

# Learning intra-speaker model parameter correlations from many short speaker segments

Anne K. Kienappel

Philips Research Laboratories  
Weißhausstraße 2, D-52066 Aachen, Germany

Anne.Katrin.Kienappel@philips.com

## Abstract

Very rapid speaker adaptation algorithms, such as eigenvoices or speaker clustering, typically rely on learning intra-speaker correlations of model parameters from the training data. On the base of this a-priori knowledge, many model parameters can be successfully adapted on the basis of few observations. However, eigenvoice training or speaker clustering is non-trivial with training databases containing many short speaker segments, where for each speaker the available data to detect intra-speaker correlations is sparse. We have trained eigenvoices that yield a small but significant word error rate reduction in on-line adaptation (i.e. self adaptation) for a telephony database with on average only 5 seconds of speech per speaker in training and test data.

## 1. Introduction

In many automatic speech recognition applications each user utters typically only a few seconds of speech. Speaker adaptation has to be very fast to be of any use for such systems. Various very fast adaptation methods have been shown to yield significant improvements with only a few seconds of adaptation speech, such as eigenvoices [1], or speaker clustering [2].

In many reports about very rapid speaker adaptation, training databases contain no more than a few hundred speakers and a reasonable amount of training data for each speaker. For applications with short user interactions, however, the training data, which is likely to stem from the target application, often consist of short speaker segments. This structure leads to problems with standard fast adaptation methods.

The practical problems in speaker clustering and standard eigenvoice training become easily apparent. Standard eigenvoice training is based on a principle component analysis (PCA) on speaker specific models. Such models can hardly be trained given only a few seconds of speech, and performing a speaker PCA on thousands of models of a few million parameters each is a numerical nightmare. Speaker clustering with a database with short speaker segments is equally non-trivial. Lacking any background information about speakers such as gender, dialect etc., a relevant numerical speaker representation and distance measure for clustering must be defined. This is not obvious given only short speech samples which may not even have many phonemes in common (see [2])<sup>1</sup>.

We believe that these problems can, at least partly, be interpreted as manifestations of a generic problem with rapid adaptation methods when training data for each individual speaker

<sup>1</sup>Note the danger of clustering e.g. speakers whose utterances contain a similar set of phonemes, rather than speakers with similar characteristics.

is sparse. For model adaptation based on few observations, a-priori knowledge about intra-speaker correlations of model parameters is needed. However, knowledge about intra-speaker parameter correlations must be learned from observations of different model parameters for the same speaker, which are limited when training data for each individual speaker is sparse.

Speaker clusters and eigenvoices are closely related. Given either for a database with short speaker segments, the other can be easily generated. Speaker clusters define a convenient training corpus structure for an eigenvoice PCA based on cluster models rather than speaker dependent (SD) models. A good set of eigenvoices, on the other hand, can be used to compute eigen-coefficients for each speaker, which in turn make good compact speaker representations for speaker clustering. This relationship can be used to iteratively improve both eigenvoices and speaker clusters. If there is no hard speaker-to-cluster assignment, but each speaker is modelled by a linear combination of cluster models (or eigenvoices), a closed maximum likelihood (ML) approach which implements this iterative principle is given by cluster adaptive training (CAT) [3] or the closely related maximum likelihood eigen-decomposition (MLED) [4].

We investigate different strategies of using eigenvoice PCA to initialise CAT/MLED for a spontaneous telephony speech database which contains  $\sim 16k$  utterances by as many different speakers and a length of on average  $\sim 7$  seconds,  $3/4$  of which are speech.

## 2. Theoretical background

In this section we first review the relevant equations for adaptation by density mean translation according to the anisotropic MAP framework [5], as well as ML eigenvoice training [3][4].

### 2.1. Notation

We use Viterbi alignments and Gaussian mixture models with the maximum approximation. Our Gaussians have diagonal covariance matrices. For simplicity, all density means and observations are noted not in the original feature space, but in a space where all components are scaled by the variance of the relevant probability density. We denote

$N_{F/E/D/S}$	$\equiv$	number of features / eigenvoices / densities / speakers,
$N_{d,s}$	$\equiv$	observation count for density $d$ and speaker $s$ ,
$\vec{\mu}_d$	$\equiv$	density mean of density $d$ ,
$\delta_{i,j}$	$\equiv$	Kronecker delta,
$\delta(d,s)$	$\equiv$	$1 - \delta_{N_{d,s},0}$ ,
$\vec{t}_d$	$\equiv$	adaptation translation for density $d$ ,

- $\vec{\Delta}_{d,s}$   $\equiv$  difference between average observation of density  $d$  for speaker  $s$  and  $\vec{\mu}_d$ ,  
 0 if  $\delta(d, s) = 0$ ,  
 $E_i$   $\equiv$  the  $i^{\text{th}}$  eigenvalue,  
 $\vec{e}_{i,d}$   $\equiv$  the  $N_F$  dimensional part of the  $i^{\text{th}}$  eigen-vector,  $\vec{e}_i$  which belongs to the density  $d$ ,  
 $\vec{c}_s$   $\equiv$  vector of eigen-coefficients for speaker  $s$ .

All of  $\vec{\mu}_d$ ,  $\vec{\Delta}_{d,s}$ ,  $\vec{t}_d$ , and  $\vec{e}_{i,d}$  may appear without the density index  $d$  to denote the super-vector of dimension  $N_D \cdot N_F$  which concatenates the vectors for all densities.

## 2.2. Adaptive ML density translation

Given orthonormal adaptation directions  $\vec{e}_i$  in equation 1 and variances between SD models  $E_i$  in direction of  $\vec{e}_i$ , the anisotropic MAP framework [5] defines the ML a posteriori adaptive translation of density means.

In this paper we restrict the translation to the sub-space spanned by a few ( $N_E$ ) preferred adaptation directions ( $\vec{e}_i$ ). In this case the adaptive translation super-vector  $\vec{t}$  is defined as<sup>2</sup>

$$\vec{t} = \sum_{i=1}^{N_E} c_i \vec{e}_i \quad (1)$$

where the vector of eigen-coefficients  $\vec{c}$  is, given a set of adaptation data, defined by  $N_E$  linear equations

$$\mathbf{A}\vec{c} = \vec{t} \quad (2)$$

which can be solved by inversion of the square, symmetric matrix  $\mathbf{A}$ . The elements of  $\mathbf{A}$  and the inhomogeneity vector  $\vec{t}$  are given by

$$A_{i,j} = \sum_{d=1}^{N_D} N_d (\vec{e}_{i,d} \cdot \vec{e}_{j,d}) + \frac{\delta_{i,j}}{S \cdot E_i} \quad (3)$$

$$t_j = \sum_{d=1}^{N_D} N_d (\vec{e}_{j,d} \cdot \vec{\Delta}_d)$$

The factor  $S$  is an empirical scaling factor for all eigenvalues, i.e.  $S = 1$  if the variances between SD models  $E_i$  are correct. Note that in this general adaptation framework, it would be more correct to call the  $E_i$  ‘‘SD model variances’’ rather than ‘‘eigenvalues’’ and the  $\vec{e}_i$  ‘‘adaptation directions’’ rather than ‘‘eigenvoices’’. However, we decided to stick to the latter, more simple expressions, which originate from the special case where  $\vec{e}_i$  and  $E_i$  are defined by a PCA on SD models.

## 2.3. Training of translation directions

### 2.3.1. Speaker model PCA

The original idea of adaptation using eigenvoices is that adaptation is most desirable in the directions of maximal SD model variation. Thus the  $\vec{e}_i$  and  $E_i$  are determined by a PCA on SD model super-vectors, i.e. an eigenvalue analysis of the SD model correlation matrix  $\mathbf{C}$ . We define the speaker specific density means  $\vec{\mu}_{d,s}$  and the average speaker mean  $\vec{\mu}_d$  as

$$\vec{\mu}_{d,s} = \delta(d, s)(\vec{\mu}_d + \vec{\Delta}_{d,s}) + (1 - \delta(d, s))\vec{\mu}_d, \quad (4)$$

$$\vec{\mu}_d = \frac{1}{N_S} \sum_{s=1}^{N_S} \vec{\mu}_{d,s} = \frac{\sum_{s=1}^{N_S} \delta(d, s) \vec{\mu}_{d,s}}{\sum_{s=1}^{N_S} \delta(d, s)}. \quad (5)$$

<sup>2</sup>We omit the speaker index  $s$  in these equations, because only a single test speaker is relevant.

With respect to the super-vectors of these speaker means, the entries of  $\mathbf{C}$  are defined by

$$C_{i,j} = \frac{1}{N_S} \sum_{s=1}^{N_S} (\mu_{i,s} - \bar{\mu}_i) \cdot (\mu_{j,s} - \bar{\mu}_j). \quad (6)$$

### 2.3.2. ML reestimation

Instead of defining translation directions by PCA, they can be defined using an ML criterion on the multi-speaker training data. If every training speaker has their Gaussian acoustic model,

$$\vec{\mu}_s = \vec{\mu} + \sum_{i=1}^{N_E} c_{i,s} \vec{e}_i \quad (7)$$

the term in the total Viterbi log-likelihood that depends on the eigenvoices  $\vec{e}_i$  and speaker eigen-coefficients  $\vec{c}_s$  is

$$L = \sum_{s=0}^{N_S} \sum_{d=0}^{N_D} \sum_{f=0}^{N_F} N_{d,s} \left( \sum_{i=0}^{N_E} c_{i,s} e_{i,d,f} - \Delta_{d,s,f} \right)^2. \quad (8)$$

By solving the zeros of the partial derivatives of this equation with respect to the  $c_{i,s}$  on the one hand and to the  $\vec{e}_i$  on the other hand, a (local) maximum of the likelihood can be found by iterative estimation of  $c_{i,s}$  and  $\vec{e}_i$ . We keep the speaker independent (SI) models  $\vec{\mu}$  and the Viterbi alignment constant during this process. The process can be initialised with either  $c_{i,s}$  or  $\vec{e}_i$ . We discuss our initialisation procedure in the next section.

Optimal speaker coefficient  $c_{i,s}$  given a set of eigenvoices are given by  $N_e$  equations  $\mathbf{A}_s \vec{c}_s = \vec{t}_s$ . The definitions of  $\mathbf{A}_s$  and  $\vec{t}_s$  differ from those in equations 3 only by the addition of the speaker index and the fact that the second term in the sum defining  $A_{i,j}$ ,  $\delta_{i,j}/(S \cdot E_i)$ , is missing. The coefficients for training speakers are thus the same as those estimated in adaptation in the limit of an infinite number of adaptation utterances,  $E_i \rightarrow \infty$ , or  $S \rightarrow \infty$ .

The optimal eigenvoices given a set of speaker coefficients are defined component-wise as

$$\mathbf{B}_d \vec{e}_{d,f} = \vec{j}_{d,f}, \quad (9)$$

where  $\mathbf{B}_d$  of dimension  $N_E^2$  is defined by

$$B_{d,i,j} = \sum_{s=1}^{N_S} N_{d,s} c_{i,s} c_{j,s}, \quad (10)$$

$$j_{d,f,i} = \sum_{s=1}^{N_S} N_{d,s} c_{i,s} \Delta_{d,s,f}. \quad (11)$$

Note that in the re-estimation process the  $\vec{e}_i$  are not orthonormal. We orthonormalise them before using them in adaptation, because our anisotropic MAP approach assumes orthonormal eigenvoices. For adaptation, we also need the  $E_i$ , i.e. the model variances in speaker direction. Given the final, orthonormalised set of  $\vec{e}_i$  and the correspondingly transformed  $\vec{c}_s$  the  $E_i$  are given by

$$E_i = \frac{1}{N_S} \sum_{s=1}^{N_S} c_{i,s}^2. \quad (12)$$

We perform the ML eigenvoice reestimation on speech observations only and use observation discounting to avoid over-training artefacts caused by observation sparsity.

### 3. Eigenvoice training strategies

In this section we describe the eigenvoice training strategies we adopted for the special case of a database with many short speaker segments and their practicalities.

The framework for iterative reestimation of eigenvoices and speaker clusters outlined in 2.3.2 offers flexibility with respect to initialisation. However, to avoid getting stuck in a bad local optimum, the quality of initialisation is important. One obvious choice expected to lead to convergence in a good local optimum is the initialisation with PCA eigenvoices. However, short speaker segment structure of the database demands a specialised PCA set-up.

#### 3.0.3. Sparse PCA

Given 15k training speakers and the sparsity of data for each speaker, we can neither compile nor store full SD models. To take advantage of the observation sparsity, we store the  $\tilde{\mu}_d$  from equation 5 and all non-zero  $\tilde{\mu}_{i,s} - \tilde{\mu}_i$ , i.e. only those parts of the SD models that belong to densities which have really been observed for the speaker. For these we define the SD model simply as the average of the observed features.

Note that  $\mathbf{C}$  is of dimension  $(N_F \cdot N_D)^2$ , which is  $\sim (10^6)^2$  for our case of  $N_D = 3 \cdot 10^4$  and  $N_F = 31$ . Eigenvalue analysis for such a large matrix is not trivial. In principle, the eigenvalue problem can be reduced to a dimensionality of  $N_S^2$  by orthonormalisation of the  $\tilde{\mu}_s$ . However, an orthonormalisation of  $N_S \approx 15 \cdot 10^3$  vectors of dimension  $10^6$  in itself is hardly feasible.

We therefore use the LASO package [6] by D.Scott, which implements the Lanczos algorithm to find a few eigenvalues of large *sparse* symmetric matrices. The correlation matrix  $\mathbf{C}$  is not intrinsically very sparse. An entry  $C_{i,j}$  is only zero if there is no speaker for which both densities to which the indices  $i$  and  $j$  belong have been observed. We find that about a third of the  $\sim 400 \cdot 10^9$  distinct<sup>3</sup> entries are zero. This leaves still many more entries than can reasonably be stored, also many more than the LASO package can easily cope with, which we found to be approximately  $50 \cdot 10^6$  non-zero entries in this case. We therefore artificially enhance sparsity and prune the matrix as follows.

1. Set entries that were observed for less than a minimum speaker count to zero.
2. Compute remaining entries, and set entries that have an absolute value of less than a minimum value to zero.

The minimum speaker count (75 for the above example) is chosen such that the number of entries to be computed is small enough (e.g.  $\sim 10^9$ ) such that step 2 can be performed in a reasonable time. The minimum value is chosen to obtain number of non-zero matrix entries that can be easily handled by the LASO package.

#### 3.0.4. Coefficient initialisation

Rather than with eigenvectors, the ML eigenvoice reestimation can be initialised with speaker eigen-coefficients. These coefficients can be taken not only from hard assignments of speakers to clusters, but also from a different eigenvoice system.

The main advantage of this approach is that the preliminary system may be based on a much simpler set of models with less parameters. This reduces both the observation sparsity and the

<sup>3</sup>Two equal entries with a swapped pair of indices are only counted once.

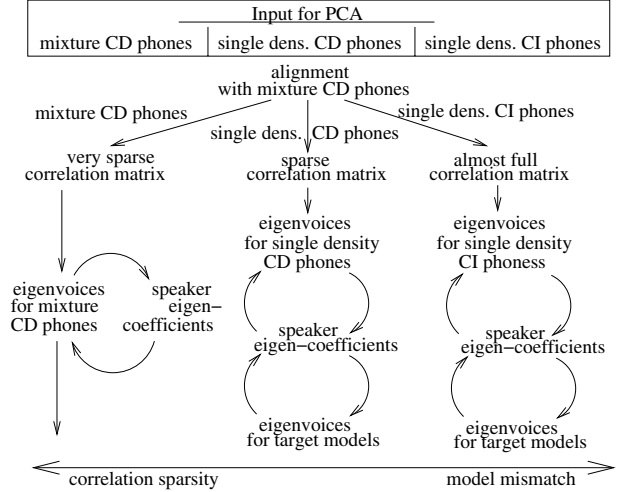


Figure 1: Eigenvoice training strategies

numerical problems with the PCA. In the case of single density context independent (CI) models this means  $\mathbf{C}$  is of dimension  $\approx (5 \cdot 10^3)^2$ , which allows an eigenvalue analysis without having to artificially enhance sparsity and the associated information loss. For single density context dependent (CD) phones, pruning of  $\mathbf{C}$  is still necessary, but to a much lesser extent.

A possible drawback is the large mismatch between PCA models and target models. It is not immediately obvious why the speaker characteristics should port well between very different acoustic models.

We compared direct PCA initialisation and coefficient initialisation from single density CD and CI phones. The columns in figure 1 summarise the eigenvoice training sequences for the three strategies respectively. They also represent the strategies' positions w. r. t. the trade-off between model mismatch and sparsity (of observations and artificially introduced into  $\mathbf{C}$ ).

## 4. Experiments

### 4.1. Database

Our automatic speech recognition task are the first 1k utterances of the OASIS database, which was collected by BText Technologies [7]. The data consists of recordings and transcriptions of the first customer utterance in operator assistance calls to BT. A further 7.5k utterances of the OASIS database as well as an additional UK English spontaneous speech database of similar size (also collected by BT and matching OASIS very well acoustically) are available for training and development of acoustic and language models. The entire acoustic database contains  $\sim 16k$  utterances by as many different speakers. One utterance lasts on average  $\sim 7$  seconds, 3/4 of which are speech. The acoustic training data contains  $\approx 25$  hours of speech.

### 4.2. Recogniser

Our acoustic models are continuous Gaussian mixture density hidden Markov models trained using a Viterbi algorithm. The feature vectors consist of 31 components computed from cepstral features using linear discriminant analysis (LDA) and a maximum likelihood linear transform (MLLT). They are computed with a 10 ms frame shift. We use triphone models and decision trees to determine state tying between them. The mod-

els contain 30k Gaussian densities, which are shared between different states, and 3k diagonal variances, which are shared between the Gaussians.

The recogniser vocabulary consists primarily of the 4k words that occur more than once in the training transcriptions. It also contains 1k additional words that have high unigram probability based on various additional text corpora (some from LDC and ELRA). The vocabulary includes 300 word phrases, which were constructed by successively merging words based on a frequency criterion. An optimised trigram language model was compiled for this vocabulary using primarily the training transcriptions, but also the additional text corpora.

We perform unsupervised on-line (or self) adaptation. A partial trace-back is performed every second. If there are new words, an alignment of *all* recognised words of this sentence is made with the *current* recogniser model and the *original*, *SI* model according to adapted using this alignment. The current recogniser model is updated with the result. At the end of each sentence (i.e. at a speaker change) the model in the recogniser is reset to the SI model.

### 4.3. Results

In figure 2(a) the word error rate (WER) is plotted depending on the number of eigenvoices and the training strategy. Performing the initial PCA on single density CD phones is the best approach, although only slightly better than that of single density CI phones. We found that eigenvoices defined by a sparse PCA using the target models, the least successful initialisation strategy, did not reduce the WER at all unless improved by ML reestimation. The optimal number of eigenvoices is about 10.

Figure 2(b) shows WER variation with respect to the eigenvoice scaling parameter  $S$ . The parameter is not critical as long as it is  $> 0.1$ . Indeed, the ML limit  $S \rightarrow \infty$  shows almost the same performance as the MAP value of  $S = 1$ .

The relative WER reduction of just over 2% for a few seconds of adaptation is still smaller than some other rapid adaptation results reported for similarly short adaptation times, but longer training speaker segments (see e.g. [5], [2]). There can be two different reasons for this very limited improvement. On the one hand, there may simply not be enough intra-speaker correlation information in the training data. On the other hand, our algorithm may not be taking proper advantage of the available information. One aspect of the latter is that we adapt density means only. We believe that appropriate adaptation of mixture weights and variances should yield a more consistent and thereby better SD model.

## 5. Conclusions

We have investigated eigenvoice training on a spontaneous telephony speech database with many short speaker segments. Eigenvoices which are initialised with a PCA on speaker dependent single density triphones and ported to the target models via ML eigen-coefficients yield a small but significant 0.7% absolute (2% relative) WER reduction by unsupervised on-line speaker adaptation on utterances containing on average 5 seconds of speech.

## 6. Acknowledgments

I would like to thank Dietrich Klakow providing the language model for the OASIS task, and Mark Farrell, David Attwater, James Allen and Peter Durston from BT for providing not only

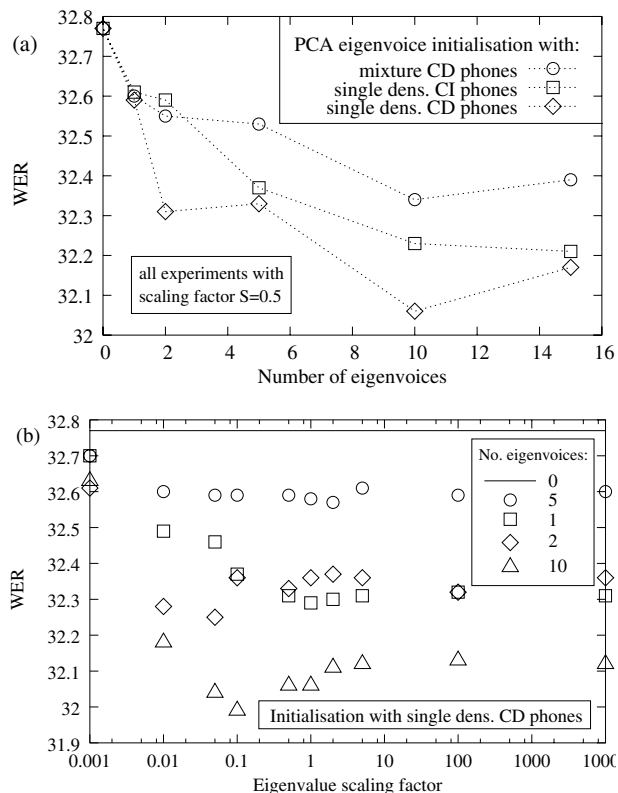


Figure 2: WER depending on the number of eigenvoices for different training strategies and depending on the eigenvoice scaling factor.

the database but also plenty of support with it.

## 7. References

- [1] Kuhn, R., Junqua, J.-C., Nguyen, P., and Zielski, N., "Rapid speaker adaptation in Eigenvoice space", IEEE Trans. Speech and Audio Proc., 8(6):695-707, 2000.
- [2] Pusateri, E.J. and Hazen, T.J., "Rapid adaptation using speaker clustering", Proc. ICSLP2002, Vol. 1, p. 61, Denver, Colorado, 2002
- [3] Gales, M., "Cluster Adaptive Training of hidden Markov Models", IEEE Trans. Speech and Audio Processing, 8(4):417-427, 2000.
- [4] Nguyen, P., Wellekens, C. and Junqua, J.-C., "Maximum Likelihood Eigenspace and MLLR for Speech Recognition in Noisy Environments" Proc. Eurospeech99, p.2519, Budapest, Hungary, 1999
- [5] Botterweck, H., "Anisotropic MAP defined by Eigenvoices for large vocabulary continuous speech recognition", Proc. ICASSP2001, p.353, Salt Lake City, Utah, 2001
- [6] Scott, D. "LASO package for computing a few eigenvalues of a large (sparse) symmetric matrix", Netlib mathematical software file server (<http://www2.ucsc.edu/cats/sc/software/netlib/>)
- [7] Durston, P.J. *et al.*, "OASIS Natural Language Call Steering Trial.", Proc. Eurospeech2001, p.1323, Aalborg, Denmark, 2001.