is given by (24).

Similarly to K-means clustering, equation (36) is optimized by alternately choosing the best clusters for each Gaussian and recomputing the cluster models (i.e. optimizing $G(\lambda_c, \phi_c; \bar{f}_c)$ using the technique of section 3.2).

To do acoustic modeling at time t, we first evaluate all the cluster distributions $p(x_t|c)$. We then use those results to make a judicious choice of $nActive$ Gaussians g from the original model for which to evaluate $p(x_t|g)$. Simple threshold values are used in place of the contribution of the unevaluated Gaussians.

6. EXPERIMENTAL RESULTS

We performed experiments using the same test data, training data with fixed Viterbi alignment (obtained using a baseline diagonal covariance model), and Viterbi decoder as was used in [1, 2, 3, 4]. The test set consists of 73743 words from utterances in small vocabulary grammar based tasks (addresses, digits, command and control) recorded in a car under idling, city driving, and highway driving conditions. The acoustic models had 89 phonemes and a total of 10253 Gaussians distributed across 680 context dependent states using BIC based on a diagonal covariance system.

The samples we used consisted of 117 dimensional vectors obtained by splicing nine consecutive thirteen dimensional cepstral vectors. As a first step, we created LDA projection matrices based on the within class and between class full covariance statistics of the samples for each state. For 8 different values of the dimension d_1 ranging from 13 to 117, we constructed matrices LDA(d_1) which project from 117 to d_1 dimensions and we built full covariance models, $FC(d_1)$, based on the projected vectors.

In order to verify that the projections used for the models FC(d_1) where good, we also used the Gaussian level statistics of the models FC(117) and FC(52) to construct LDA and HDA projection matrices (as well as a successful variant of HDA presented in [9]). The models FC(d_1) gave WERs within 3% relative of the best performing of all of the full covariance system (with the same projected dimension), with the sole exception that FC(13) had an error rate of 4.04%, whereas the system built on vectors output by the composition of LDA(52) and the 13×52 LDA matrix constructed based on the statistics of FC(52) had a WER of 3.90% (a 3.5% relative improvement).

Next, we built the systems we will refer to as MLLT(d_1), which are MLLT systems for vectors produced by multiplication with LDA(d_1). (As a check on these MLLT systems, we observed that they did as well or better than the MLLT system based on features built using the diagonal version of HDA.) We also built the systems SPAM($d = d_1, D = d_1, \delta = d_1$), which are SPAM systems with unconstrained means in dimension d_1 with precision matrices constrained to a $D = d_1$ dimensional subspace spanned by matrices $\{S_k\}$ obtained using the quadratic approximation (to the total likelihood function) technique of [1]. Figure 1 shows that the SPAM models achieve a significant fraction of the total improvement possible in going from MLLT to full covariance (while maintaining the same per Gaussian computational cost as the MLLT system).

Next, again using the techniques of [2], we built the models SPAM($d = 52, D = d_1, \delta = 52$), which are SPAM models for vectors produced by LDA(52) which have unconstrained means and have $D = 13,\ 20,\ 39$. Using these models to provide E-step statistics, we computed a $\delta = d_1$ by $d = 52$ matrix

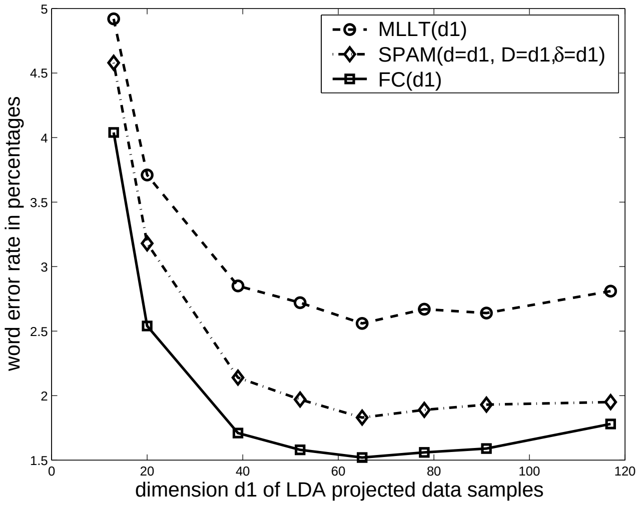

Fig. 1. WER as function of dimension showing SPAM model achieves significant fraction of improvement from MLLT to full covariance model. SPAM model has same per Gaussian compute time as MLLT.

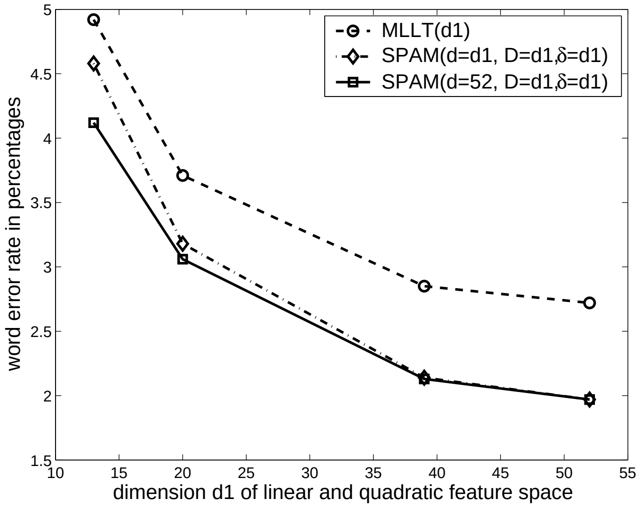

Fig. 2. WER as function of linear and quadratic feature space dimension showing that SPAM features from 52 dimensional model do better than SPAM features constrained to the LDA projected subspace.

L by the technique of section 3.1. Fixing this L and the $\{S_k\}$, $k = 1, ..., D = d_1$, we performed the EM algorithm, with the technique of section 3.2 for the M-step, to optimize $\{\pi_g, \lambda_g, \phi_g\}$. The models obtained are called SPAM$(d = 52, D = d_1, \delta = d_1)$. Figure 2 show that the system SPAM$(d = 52, D = d_1, \delta = d_1)$ ties or outperforms (significantly when $d_1 = 13$) the systems SPAM$(d = d_1, D = d_1, \delta = d_1)$ and (the even worse) MLLT(d_1). All of these system have equal per Gaussian computational cost.

We conclude with two experiments showing that the precomputation cost as well as the number of Gaussians that need to be evaluated can be reduced.

Both Figure 1 and 2 show that SPAM$(d = 39, D = 39, \delta = 39)$ has error rate 2.14%. A comparable error rate of 2.13% is obtained from a hybrid model trained by the techniques of [2] to give a SPAM model with $d = D = \delta = 39$, but with the S_k constrained to be linear combinations of $K = 156$ rank one matrices. This comparable error rate was obtained by balancing the small degradation due to the constraint on the S_k with the small improvements due to the fact that an affine S_0 was included and the S_k were trained in a true maximum likelihood fashion. The model reduces the feature precomputation cost from $Dd(d+1)/2 \approx 30000$ to $Kd+Dk \approx 12000$. Applying clustering of Gaussians to this model with $nClusers = 1024$ and decoding with $nActive = 1000$, as described in section 5, we found that the error rate increased only slightly, to 2.19%. This is at a significant savings from evaluating 10253 Gaussians to evaluating only 2024 Gaussians.

7. CONCLUSION

A SPAM model is just a state dependent mixture of exponential models with linear and quadratic features shared across all Gaussians. We have described how to train such models and have shown that *both* the flexibility to constrain the quadratic features and the flexibility to constrain the linear features can lead to improved accuracy at fixed computational cost per Gaussian. Furthermore, we have seen that the total computational cost can be lowered singificantly by choosing features that can be precomputed quickly and by clustering the Gaussians (as exponential models with common feature space) so that only a fraction of the Gaussians need to be evaluated.

8. REFERENCES

[1] S. Axelrod, R. Gopinath, and P. Olsen, “Modeling with a subspace constraint on inverse convariance matrices,” in *Proc. ICSLP*, 2002.

[2] K. Visweswariah, P. Olsen, R. Gopinath, and S. Axelrod, “Maximum likelihood training of subspaces for inverse covariance modeling,” Submitted ICASSP 2003.

[3] P. Olsen and R. Gopinath, “Modeling inverse covariance matrices by basis expansion,” in *Proc. ICASSP*, 2002.

[4] P. Olsen and R. Gopinath, “Modeling inverse covariance matrices by basis expansion,” *IEEE Transactions in Speech and Audio Processing*, submitted.

[5] R. Gopinath, “Maximum likelihood modeling with gaussian distributions for classification,” in *Proc. ICASSP*, 1998.

[6] M. J. F. Gales, “Semi-tied covariance matrices for hidden Markov models,” *IEEE Transactions in Speech and Audio Processing*, 1999.

[7] N. K. Goel and A. G. Andreou, “Heteroscedastic discriminant analysis and reduced-rank HMMs for improved speech recognition,” *Speech Comm.*, vol. 26, pp. 283–297, 1998.

[8] E. Bocchieri, “Vector quantization for efficient computation of continuous density likelihoods,” in *Proc. ICASSP*, 1993.

[9] G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, “Maximum likelihood discriminant feature spaces,” in *Proc. ICASSP*, 2000.