ENHANCED CONTROL AND ESTIMATION OF PARAMETERS FOR A TELEPHONE BASED ISOLATED DIGIT RECOGNIZER

Josef G. Bauer

Siemens AG, Corporate Research & Development
81730 Munich, Germany
Josef.Bauer@mchp.siemens.de

ABSTRACT

The paper studies the use of discriminative techniques for a telephone based isolated digit recognizer with respect to a reduced system complexity. The combination of Linear Discriminant Analysis (LDA) and Minimum Error Classification (MEC) training provides improved system performance at reduced costs for the training process and for the application. Experiments are performed on an isolated digit database recorded over public lines including approximately 700 speakers. The use of a single linear transformation matrix based on LDA allows the use of density modeling, that doesn't consider variances explicitly, at a high recognition rate. Minimum Classification Error training is found to perform best in case of a small amount of system parameters. A reduction of error rate up to 80% was achieved by the combination of the two methods for such a system configuration.

1. INTRODUCTION

In spite of the small vocabulary size isolated digit recognition over telephone lines is still a challenging task mainly due to the heavily varying channel and speaker characteristics. As for most speech recognition tasks Continuous Density Hidden Markov Models are known to be among the classifiers that give best performance for this application. The problem of applying this technique for real world systems is the high computational load especially for mixture density modeling. We describe our ongoing efforts on exploiting the performance of CDHMMs with particular respect to the memory and computational expense.

Reduction of computational costs is primarily carried out by simplifications applied to the speech modeling techniques like the use of a smaller number of densities. As discriminative methods are known to be most efficient in case of a mismatch between model assumptions and the described statistic process the use of such methods is appropriate for the considered systems.

Our most advanced system combines discriminative techniques namely Linear Discriminant Analysis (LDA) and Minimum Error Classification (MEC) training. In this paper we focus on giving detailed insight into the MEC training using the General Probabilistic Descent (GPD) method with respect to the use of LDA giving practical extensions to this technique.

2. FEATURE EXTRACTION AND LDA

The digitized speech at 8 kHz sampling rate is filtered using a preemphasis filter and a 256 point FFT is applied every 10 ms to a 25 ms Hamming windowed signal portion. 24 cepstral coefficients are extracted by convolution of the logarithmic power spectrum with mel scaled *sinc* kernels. To achieve a loudness normalization the mean of these 24 mel cepstral coefficients is subtracted from every coefficient. A Maximum Likelihood based on the fly cepstral mean removal (MLCC [1]) is applied to this 24 dimensional vector to compensate for different channel transfer functions. Further first and second order derivatives of the smoothened (by the factor 2) 24 coefficients are added to the feature vector. Using the high pass filtered logarithm of the total frame energy and it's first and second derivative we end up with a 51 dimensional feature vector. Note that all operations are carried out *on the fly* avoiding any normalization causing processing delays which are prohibitive for real time processing.

Linear Discriminant Analysis (LDA) [2, 4] is performed on a super-vector $\vec{y}$ consisting of two 51 dimensional subsequent mel cepstral feature vectors that form an enlarged vector with $2 \cdot 51$ components. So $\vec{y}$ includes more contextual information in time which can help to compensate the independence assumptions made in first order HMMs [3]. The LDA is used to find a single transformation matrix A that maximizes the objective function $\frac{sp(S_B)}{sp(S_W)}$ on the transformed vector $\vec{x} = A\vec{y}$. This objective function is based on discrimination of classes which are in our case chosen to be the states of the HMMs.

In a first step a decorrelation and a whitening transformation is performed on the feature space so that the mean covariance matrix S_w becomes the identity matrix. The second step of the LDA aims to maximize the class separability expressed by $tr(S_B)$, the trace of the between class covariance matrix.

The components of the resulting feature vector $\vec{x}$ are ordered with respect to the class discrimination measure. This allows us to reduce the dimension of $\vec{x}$. For all experiments with LDA transformed features we used the first 24 components of $\vec{x}$. Due to the reduced size of features vectors the computational effort for the computation of emission probabilities is reduced dramatically without loss of recognition performance.

3. MODEL TOPOLOGY

Each of the 11 German digit words is modeled by a strict left to right HMM λ_i consisting of S_i states. S_i is chosen proportionally to the mean duration of the specific word as seen in the training corpus. Transition probabilities between states are not explicitly trained. Instead, the penalty (logarithmic transformed probability) for a transition to the immediate succeeding state is 0 while penalties for a self loop and one state skip are both set to a fixed value T. All other state transitions are not allowed. Silence and background noise is modeled by a one state HMM that is only allowed before or after a digit model. The transition penalty for the self loop of this state is explicitly set to 0.

4. DENSITY MODELING

Our density modeling is strictly based on the CDHMM approach, so each density corresponds to one particular mixture. Mixtures of Gaussian densities model the probability density functions (pdfs) of the HMM states. All covariance matrices are explicitly set to the identity matrix. As we are using decorrelated and whitening transformed features this is closely related to the use of one covariance matrix, that is tied over all states of the HMMs. As an advantage of this procedure, we do not consider any variance modeling at the stage of density modeling.

The probability of one mixture is approximated by the highest probability of all densities within this mixture. By this means no summation of probabilities has to be carried out.

5. PARAMETER ESTIMATION

5.1. ML Parameter Estimation

Mixture densities are first initialized using a clustering algorithm to a high number of densities per mixture. In order to reduce the overall number of densities those densities with a low absolute occurrence measure are merged with their nearest neighbor. This corresponds to a maximum a-posteriori approach for controlling the the number of densities per mixture. The mean vectors of the Gaussian densities are reestimated using an iterative Maximum Likelihood based Viterbi training.

5.2. MEC Parameter Estimation

Using Minimum Error Classification [5] training the recognizer parameters can be adjusted to achieve a local minimum of the word error rate on the training set.

We are using a simplified misclassification measure $d_i(X, \Lambda)$ for a model λ_i given a parameter set Λ and the feature set X of an utterance

$$d_i(X, \Lambda) = -g_i(X, \lambda_i) + \max_{j \neq i} g_j(X, \lambda_j)$$

with the log likelihood score of an utterance given a model $\lambda_i : g_i(X, \Lambda) = \log p(X|\lambda_i)$ [6]. So d_i is the difference between the scores for a model λ_i and the best competitive model.

The misclassification measure is embedded in a sigmoid function

$$l_i(d_i(X, \Lambda)) = \frac{1}{1 + e^{-\gamma d_i(X,\Lambda)}}$$

and the objective function is formed as the loss function for the whole training set $\{X^1, X^2, \cdots X^R\}$

$$l(\Lambda) = \frac{1}{R} \sum_{r=1}^{R} l_{C(X^r)}(X^r, \Lambda)$$

where $C(X^r)$ is the correct class corresponding to X^r.

$l(\Lambda)$ is a differentiable approximation of the error rate on the training set. Using the General Probabilistic Descend method model parameters Λ can be adjusted in order to minimize this objective function. Parameters are updated iteratively (iteration index n):

$$\Lambda_{n+1} = \Lambda_n - \epsilon_n U_n \nabla l(\{X^r\}, \Lambda_n)$$

ϵ_n is a variable scalar and U_n a matrix both used for scaling the gradient ∇l. For U_n, the use of diagonal matrix containing the variances of feature components is reported to be useful [5]. Due to the fact that the mean class within variances of LDA transformed features are all equal we use the identity matrix for U_n.

One of the problems for practical application of the MEC/GPD method is the adjustment of the estimation process. In our case we have to adjust γ, which controls the form of the sigmoid function, and ϵ_n controlling the scaling of the gradient. In order to find a suitable method to adjust γ we take a closer look at the parameter update formula that can be derived from the above equations.

Viterbi decoding of an utterance with corresponding feature sequence X provides the log likelihood scores $g_i(X, \lambda_i)$ as well as the time alignment of the feature vectors to the states of the models. Given this information we have to apply the GPD update to minimize $l_{C(X)}$. Therefore we have to consider the model for the correct (spoken) word λ_J ($J = C(X)$) as well as the best competitive model λ_K with $K = \text{argmax}_{i \neq J}\, g_i(X, \lambda_i)$.

For the case of one single Gaussian density per state with the covariance matrix being the identity matrix a simple and illustrative parameter update formula can be derived:

$$\hat{\vec{\mu}}_s = \vec{\mu}_s + Q \cdot \frac{\partial l(d)}{\partial d} \cdot (\vec{x}_t - \vec{\mu}_s)$$

This formula describes how the mean vector of a state s is influenced by a feature vector $\vec{x}_t$ that was aligned to this state in time frame t. Q is a positive constant for the correct class and has the same absolute value with a negative sign for the class C_K. This parameter update can be interpreted as drawing the densities for the correct class towards the feature vector while performing the opposite for the incorrect class (see figure 1).

The factor $\frac{\partial l(d)}{\partial d} = \frac{\partial l_J(d_J(X,\Lambda))}{\partial d_J(X,\Lambda)}$ controls the scaling of the difference vector for a specific X though it only depends on $d_{C(X)}(X, \Lambda)$. Although the above formula only holds for single densities it can also be applied to mixture densities using the best density approximation (see section 4.). Given this approximation this is conform with the MEC/GPD approach because only the best density of one state for a particular time frame has influence on the objective function.

Figure 2 shows $\frac{\partial l(d)}{\partial d}$ and a typical histogram for the frequency of values d_J for a first MEC training iteration. A

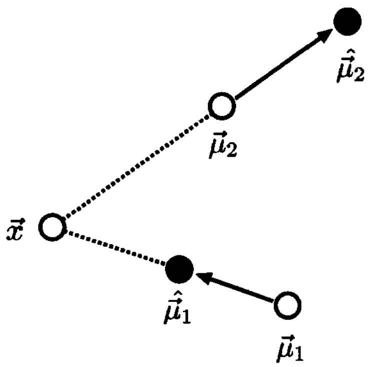

Figure 1. Illustration of the parameter update caused by a feature vector $\vec{x}$ in a two dimensional feature space. $\vec{x}$ was aligned to state 1 belonging to the correct model and state 2 of an incorrect model. $\vec{\mu_1}$ is drawn towards $\vec{x}$ while $\vec{\mu_2}$ is drawn away from the feature vector.

positive value for d_J stands for a misclassification while a negative value stands for a correct classification. So the diagram illustrates how training samples X influence the parameter update according the misclassification measure d_J. For values of d_J around 0 the influence expressed by $\frac{\partial l(d_J)}{\partial d_J}$ reaches a maximum, while for high values of $|d|$ the influence decreases.

Following the above ideas we propose a heuristic strategy on the adjustment of the parameter γ: γ should be set to such a value that $\frac{\partial l(d)}{\partial d}$ has fallen to almost 0 for the maximum positive values d that occur during training. Such high values of d can be interpreted as crude misclassification eventually caused by corrupted utterances which should not be considered for parameter estimation. Note that hereby γ is not set to a value that gives best correspondence between the objective function l and the real error rate on the training patterns. With the applied simplified misclassification measure a value $\gamma \rightarrow \infty$ would provide a perfect correspondence but the objective function would no longer be differentiable. The idea behind the proposed method is a compromise between avoiding over-adaptation and considering as much training material as possible.

It is possible to derive another useful practical extension from the above considerations: Utterances with extremely low influence on the parameter update during the first iteration are very unlikely to influence training during any further iteration. Therefore it is not necessary to consider utterances with $\frac{\partial l(d)}{\partial d}$ lower than a certain threshold for all subsequent iterations. It is hereby possible to reduce the effort for MEC/GPD training without loss of accuracy.

We are using a fixed $\epsilon_n = \epsilon_0$ as we did not find improved convergence through a variable ϵ_n during the iteration process. With an appropriate value for ϵ_0 we always observed fast convergence in less than 10 iterations. For the purpose of choosing an appropriate value for ϵ_0 the standard deviation σ_∇ of all accumulated parameter updates during the first iteration is considered. We adjust ϵ_0 in such a way that σ_∇ equals a fixed portion of the mean class within deviation of the features which equals 1 for LDA transformed features.

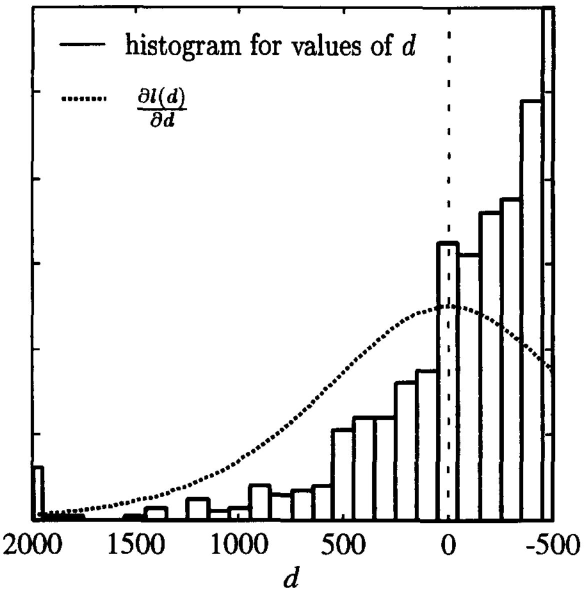

Figure 2. (smoothened) histogram for values of d occurring during first MEC iteration and $\frac{\partial l(d)}{\partial d}$

6. EXPERIMENTS

For all experiments the part of the German Voice Mail database containing only isolated spoken German digits (11 words) was used for training and testing. For this database utterances of 700 speakers from various regions were recorded over German public telephone lines (including cellular phones). Approximately 600 speakers where selected for training and the remaining 100 for the test set.

The baseline system uses the mel cepstral feature vector with 51 components without the LDA based feature transformation. In this case the parameters of the densities were trained using 10 iterations of ML Viterbi training. The use of 400 Gaussian densities (with unified variance modeling) lead to 7.4% word errors on the test set. The high error rate for this systems results basically from the simple density modeling. The application of more complex modeling techniques would reduce the gain achieved with LDA quite a lot.

In all following experiments we used the 24 dimensional LDA based feature vector.

In order to highlight the importance of the amount of free parameters for ML and especially MEC training we performed 3 experiments using a total number 400, 1000 and 2000 Gaussian densities. For each of the 3 recognizer configurations 10 iterations of Maximum Likelihood training were applied. Taking the ML trained models for initial parameter sets 10 iterations of MEC/GPD training where performed.

Figure 3 compares the error rates for Maximum Likelihood and MEC training for the training set as well as for

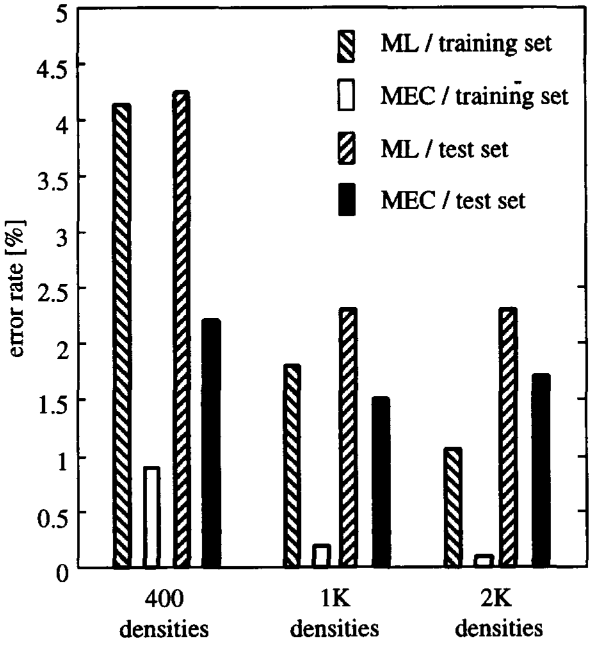

Figure 3. comparison of MLE and MEC training for different numbers of free parameters

the test set.

In the case of ML trained models an increasing number of densities reduces the error rate on the training set from 4.1% (400 dens.) to 1.8% (1K dens.) and 1.1% (2K dens.). Looking at the performance of the ML trained recognizer on the test set one finds that the use of 1000 densities instead of 400 reduces the error rate from 4.2% to 2.3%, while using an other 1000 densities more did not reduce the the errors on the test set any further.

For the considered configurations all error rates achieved with MEC/GPD training are below those achieved with ML training. Comparing the results for the three investigated number of free parameters we find similar trends as for ML training. The error rates on the training set are 0.9%, 0.2% and 0.1%. On the test set the error rates resulting from MEC trained models were 2.2% 1.5% and 1.7% for 400, 1000 and 2000 densities. It can be seen that the reduction of error rate (on the test set) through discriminative training decreases with a higher number of parameters from 48% to 33% and 26%. With the use of 2000 densities the error rate was even higher in the case of MEC training compared to the use of 1000 densities. One can assume that over-adaptation appeared here.

The amount of free parameters of the best performing system (LDA based features, 1000 densities, MEC training) is almost the same as for the baseline system which uses the mel cepstral feature vector without LDA based transformation and MEC training. The reduction of word errors at comparable computational costs is 80%.

7. CONCLUSION

We successfully applied two discriminative training techniques to a CDHMM based isolated digit recognizer achieving high recognition performance using limited memory and computational resources.

A linear feature transformation based on a LDA maximizing a class discrimination measure with the classes being the states of HMMs was applied to a high dimensional speech representation containing a high amount of contextual (dynamic) information. The resulting feature vector offers a compact representation of the speech signal with the practical advantage of decorrelated and variance whitened components. The latter attributes allow the use of modeling techniques with reduced complexity and result in simplified parameter estimation procedures.

The MEC/GPD method is applied to reestimate the parameters of a HMM classifier using the LDA transformed features. The approach leads to quite simple and illustrative parameter update expressions in case of the applied model assumptions. Data driven considerations are found to provide methods for easy adjustment of the estimation process.

In experiments that use different numbers of parameters discriminative training methods perform best in the case of a high ratio between training samples and free parameters. Over-adaptation of the recognizer on the training samples is a severe problem that has to be considered here. A recognizer configuration that uses only 400 Gaussian densities was found to perform better when trained with the MEC/GPD method than a ML trained system with 1000 densities. Using the same amount of free parameters MEC/GPD training lead to reduction of the error rate up to 48%.

REFERENCES

[1] Hauenstein A., Marschall E, *Methods for Improved Speech Recognition over Telephone Lines*, Proceedings ICASSP 95

[2] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973

[3] C. Wellekens, *Explicit Time Correlation in Hidden Markov Models for Speech Recognition*, Proceedings ICASSP 1987

[4] R. Haeb-Umbach, D. Geller, H. Ney, *Improvements in Connected Digit Recognition Using Linear Discriminant Analysis And Mixture Densities*, Proceedings ICASSP 1993

[5] Chou W., Juang B.H., Lee C.H., *Segmental GPD Training of HMM Based Speech Recognizer*, Proceedings ICASSP 92

[6] Euler S., Zinke J, *Experiments on the use of the Generalized Probabilistic Descent Method in Speech Recognition*, Proceedings ICSLP 1992