Rapid Speaker Adaptation in Latent Speaker Space with Non-negative Matrix Factorization

Xueru Zhang^{1,*}, Kris Demuynck, Hugo Van hamme

*Katholieke Universiteit Leuven, Department of Electrical Engineering - ESAT
Kasteelpark Arenberg 10, Bus 2441, B-3001 Leuven, Belgium*

Abstract

A novel speaker adaptation algorithm based on Gaussian mixture weight adaptation is described. A small number of latent speaker vectors are estimated with non-negative matrix factorization (NMF). These latent vectors encode the distinctive systematic patterns of Gaussian usage observed when modeling the individual speakers that make up the training data. Expressing the speaker dependent Gaussian mixture weights as a linear combination of a small number of latent vectors reduces the number of parameters that must be estimated from the enrollment data. The resulting fast adaptation algorithm, using 3 s of enrollment data only, achieves similar performance as fMLLR adapting on 100+ s of data. In order to learn richer Gaussian usage patterns from the training data, the NMF-based weight adaptation is combined with vocal tract length normalization (VTLN) and speaker adaptive training (SAT), or with a simple Gaussian exponentiation scheme that lowers the dynamic range of the Gaussian likelihoods. Evaluation on the Wall Street Journal tasks shows a 5% relative word error rate (WER) reduction over the speaker independent recognition system which already incorporates VTLN. The WER can be lowered further by combining weight adaptation with Gaussian mean adaptation by means of eigenvoice speaker adaptation.

Keywords: Speaker adaptation, NMF, SAT, fMLLR, eigenvoice

*Corresponding author. Tel: +32-16-321829; fax: +32-16-321723

Email addresses: {Xueru.Zhang}@esat.kuleuven.be (Xueru Zhang),
{Kris.Demuynck}@esat.kuleuven.be (Kris Demuynck),
{Hugo.Vanhamme}@esat.kuleuven.be (Hugo Van hamme)

¹This work is funded by the Dutch-Flemish IMPact program (NWO-iMinds).

1. Introduction

Mismatches between training and testing conditions can severely degrade the performance of an automatic speech recognition (ASR) system. There are two main causes for such mismatches.

The first cause of mismatch is a change in the characteristics of the recording environment, which could be a consequence of using different microphones, different room acoustics or recording under different background noise conditions. In theory, the effect a change in the environment (additive and convolutive noise) has on the speech features can be modeled and hence the negative impact of a change in environment can be counteracted by either normalizing the training and testing condition, i.e. transforming to some standard environment, or by adjusting the acoustic model. The literature contains ample examples of both feature and model based environment compensation techniques; see for example Huang et al. (2001), chapter 10.

The second main cause of mismatch is the speaker variability. This variability can be related to inter-speaker differences such as gender, vocal tract length, dialect and speaking style. The effect of differences in vocal tract length on the acoustic features can be calculated, and hence feature normalization – more specifically vocal tract length normalization (Tuerk and Robinson, 1993) – can be applied. However, most other sources of speaker variability are complex in nature and no straightforward normalization or model compensation techniques exist. Hence, the adaptation techniques must either learn speaker transformations from training examples or will require quite some data to adjust the large set of relevant parameters in the acoustic model. This paper presents a novel method to learn speaker transformations so that the acoustic model can be quickly adapted to a target speaker.

Given sufficient amounts of speaker specific training data, *speaker dependent* (SD) speech recognizers generally outperform their *speaker independent* (SI) counterparts (Huang et al., 2001). However, for most applications, only limited amounts of speaker dependent data are available, insufficient to generate a reliable speaker dependent system. Some example applications that are in this situation are speech based automatic vending machines, automated telephone services, or speech controlled appliances. In such applications, only a few seconds of speech from a specific speaker are available to the speech recognizer. Under these circumstances, *rapid speaker adaptation* forms an attractive solution. Speaker adapted models transform the SI acoustic model so that, given some limited amounts of example data from that speaker, the adjusted acoustic model better describes the target

speaker’s speech.

Speaker adaptation techniques can also reduce the complexity of the speech recognition system by employing acoustic models with fewer parameters than an SI system would require, a property that is quite appealing to some real applications with computational complexity constraints (Woodland, 2001).

Speaker adaptation techniques can be characterized by the following aspects:

1. Fast adaptation, i.e. can an acoustic model that approximates the SD model be produced from limited enrollment data ?
2. Generalization, i.e. can the model parameters (context-dependent phone distributions) for which no or little relevant enrollment data are observed (henceforth ”unseen parameters”) be derived from those model parameters that were observed (”seen parameters”) ?
3. Susceptibility to overfitting when only small amounts of enrollment data are available (Nguyen, 1998). This implies that the speaker adaptation algorithm should aim to get closer to the real underlying SD acoustic model instead of just fitting the limited amounts of observed data optimally.
4. Convergence to the SD model: with large amounts of enrollment data, the speaker adapted model should be at least as good as the SD model.

This work focuses on model-based fast speaker adaptation techniques to find new Gaussian Mixture Model (GMM) state emission distributions which describe the speech of a target speaker well, given limited amounts of enrollment data. In the last decades, several model-based adaptation techniques have been investigated and the most commonly used ones and those relevant for the remainder of this work are now listed.

Maximum a posteriori (MAP) adaptation (Gauvain and Lee, 1994; Woodland, 2001; Zavaliagkos and Schwartz, 1996) maximizes the posterior probability of the model parameters given the enrollment data, with the SI acoustic model parameters used as priors. Given infinite amounts of enrollment data, MAP estimates converge to the maximum likelihood (ML) solution. When the Gaussian mixture weights are also adapted in addition to the mean and variance, MAP will ultimately find the SD state emission distributions. For finite enrollment data, overfitting is counteracted by incorporating the SI information as the prior. A disadvantage of the MAP algorithm is that the generalization of the model is not guaranteed: only the observed model parameters are updated, and the unseen parameters retain their corresponding SI model parameter values. Therefore, large amounts

of enrollment data are required to provide sufficient evidence for estimating every emission distribution separately. MAP also has no knowledge on what typical inter-speaker differences are, and hence adaptation is quite slow.

Eigenvoice speaker adaptation (Kuhn et al., 2000) expresses the means of the Gaussians in the acoustic model as linear combinations of a small number of eigenvoices of Gaussian means. The eigenvoices are learned offline by singular value decomposition of a matrix containing the SD Gaussian means of the individual training speakers. By exploiting the correlations between Gaussian mean estimates across the different training speakers as encoded in the eigenvoices, this method can, based on very small amounts of enrollment data, infer the Gaussian means reliably for both seen and unseen distributions. Combination with MAP allows the acoustic model to converge to the true SD acoustic model with infinite amounts of enrollment data. By relying on a pre-trained model that describes what typical inter-speaker differences are, eigenvoices allow very fast adaptation. However, it is suboptimal in the sense that the Gaussian mixture weights are not adapted.

Unconstrained and constrained (feature-space) maximum likelihood linear regression ((f)MLLR) (Leggetter and Woodland, 1994; Gales, 1998) estimates speaker specific linear transformations of the Gaussian means and corresponding transformations of the variances by maximizing the likelihood of the enrollment data. Generalization is largely dependent on whether a linear transformation is a good model to characterize speakers. Since this is not the case, adaptation with large amounts of enrollment data requires multiple transformations (e.g. organized in regression classes as in (Leggetter and Woodland, 1995)) to approach SD accuracy. For adaptation from limited amounts of enrollment data, the number of free parameters in the linear transformations must be limited to avoid overfitting, for example by using eigenspaces (Chen et al., 2000) or using a diagonal scaling matrix (Leggetter and Woodland, 1994). Finally, fMLLR can only approximate the true SD emission distributions because it does not adapt the Gaussian mixture weights.

The methods mentioned above all focus on transformations in the model space of either the Gaussian means or both the Gaussian means and variances. By contrast, in this paper, speaker-specific state distributions are learned in the model space of Gaussian mixture weights to achieve fast speaker adaptation. Similar to eigenvoice speaker adaptation, the Gaussian mixture weights are expressed as a linear combination of a set of latent vec-

tors. These vectors encode the typical Gaussian usage patterns² as learned from the training speakers. Whereas in Duchateau et al. (2008) the latent vectors were obtained with non-negative matrix factorization (NMF) (Lee and Seung, 1999, 2001) of the SD mixture weights, we opt to construct the latent vectors so that their span³ maximizes the likelihood of the training data, similar to what is being done in probabilistic latent semantic analysis (PLSA) (Hofmann, 1999a). By exploiting the learned patterns, the method can infer the Gaussian mixture weights reliably for both seen and unseen distributions. Like eigenvoices, it provides good generalization, adaptation from small amounts of enrollment data and using a limited number of latent vectors makes it insusceptible to overfitting. Unlike eigenvoices, estimating the cumulative Gaussian posteriors (the input for the weight adaptation) requires less memory and computation overhead than forming the high dimensional Gaussian mean supervectors and the resulting latent vectors are more compact than eigenvoices, which is beneficial for devices with resource limitations. Given the tendency of modern hidden Markov model (HMM) systems to use large Gaussian mixtures to model the emission probability density distributions, weight adaptation is also surprisingly flexible: unlike fMLLR and the eigenvoice method, NMF-based weight adaptation allows complex non-linear redistributions of the probability mass. However, in order to converge to the true SD acoustic model with infinite amounts of enrollment data, combination with a scheme that adjusts the Gaussian means and variances is still needed. Hence, the effectiveness of weight adaptation in combination with vocal tract length normalization (VTLN), adaptation of the Gaussian means and variances by means of fMLLR, mean-based eigenvoice speaker adaptation and speaker adaptive training (SAT) (Anastasakos et al., 1997) are also investigated in this paper. In fact, SAT combined with good speaker normalization and adaptation schemes results in more active Gaussians per speaker and hence makes it easier to discover the distinctive Gaussian usage patterns. It will be shown that the same effect can also be achieved with a simple Gaussian exponentiation scheme that lowers the dynamic range of the Gaussian likelihoods. The combination with the eigenvoice method is preferential since it preserves the "fast adaptation"

²The phrase "Gaussian usage" or "Gaussian activation" refers to the weights associated to the Gaussians in the emission densities or when viewed as a generative model: how many observations were generated for each Gaussian in the mixtures. The term "Gaussian usage/activation patterns" refers to the relations observed in the Gaussian usage for different speakers.

³Subject to the fact that the weights must form a proper probability distribution.

property.

The remainder of this paper is organized as follows. In section 2, we describe the Gaussian mixture weight speaker adaptation algorithm and its relation with non-negative matrix factorization and probabilistic latent semantic analysis. In section 3, we describe how the speaker dependent Gaussian mixture weights are combined with SAT based on speaker dependent fMLLR matrices and a single shared set of Gaussians. The Gaussian exponentiation scheme to facilitate the detection of relevant Gaussian usage patterns is explained in section 4. The combination of the proposed weight-based speaker adaptation algorithm with eigenvoice speaker adaptation is handled in section 5. The recognition system used for the experiments and a comparison of the recognition results with different speaker adaptation algorithms is described in section 6. In section 7, conclusions and possible future research topics are presented.

2. Gaussian mixture weight adaptation using non-negative matrix factorization

Figure 1 gives a schematic overview of the main steps in the NMF-based speaker adaptation system. In the *train* stage, Gaussian usage patterns – called latent speaker vectors – are learned. First, the cumulative Gaussian posterior probabilities γ_r , $r = 1 \dots R$ are collected for the R different speakers in the training database. Based on the collected statistics, a low rank matrix $\mathbf{W}$ containing the latent vectors and the corresponding latent vector coefficients $\mathbf{H} = [\mathbf{h}_1 \dots \mathbf{h}_R]$ are estimated so that they are in an ML sense optimal for estimating the SD Gaussian mixture weights λ_r for the training speakers: $\lambda_r = \mathbf{W}\mathbf{h}_r$. In the *test* stage, the necessary statistics γ_e corresponding to the target speaker e are estimated, either based on transcribed enrollment data or on transcriptions provided by the recognition system itself (unsupervised speaker adaptation). Based on the estimated statistics, the optimal latent vector coefficients $\mathbf{h}_e$ and hence Gaussian mixture weights λ_e can be derived.

Before working out NMF-based speaker adaptation in more detail, we shortly review NMF, PLSA and other related techniques.

2.1. Non-negative matrix factorization

Matrix factorization algorithms decompose a (large) matrix $\mathbf{V}$ into the product of two lower rank matrices $\mathbf{W}$ and $\mathbf{H}$:

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} \quad (1)$$

Figure 1: Diagram of the proposed NMF speaker adaptation algorithm.

The rationale behind this operation is that if a matrix decomposition can be found that is both compact and accurate, the (underlying) structure in $\mathbf{V}$ has been discovered automatically.

If $\mathbf{V}$ is structured column-wise with each column containing one example of a certain entity, then after the decomposition, matrix $\mathbf{W}$ contains a set of vectors for reconstructing the columns in $\mathbf{V}$ and matrix $\mathbf{H}$ contains the vector coefficients. In non-negative matrix factorization, all elements in $\mathbf{V}$, $\mathbf{W}$ and $\mathbf{H}$ are constrained to be non-negative. The factorization now learns the additive parts based representation of an object (Lee and Seung, 1999), where the parts are given by the columns of matrix $\mathbf{W}$ and their activation is given by the corresponding rows of matrix $\mathbf{H}$.

If the columns of $\mathbf{V}$ can be viewed as observations generated by a factorial probabilistic model $\mathbf{\Lambda}$, one should change the aim of the matrix factorization slightly: instead of finding the best decomposition of $\mathbf{V}$ (see equation (1)), one should now try to find the matrices $\mathbf{W}$ and $\mathbf{H}$ so that the probabilistic model $\mathbf{\Lambda} = \mathbf{WH}$ optimally explains the observations:

$$\hat{\mathbf{W}}, \hat{\mathbf{H}} = \arg \max_{\mathbf{W}, \mathbf{H}} P(\mathbf{V}|\mathbf{\Lambda}), \quad \mathbf{\Lambda} = \mathbf{WH} \quad (2)$$

The above described probabilistic matrix factorization scheme has received different names, depending on the application domain. When operating on texts ($\mathbf{V}$ contains word counts) the terms probabilistic latent semantic analysis (PLSA) (Hofmann, 1999a) and probabilistic latent semantic indexing (PLSI) (Hofmann, 1999b) are used. A more generic name is probabilistic latent component analysis (PLCA) (Smaragdis et al., 2006). Furthermore,

Figure 2: Emission probability with respect to latent variable layers.

it has been noticed that NMF which minimizes the generalized Kullback-Leibler divergence between the left and right-hand side of equation (1) is equivalent to PLSA/PLCA which estimates its parameters via the maximum likelihood algorithm (Gaussier and Goutte, 2005). As such, the term NMF can be used when referring to the probabilistic variant as well. In this paper, we adopt this last naming convention.

NMF has been applied in different fields: for example, text mining (Xu et al., 2003) where a term document matrix is decomposed into a term-topic and a topic-document matrices; source separation (Virtanen, 2007; Smaragdis and Brown, 2003) where the spectrogram of the mixture signal is factorized; image processing (Lee and Seung, 2001); speech recognition (Van hamme, 2008) and so on.

When compared to least squares based methods, the NMF-based techniques have the advantage that they are either theoretically sound or that they require few approximations on probabilistic data. Their main disadvantages are that no closed form solutions can be found and that the optimization problem has multiple local minima. SVD and eigenvalue based methods on the other hand have a single global optimum, but require several approximations when dealing with probabilities and counts.

2.2. Estimating the latent vectors

For the NMF-based weight adaptation, the speaker dependent Gaussian mixture weights $\lambda_{r;sk}$, with r , s and k being the speaker, state and Gaussian

component index respectively, are decomposed into a linear combination of latent speaker vectors $\mathbf{W}$:

$$\lambda_{r;sk} = \sum_l w_{(s,k),l} h_{l,r} \quad (3)$$

In the above equation, $h_{l,r}$ refers to the element at row l and column r in matrix $\mathbf{H}$; $w_{(s,k),l}$ refers to the element at row (s,k) and column l in matrix $\mathbf{W}$ with the subscript (s,k) indicating that state and Gaussian mixture component are flattened into one new row index. The decomposition is subject to the following constraints on $\mathbf{W}$ and $\mathbf{H}$ to assure proper emission probability distributions:

$$\begin{cases} \sum_k w_{(s,k),l} = 1, w_{(s,k),l} \geq 0, \forall s, \forall l \\ \sum_l h_{l,r} = 1, h_{l,r} \geq 0, \forall l \end{cases} \quad (4)$$

Figure 2 illustrates that NMF decomposition of the Gaussian mixture weights is tantamount to emission probability densities described by a mixture of Gaussian mixture models. The state of the resulting model when generating an output vector can be described by three hidden variables each with their own probability distribution:

- a state index s , which is governed by the transition probabilities a_{ij} ,
- a latent speaker vector index l with speaker dependent priors $\mathbf{h}_r$, and
- a Gaussian mixture component index k with state and latent vector dependent priors that are stored in $\mathbf{W}$.

The joint probability of generating an observable output sequence $\mathbf{O}_r = \mathbf{o}_1 \cdots \mathbf{o}_T$ for a given speaker r in combination with hidden sequences $\mathcal{S} = s_1 \cdots s_T$, $\mathcal{L} = l_1 \cdots l_T$ and $\mathcal{K} = k_1 \cdots k_T$ equals:

$$p(\mathbf{O}_r, \mathcal{S}, \mathcal{L}, \mathcal{K} | \Phi) = a_0 \prod_{t=1}^T a_{s_{t-1}s_t} w_{(s_t, k_t), l_t} h_{l_t, r} \mathcal{N}_{s_t k_t}(\mathbf{o}_t) \quad (5)$$

with $\mathcal{N}(\cdot)$ a single Gaussian density function and Φ the current set of acoustic model parameters. For brevity of notation, the Gaussian component dependent mean and variances are omitted. Note that equation (5) and the derivations below hold for both HMM systems with state specific Gaussians and for HMM systems that to some extent tie Gaussians between states.

As was stated earlier, the model parameters for the NMF-based rapid speaker adaptation, i.e. the latent speaker vectors $\mathbf{W}$ and corresponding

speaker dependent coefficients $\mathbf{H}$ are trained in maximum likelihood sense. In other words, during the *training* stage of the NMF-based rapid speaker adaptation, the (speaker dependent) model parameters are estimated by maximizing the likelihood of the observations for all training speakers:

$$\begin{aligned}\hat{\Phi} &= \arg \max_{\Phi} P(\mathbf{O}|\Phi) = \arg \max_{\Phi} \prod_{r=1}^R P(\mathbf{O}_r|\Phi) \\ &= \arg \max_{\Phi} \prod_{r=1}^R \left(\sum_{\mathcal{S}, \mathcal{L}, \mathcal{K}} p(\mathbf{O}_r, \mathcal{S}, \mathcal{L}, \mathcal{K}|\Phi) \right)\end{aligned}\quad (6)$$

with $\hat{\Phi}$ the re-estimated set of acoustic model parameters. The Expectation-Maximization (EM) approach (Dempster et al., 1977; Huang et al., 2001) states that maximizing equation (6) is equivalent to maximizing the following auxiliary function:

$$\begin{aligned}Q(\Phi, \hat{\Phi}) &= \sum_r \sum_{\mathcal{S}, \mathcal{L}, \mathcal{K}} p(\mathcal{S}, \mathcal{L}, \mathcal{K}|\mathbf{O}_r, \Phi) \log p(\mathbf{O}_r, \mathcal{S}, \mathcal{L}, \mathcal{K}|\hat{\Phi}) \\ &= \sum_r \sum_{\mathcal{S}, \mathcal{L}, \mathcal{K}} p(\mathcal{S}, \mathcal{L}, \mathcal{K}|\mathbf{O}_r, \Phi) \left\{ \log \hat{a}_0 + \sum_t \log \hat{a}_{s_{t-1}s_t} + \right. \\ &\quad \left. \sum_t \log \hat{\mathcal{N}}_{s_t k_t}(\mathbf{o}_t) + \sum_t \log \hat{w}_{(s_t, k_t), l_t} + \sum_t \log \hat{h}_{l_t, r} \right\} \\ &= Q^a(\Phi, \hat{a}) + Q^b(\Phi, \hat{\mathcal{N}}) + Q^W(\Phi, \hat{w}) + Q^H(\Phi, \hat{h})\end{aligned}\quad (7)$$

with $Q^a(\Phi, \hat{a})$, $Q^b(\Phi, \hat{\mathcal{N}})$, $Q^W(\Phi, \hat{w})$, and $Q^H(\Phi, \hat{h})$ independent terms depending on the transition probabilities, the single Gaussian emission probabilities, the latent speaker vectors $\mathbf{W}$ and the latent speaker vector coefficients $\mathbf{H}$ respectively. Note that the partial auxiliary functions $Q^a(\Phi, \hat{a})$ and $Q^b(\Phi, \hat{\mathcal{N}})$ remain unchanged w.r.t. those obtained for a standard HMM training. Hence, if all model parameters are to be updated, the standard update equations for transition probabilities and Gaussian mean and variances can be applied. Since these are kept fixed in this work, only $Q^W(\Phi, \hat{w})$, and $Q^H(\Phi, \hat{h})$ need to be optimized:

$$\begin{cases} Q^W(\Phi, \hat{w}) &= \sum_r \sum_t \sum_{s, l, k} p(s_t = s, l_t = l, k_t = k|\mathbf{O}_r, \Phi) \log \hat{w}_{(s, k), l} \\ Q^H(\Phi, \hat{h}) &= \sum_r \sum_t \sum_{s, l, k} p(s_t = s, l_t = l, k_t = k|\mathbf{O}_r, \Phi) \log \hat{h}_{l, r} \end{cases}\quad (8)$$

Re-estimating the speaker dependent Gaussian mixture weights (without decomposing the Gaussian mixture weights) only requires the sum of the

posterior probabilities $\gamma_{r;sk}(t)$, with $\gamma_{r;sk}(t)$ denoting the posterior of the k th Gaussian for speaker r and state s at time t . These values can be readily calculated with the forward-backward algorithm (Baum, 1972; Huang et al., 2001), and can be expressed in function of the state posteriors $\gamma_{r;s}(t)$, i.e. the posterior for speaker r and state s at time t , as follows:

$$\begin{aligned}\gamma_{r;sk}(t) &= p(s_t = s, k_t = k | \mathbf{O}_r, \Phi) \\ &= \gamma_{r;s}(t) \frac{\lambda_{r;sk} \mathcal{N}_{sk}(\mathbf{o}_t)}{\sum_{k'} \lambda_{sk'} \mathcal{N}_{sk'}(\mathbf{o}_t)} \\ &= \gamma_{r;s}(t) \frac{\sum_l w_{(s,k),l} h_{l,r} \mathcal{N}_{sk}(\mathbf{o}_t)}{\sum_{k'} \sum_{l'} w_{(s,k'),l'} h_{l',r} \mathcal{N}_{sk'}(\mathbf{o}_t)}\end{aligned}\quad (9)$$

Updating $\mathbf{W}$ and $\mathbf{H}$ requires the posterior probabilities $\gamma_{r;slk}(t)$, i.e. the posterior of the k th Gaussian for state s at time t with latent speaker vector l of training speaker r , which can be calculated as follows:

$$\begin{aligned}\gamma_{r;slk}(t) &= p(s_t = s, l_t = l, k_t = k | \mathbf{O}_r, \Phi) \\ &= \gamma_{r;s}(t) \frac{w_{(s,k),l} h_{l,r} \mathcal{N}_{sk}(\mathbf{o}_t)}{\sum_{k'} \sum_{l'} w_{(s,k'),l'} h_{l',r} \mathcal{N}_{sk'}(\mathbf{o}_t)} \\ &= \frac{\gamma_{r;sk}(t)}{(WH)_{(s,k),r}} w_{(s,k),l} h_{l,r}\end{aligned}\quad (10)$$

Or, after summation over t :

$$\gamma_{r;slk} = \sum_t \gamma_{r;slk}(t) = \frac{\gamma_{r;sk}}{(WH)_{(s,k),r}} w_{(s,k),l} h_{l,r} \quad (11)$$

Hence, only the cumulative posteriors $\gamma_{r;sk}$, i.e. the same statistics as needed to update the speaker dependent Gaussian mixture weights $\lambda_{r;sk}$, need to be stored during the training.

Filling equation (10) for the posterior probability into the auxiliary functions $Q^W(\Phi, \hat{w})$, and $Q^H(\Phi, \hat{h})$ from equation (8) and adding the constraints from equation (4) as Lagrange multipliers gives:

$$\begin{cases} Q(\Phi, \hat{w}) &= \sum_{s,l,k} \sum_r \gamma_{r;slk} \log \hat{w}_{(s,k),l} + \sum_{s,l} \eta_{sl}^W (1 - \sum_k \hat{w}_{(s,k),l}) \\ Q(\Phi, \hat{h}) &= \sum_{s,l,k} \sum_r \gamma_{r;slk} \log \hat{h}_{l,r} + \sum_r \eta_r^H (1 - \sum_l \hat{h}_{l,r}) \end{cases} \quad (12)$$

Differentiating equation (12) with respect to $\hat{w}_{(s,k),l}$ and $\hat{h}_{l,r}$ and setting them to zero gives the following update rules:

$$\begin{cases} \hat{w}_{(s,k),l} &= \frac{1}{\eta_{sl}^W} w_{(s,k),l} \sum_r \frac{\gamma_{r;sk} h_{l,r}}{(WH)_{(s,k),r}} \\ \hat{h}_{l,r} &= \frac{1}{\eta_r^H} h_{l,r} \sum_{s,k} \frac{\gamma_{r;sk} w_{(s,k),l}}{(WH)_{(s,k),r}} \end{cases} \quad (13)$$

with η_{sl}^W and η_r^H normalization terms which ensure that the constraints from equation (4) are fulfilled.

The above derived multiplicative updates (MU) are the same as those obtained when minimizing the generalized Kullback-Leibler divergence between a matrix $\mathbf{V}$ with elements $v_{(s,k),r} = \gamma_{r;sk}$ and the matrix product $\mathbf{WH}$ (Lee and Seung, 2001), except for the extra state and column-wise L_1 normalization of $\mathbf{W}$ and column-wise L_1 normalization of $\mathbf{H}$ after each iteration. The Kullback-Leibler divergence is given as:

$$D(\mathbf{V}||\mathbf{WH}) = \sum_{s,k,r} \left\{ v_{(s,k),r} \log \frac{v_{(s,k),r}}{(\mathbf{WH})_{(s,k),r}} - v_{(s,k),r} + (\mathbf{WH})_{(s,k),r} \right\} \quad (14)$$

2.3. Estimating the target speaker Gaussian mixture weights

Given the latent speaker matrix $\mathbf{W}$ obtained by iterating over equation (13), the Gaussian mixture weights $\lambda_{e;sk}$ for component k in state s of the speaker adapted model for target speaker e can be expressed as a linear combination of the latent speaker vectors:

$$\lambda_{e;sk} = \sum_l w_{(s,k),l} h_{e;l} \quad (15)$$

with $\mathbf{h}_e$ the latent speaker vector coefficients of the target speaker e . The latent speaker vector coefficients are estimated by maximizing the likelihood of the enrollment data from the target speaker. Similar to section 2.2, applying the EM algorithm on a single speaker e given the enrollment data for that speaker and given the latent vectors $\mathbf{W}$ leads to auxiliary function equation (8). Maximizing equation (8) results in the following iterative multiplicative update rule for the latent speaker coefficients:

$$\hat{h}_{e;l} = \frac{1}{\eta_e^H} h_{e;l} \sum_{s,k} \frac{\gamma_{e;sk} w_{(s,k),l}}{\sum_{l'=1}^L w_{(s,k),l'} h_{e;l'}} \quad (16)$$

with η_e^H a normalization term to ensure that $\mathbf{h}_e$ is L_1 normalized.

3. Combining NMF with fMLLR and speaker adaptive training

Whereas most other speaker adaptation schemes adjust the Gaussian means and variances, the proposed NMF-based speaker adaptation scheme focuses on the Gaussian mixture weights. Hence, the question raises whether NMF-based weight adaptation can be augmented with mean and variance adaptation, and continuing on that line of reasoning, whether NMF-based weight adaptation has benefits in combination with speaker adaptive training. In this section, Gaussian mean and variance adaptation by means of fMLLR as described in (Gales, 1998) are considered. This choice was prompted by the fact that fMLLR is the most commonly used technique and the fact that SAT with fMLLR is straightforward.

Speaker adaptive training improves the performance of speech recognition systems by reducing the inter-speaker variation and meanwhile more accurately representing the phonetic variations in the training data (Anastasakos et al., 1996, 1997). The speaker adaptation technique is therefore applied during the training of the acoustic model, where it either transforms speaker specific training data to resemble that of the average speaker as well as possible or it transforms a common acoustic model to speaker specific versions. When using fMLLR for example, the features undergo a speaker dependent linear transformation so that the speech of different speakers can be optimally represented by a common set of Gaussians, or equivalently, the common set of Gaussians is made speaker dependent by means of a linear transformation. This makes that the Gaussian mixtures can focus on modeling the phonetic variations more accurately instead of modeling inter-speaker variations.

Figure 3 gives a schematic overview of the speaker adaptive training scheme with the different model parameters. Algorithm 1 explains the SAT model parameter estimation in detail. Let $\mu, \Sigma, \mathbf{M}$ represent the Gaussian mean vectors, variance matrices, and the fMLLR linear transformation matrix respectively. Subscripts SI and SAT represent the SI acoustic model and the SAT estimated acoustic model respectively.

During training, speaker specific fMLLR matrices $\mathbf{M}_r$ are estimated for all speakers in the training database. $\mathbf{M}_{\text{SAT}}$ is the fMLLR matrix trained on the data of all the training speakers, i.e. the speaker independent transformation matrix that maximizes the observed data from all training speakers. The SD fMLLR matrices $\mathbf{M}_r$ transform the shared speaker adaptive acoustic model parameters μ_{SAT} and Σ_{SAT} to speaker dependent variants.

The shared model parameters λ_{SAT} , μ_{SAT} and Σ_{SAT} are updated so that they maximize the observed data given the SD fMLLR matrices. The

Figure 3: Overview of the proposed NMF speaker adaptation algorithm combined with SAT.

Gaussian mixture weights λ_r are speaker dependent, i.e. they are not derived from common model parameters. However, a global speaker independent set of Gaussian mixture weights λ_{SAT} is estimated as well.

As can be seen from algorithm 1, the SAT model parameters $\{\lambda_r\}_{r=1}^R$, λ_{SAT} , μ_{SAT} , Σ_{SAT} , and $\{\mathbf{M}_r\}_{r=1}^R$ are trained in a nested loop using maximum likelihood re-estimation. The outer loop optimizes the feature-space transformation matrices. In *Step 2*, fMLLR is used to estimate the SD matrices $\mathbf{M}_r$ based on the data of the individual training speakers and based on the current estimate of the SAT model, i.e λ_r , μ_{SAT} , and Σ_{SAT} . It has been proven in (Gales, 1998) that feature space linear transformation is equivalent to the mean and variance transformation in the model space:

$$\bar{\mathbf{o}}_t = \mathbf{A}\mathbf{o}_t + \mathbf{b} = \mathbf{M}\zeta_t \quad (17)$$

with $\mathbf{M} = [\mathbf{b} \ \mathbf{A}]$ and $\mathbf{A}$ a full transformation matrix, and ζ_t the observation vector $\zeta_t = [\mathbf{1}, \mathbf{o}_t^T]^T$ (superscript T here represents vector transpose). For reasons we will explain later on, we also estimate a common fMLLR transformation matrix $\mathbf{M}_{\text{SAT}}$ on all speakers jointly using μ_{SAT} , Σ_{SAT} and λ_{SAT} , the latter being the common mixture weights as formed in *Step 3.2*.

Transforming the observation vectors ζ_t with $\mathbf{M}$ allows the inner loop to use the standard EM algorithm to update the Gaussian distributions and mixture weights (*Steps 3.1+3.2*). In *Step 3.1*, the update rules (EM) are given as follows:

$$\lambda_{r;sk} = \frac{\gamma_{r;sk}}{\sum_{k'} \gamma_{r;sk'}} \quad (18)$$

Algorithm 1 SAT in the combined speaker adaptation algorithm.

Step 1 : Initialize:

$\lambda_{\text{SAT}} = \lambda_{\text{SI}}, \mu_{\text{SAT}} = \mu_{\text{SI}}, \Sigma_{\text{SAT}} = \Sigma_{\text{SI}}, \{M_r\}_{r=1}^R = [\mathbf{0}, \mathbf{I}], \{\lambda_r\}_{r=1}^R = \lambda_{\text{SI}}$
for $1 \cdots N_o$ **do**

Step 2 : Estimate $\{M_r\}_{r=1}^R$ using fMLLR so that the likelihood of the observation data is maximized, given $\{\lambda_r\}_{r=1}^R, \mu_{\text{SAT}}, \Sigma_{\text{SAT}}$.

Step 3 : Update $\{\lambda_r\}_{r=1}^R, \mu_{\text{SAT}}, \Sigma_{\text{SAT}}, \{\gamma_r\}_{r=1}^R$. The symbol γ_r is used as shorthand for $\gamma_{r;sk}$, i.e. the complete set of Gaussian posterior probabilities of training speaker r .

for $1 \cdots N_i$ **do**

Step 3.1 : Estimate $\{\lambda_r\}_{r=1}^R, \mu_{\text{SAT}}, \Sigma_{\text{SAT}}, \{\gamma_r\}_{r=1}^R$ using EM so that the likelihood of the observation data is maximized, given $\{M_r\}_{r=1}^R$.

Step 3.2 : Update $\lambda_{\text{SAT};sk} = \frac{\sum_r \gamma_{r;sk}}{\sum_{k'} \sum_r \gamma_{r;sk'}}$

end for

end for

$$\mu_{\text{SAT};k} = \frac{\sum_r \sum_t \gamma_{r;k}(t) \bar{\mathbf{o}}_t}{\sum_r \gamma_{r;k}} \quad (19)$$

$$\Sigma_{\text{SAT};k} = \frac{\sum_r \sum_t \gamma_{r;k}(t) (\bar{\mathbf{o}}_t - \mu_{\text{SAT};k})(\bar{\mathbf{o}}_t - \mu_{\text{SAT};k})^T}{\sum_r \gamma_{r;k}} \quad (20)$$

with $\gamma_{r;k}(t)$ the posterior probability of Gaussian component k at time t for speaker r .

4. Creating expressive latent speaker vectors

In order for the NMF-based speaker adaptation scheme to be effective, the latent speaker vectors $\mathbf{W}$ must encode the systematic patterns of variation between speakers. The effect of some inter-speaker differences on the acoustics, for example vocal tract length (VTL), is well known and easy to model. As was shown in (Duchateau et al., 2008), NMF-based weight adaptation can effectively cope with VTL differences. However, if both the Gaussians in the acoustic model and the latent speaker vectors in $\mathbf{W}$ must cope with such predictable inter-speaker variation, less modeling power – both in the Gaussians and in the latent speaker vectors – is available to represent other less predictable but important phonetic variations. Hence, a first step towards creating expressive latent speaker vectors is an adequate feature normalization. In this paper, we use both spectral mean-normalization and VTLN (Duchateau et al., 2006) to that end.

Continuing on the above line of reasoning, we can expect that the fMLLR based speaker normalizations used during the speaker adaptive training have a similar effect. The reduction in inter-speaker variability not only allows the Gaussians to more accurately represent the phonetic variations in the training data (Anastasakos et al., 1996, 1997), but also allows the latent speaker vectors to focus on those aspects which cannot be solved with a linear transformation. The effect of this will be further investigated in section 6. Combining NMF-based weight adaptation and fMLLR also brings more flexibility into the adaptation scheme. Whereas NMF-based weight adaptation is limited to the model space encoded by the latent speaker vectors (interpolation), fMLLR can adapt the acoustic model to situations not observed in the training data (extrapolation). However, this added flexibility comes at a cost: firstly, there is an increased susceptibility to overfitting on small amounts of adaptation data and a risk for divergence when using unsupervised adaptation, and secondly, the adaptation speed is now in theory limited to that of fMLLR.

A third important rule for obtaining expressive latent speaker vectors is that the $\mathbf{V}$ matrix must be rich enough so that the NMF decomposition can discover the distinctive Gaussian usage patterns. Due to the non-negative nature of the elements in $\mathbf{V}$ and the use of NMF, zero values carry no information about Gaussian usage. In other words, it is preferred that a wide variety of Gaussians are activated and hence contribute to the cumulative posteriors in order to be able to detect the distinctive correlations between Gaussian activations.

In (Zhang et al., 2011), we noticed that combining NMF-based weight adaptation and SAT leads to a substantially less sparse $\mathbf{V}$ matrix. Given that the SAT and SI Gaussians (see figure 3) are strongly related, we also tried combining the SAT latent speaker vector $\mathbf{W}$ in combination with the SI model λ_{SI} , μ_{SI} , Σ_{SI} to perform the weight adaptation. This resulted in surprisingly good results when compared to the $\mathbf{W}$ from the SI model which was obtained after decomposing the fairly sparse matrix $\mathbf{V}$ from the SI model. This underscores the importance of having a sufficiently dense $\mathbf{V}$ matrix.

Boosting the Gaussian activation so that more relations between Gaussian activations are recorded can also be achieved with a simple Gaussian exponentiation (GE) scheme that lowers the dynamic range of the Gaussian likelihoods. The “likelihood” of an observation is now calculated as follows:

$$p(o_t|s, r, \Phi) = \sum_k \lambda_{r;sk} (\mathcal{N}_{sk}(o_t))^\alpha \quad (21)$$

Figure 4: Diagram of the NMF-based rapid speaker adaptation system combined with the Gaussian exponentiation scheme.

with α a constant in the range $[0, 1]$.

Although $p(o_t|s, r, \Phi)$ is no longer a proper probability distribution, one can still update all model parameters Φ so that the total “log likelihood” over all observations is maximized. The update rules for the Gaussian means and variances are identical to the standard update rules, i.e. $\alpha = 1$. The update rule for the Gaussian mixture weights changes to:

$$\lambda_{r;sk} = \frac{\gamma_{r;sk}^{(\alpha)}}{\sum_{k'} \gamma_{r;sk'}^{(\alpha)}} \quad (22)$$

with

$$\gamma_{r;sk}^{(\alpha)} = \sum_t \frac{\lambda_{r;sk} (\mathcal{N}_{sk}(o_t))^\alpha}{\sum_{k'} \lambda_{r;sk'} (\mathcal{N}_{sk'}(o_t))^\alpha} \gamma_{r;s}(t) \quad (23)$$

The Gaussian mixture weights are initialized with the SI weights: $\lambda_{r;sk} = \lambda_{\text{SI};sk}$.

Figure 4 illustrates the overall structure of the training stage of the NMF-based speaker adaptation system when combined with the Gaussian exponentiation scheme. First, the original SD Gaussian mixture weights $\lambda_{r;sk}$ are re-estimated to accommodate for the Gaussian exponent α by means of several EM iterations. Next, the updated SD cumulative Gaussian posteriors $\gamma_{r;sk}^{(\alpha)}$ are used to formulate the NMF $\mathbf{V}$ matrix which in turn is used to estimate the latent speaker vectors (see section 2). Since all updates are still based on the EM algorithm, the derivation in section 2.2 still holds.

5. Combining NMF with eigenvoice speaker adaptation

Eigenvoice speaker adaptation (Kuhn et al., 2000) is another popular Gaussian mean adaptation algorithm. It allows fast speaker adaptation

by expressing the Gaussian means as a combination of eigenvoices. The eigenvoice method and the NMF-based weight adaptation techniques can be expected to combine very well for several reasons. i) Both eigenvoice speaker adaptation and NMF-based weight speaker adaptation maximize the likelihood of the enrollment data. ii) NMF-based weight speaker adaptation provides an elegant solution to the problem that eigenvoice speaker adaptation technique cannot be readily used to adapt the Gaussian mixture weights. iii) For both techniques, the acoustic models of the evaluation speaker are expressed as linear combinations of a set of latent vectors, which are estimated from the speaker dependent acoustic models of the training speakers. iv) Both techniques are rapid speaker adaptations with a limited number of parameters to be estimated based on the enrollment data.

In eigenvoice speaker adaptation, the set of Gaussian mean vectors μ_e of an evaluation speaker e is expressed as a linear combination of eigenvoices, which are derived from the training speakers. These eigenvoices are estimated as follows. Firstly, the SD mean vectors $\{\mu_r\}_{r=1}^R$ of the R training speakers are estimated. Next, principle component analysis (PCA) is used to estimate the eigenvectors from the covariance or correlation matrix of the training speaker SD mean vectors. The J eigenvectors with the largest corresponding eigenvalues are kept as eigenvoices $\{\phi_j\}_{j=1}^J$ with j the eigenvoice index.

The desired evaluation speaker mean vector μ_e is expressed as a linear combination of the derived eigenvoices.

$$\mu_e = \phi_1 \rho_1 + \sum_{j=2}^J \phi_j \rho_j \quad (24)$$

with weighting coefficients $\{\rho_j\}_{j=2}^J$. The first eigenvoice ϕ_1 is set to the SI mean vector μ_{SI} , with corresponding coefficient $\rho_1 = 1.0$. The other weighting coefficients $\{\rho_j\}_{j=2}^J$ are estimated by maximizing the likelihood of the enrollment data. As is described in (Kuhn et al., 2000), this can be done by solving matrix equation (25):

$$\sum_{s,k,t} \gamma_{e;sk}(t) \phi_{j;sk}^T \Sigma_{\text{SI};sk}^{-1} (\mathbf{o}_t - \phi_1 \rho_{1;sk}) = \sum_{s,k,t} \gamma_{e;sk}(t) \left\{ \sum_{i=2}^J \rho_i \phi_{i;sk}^T \Sigma_{\text{SI};sk}^{-1} \phi_{j;sk} \right\}, \quad \forall j = 2 \cdots J \quad (25)$$

where $\Sigma_{\text{SI};sk}$ is the speaker independent covariance matrix of Gaussian k at state s respectively.

Like the NMF-based weight speaker adaptation algorithm, the eigenvoice speaker adaptation algorithm updates both the seen and unseen model parameters. The relations between the SD means of the training speakers are encoded in the eigenvoices. By exploiting these relations, means of Gaussians not activated during the enrollment phase (unobserved Gaussians) will still be updated by extrapolating their transformation based on those that were observed. Furthermore, the number of degrees of freedom to fit a target speaker equals the number of eigenvoices minus one ($\phi_1 = \mu_{\text{SI}}, \rho_1 = 1.0$). Given that the number of eigenvoices is small, only a few parameters have to be estimated and hence the eigenvoice speaker adaptation technique can be used to perform rapid speaker adaptation given small amounts of enrollment data.

The procedure used for combined eigenvoice and NMF-based weight adaptation in a SAT framework are outlined in Algorithm 2. As outlined in (Zhang et al., 2012) it cascades the eigenvoice and NMF estimation procedures. During evaluation, the eigenvoice weighting coefficients $\{\rho_j\}_{j=2}^J$ and the NMF latent vector coefficients $\mathbf{h}_e$ of the evaluation speakers are estimated by maximizing the likelihood of the enrollment data, given that the adapted means and weights are expressed as linear combinations of latent vectors (Eqs (24) and (15)). Maximizing the likelihood is equivalent to maximizing the auxiliary function using EM.

$$Q(\mathbf{h}_e, \rho_j) = Q(\mathbf{h}_e) + Q(\rho_j) \quad (26)$$

with

$$\begin{aligned} Q(\mathbf{h}_e) &= \sum_{s,k,t} \gamma_{sk}(t) \log(\sum_l w_{(s,k),l} h_{e;l}) \\ Q(\rho_j) &= -\frac{1}{2} \sum_{s,k,t} \gamma_{sk}(t) \{n \log(2\pi) + \log |\Sigma_{\text{SAT};sk}| + \\ &\quad (\mathbf{o}_t - \sum_{j=1}^J \phi_j \rho_j)^T \Sigma_{\text{SAT};sk}^{-1} (\mathbf{o}_t - \sum_{j=1}^J \phi_j \rho_j)\} \end{aligned} \quad (27)$$

with n the feature dimension. The optimum can be found by iterating equation (25) (with $\mu_{\text{SI}} = \mu_{\text{SAT}}$ and $\Sigma_{\text{SI}} = \Sigma_{\text{SAT}}$) and equation (15) until the coefficients converge.

6. Experiments

6.1. Recognition system

The Wall Street Journal (WSJ) corpus is used for training, developing and testing the proposed weight adaptation algorithm. Training is done

Algorithm 2 Deriving the latent vectors for adaptation

Step 1 : Initialize the SD mean vectors $\{\mu_r\}_{r=1}^R$ and Gaussian mixture weights $\{\lambda_r\}_{r=1}^R$ of the training speaker,

$$\mu_r = \mu_{\text{SAT}}; \lambda_r = \lambda_{\text{SAT}} \text{ with } r = 1 \cdots R$$

Step 2 : Estimate the R SD mean vectors $\{\mu_r\}_{r=1}^R$ of the training speakers, using $\mathbf{M}_{\text{SAT}}$ as the transformation matrix.

Step 3 : Reestimate the SD mean vectors using MAP.

$$\mu_{r;sk}^{(\text{MAP})} = \frac{\gamma_{r;sk}}{\gamma_{r;sk} + \Gamma} \mu_{r;sk} + \frac{\Gamma}{\gamma_{r;sk} + \Gamma} \mu_{\text{SAT};sk} \quad (28)$$

where $\gamma_{r;sk}$ is the cumulative Gaussian posterior probability; Γ is the parameter to control the relative weight of the prior μ_{SAT} .

Step 4 : Estimate the eigenvoices $\{\phi_j\}_{j=1}^J$ by applying PCA to the correlation matrix of the Gaussian means $\{\mu_r^{(\text{MAP})}\}_{r=1}^R$.

Step 5 : Estimate the eigenvoice weighting coefficients $\{\rho_j\}_{j=2}^J$ of the training speakers by maximizing the likelihood of the data. The coefficients can be computed from equation (25). This step is iterated until the coefficients $\{\rho_j\}_{j=2}^J$ converge.

Step 6 : Estimate the SD weight vectors $\{\lambda_r\}_{r=1}^R$ of the training speakers using the adapted mean vectors estimated in *Step 5*.

Step 7 : Form the NMF $\mathbf{V}$ matrix. The r th SD weight vector λ_r forms the r th column of $\mathbf{V}$.

Step 8 : Estimate the latent vectors $\mathbf{W}$ using NMF by maximizing the likelihood of the training data.

on the SI-284 data from both WSJ0 and WSJ1 comprising 81 hours from 284 speakers. The baseline speech recognizer used in our experiments is a semi-tied Gaussian mixture HMM system. The system uses a shared pool of 32.754 Gaussians to model the observations in 5967 cross-word context-dependent tied triphone states, using 94 Gaussian probability densities on average per state. All acoustic units — context-dependent variants of one of the 42 phones or silence — have a 3-state left-to-right topology. The acoustic features consist of 22 MEL spectra with mean normalization and VTLN (Duchateau et al., 2006), augmented with their first and second order time derivatives, which results in 66 dimensional feature vectors. These features are then mapped to a 39 dimensional space by means of a discriminative linear transformation and decorrelation (Demuynck, 2001).

For developing and evaluating the system, we combined the WSJ 5k

	development data				evaluation data			
	dev92		dev93		Nov92		Nov93	
corpus	5k	20k	5k	20k	5k	20k	5k	20k
#speakers	10	10	10	10	8	8	10	10
#sentences	410	403	513	503	330	333	215	213
#words	6780	6724	8639	8235	5353	5643	3854	3448
#spch/spkr	220	220	283	274	226	238	122	115
mean $\pm$ std.dev.	± 15	± 16	± 35	± 35	± 18	± 28	± 21	± 19

Table 1: Development and evaluation corpora properties. *#speakers*: total number of speakers. *#sentences*: total number of sentences. *#spch/spkr mean $\pm$ std.dev.* : average amount of speech (excluding silence) in seconds per speaker and standard deviation.

closed and 20k open vocabulary non-verbalized punctuation Nov92 and Nov93 tasks. By combining all evaluation data, we obtained one large evaluation set containing 101 minutes of speech (18.298 words). The combination of all corresponding development data was used to tune system parameters such as pruning thresholds and the weight ratio between the language model and the acoustic model. The properties of the development and evaluation data are given in table 1. Note that each pair of 5k and 20k sub-tasks share the same speakers. This property allowed us to select the enrollment data outside the current sub-task, i.e. draw enrollment data from the 5k sub-task to recognize the sentences from the corresponding 20k sub-task and vice versa. This way, the amount of enrollment data could also be easily varied, up to at least 100 s of data per speaker.

The recognition system uses a 75k lexicon, containing the most frequent words from the 119M word corpus provided with the WSJ0+1 database. This resulted in an out-of-vocabulary ratio of 0.08% – 0.12% on the development and evaluation set respectively. The phonetic transcriptions for the train data, enrollment data and the 75k recognition lexicon were drawn from CMUdict 0.6d. As language model, a standard trigram using modified Kneser-Ney discounting (Chen and Goodman, 1998) trained on the WSJ0+1 119M word corpus was employed.

For unsupervised adaptation, a two-pass scheme is required. Note that in order to be able to compare results, we use the exact same enrollment data for both the supervised and unsupervised adaptation tests. In unsupervised mode, the recognizer first processes the enrollment data, generating both single best word sequences and word lattices. Based on the word posteri-

ors derived from the word lattices, the 30% least likely words are discarded from the single best word sequences. From there on, the recognition proceeds as with supervised adaptation: Viterbi alignment is used to find the best state alignment, which is then used to estimate the Gaussian mixture weight posterior probabilities and/or the fMLLR statistics needed for the adaptation.

Note that when combining SAT and NMF adaptation with only a few seconds of adaptation data, the speaker specific fMLLR estimate $\mathbf{M}_e$ was found to be unreliable and hence was replaced with $\mathbf{M}_{\text{SAT}}$ (see figure 3) during adaptation and evaluation. $\mathbf{M}_e$ was only estimated and used when adapting on all data of a speaker.

For the NMF-based weight adaptation, $\mathbf{W}$ is initialized with the absolute value of standard normally distributed pseudo random numbers. A small value of 0.01 is added to avoid zeros. Matrix $\mathbf{W}$ is then L_1 normalized per column and per state. Matrix $\mathbf{H}$ is initialized as $\mathbf{W}^T \mathbf{V}$ followed by L_1 normalization per column. The vector $\mathbf{h}_e$ is initialized analogous to $\mathbf{H}$. Preliminary experiments using random initialization for $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{h}_e$ showed no significant difference in recognition performance, given a sufficient number of MU iterations.

4000 MU iterations were used to construct the latent vectors $\mathbf{W}$. We applied 3 iterations ($N_0 = N_i = 3$ in Algorithm 1) of EM in the SAT experiment. 10 MU iterations were used to estimate the Gaussian mixture weights during the adaptation phase. These values were optimized on the development set: higher values provide no improvement in WER.

6.2. Experimental results

6.2.1. Word error rate

Table 2 shows the word error rate (WER) in % on the evaluation data for the different speaker adaptation algorithms with SI and SAT acoustic models respectively.

For the NMF decomposition, the number of latent speaker vectors is set to 10 for all but four experiments and the cumulative posteriors corresponding to the 3-state silence model are discarded from $\mathbf{V}$. Hence, the silence states retain their SI/SAT model parameter values. The constant α for the Gaussian exponentiation was set to 0.2 since this gave the best results on the development data.

As can be seen from table 2, supervised NMF-based weight adaptation on the baseline SI acoustic model ($SI+NMF[10]$ and $SI+NMF[40]$ — the number between square brackets is the number of latent speaker vectors) only provides a modest reduction in WER over the SI baseline. This contrasts

adaptation method	amount of adaptation data (s)				
	0	3	6	8	100+
supervised adaptation, SI acoustic models					
	6.42				
NMF[10]		6.27	6.30	6.32	6.28
NMF[40]					6.19
fMLLR		7.02	6.28	6.17	5.97
GE, NMF[10]		6.13	6.08	6.15	
unsupervised adaptation, SI acoustic models					
GE, NMF[10] (30%)		6.13			
supervised adaptation, SAT acoustic models					
	6.37				
NMF[10]		6.06	6.08	6.09	6.01
NMF[40]		6.01	6.06	6.08	6.03
fMLLR		6.96	6.33	6.18	6.01
eigenvoice[10]		5.95	5.99	5.90	5.97
eigenvoice[40]		6.15	6.04	6.00	6.04
fMLLR + NMF[10]		6.79	5.97	5.91	5.68
eigenvoice[10] + NMF[10]		5.87	5.89	5.76	5.81
eigenvoice[40] + NMF[40]		6.01	5.88	5.78	5.69
unsupervised adaptation, SAT acoustic models					
NMF[10] (30%)		6.10			
NMF[10] (0%)		6.02			

Table 2: Evaluation data WER (%) obtained with SI and SAT acoustic model and different speaker adaptation algorithms. The number of latent speaker vectors is given between square brackets in the first column. *100+* : the amount of adaptation data per speaker is around 240 seconds for Nov92 and 100 seconds for Nov93 (see table 1). (*·*%) : percentage of words discarded during unsupervised adaptation.

with the results reported in (Duchateau et al., 2008) which show a 5% to 15% relative improvement in this situation. However, the setup in (Duchateau et al., 2008) did not include VTLN in the front-end preprocessing. Consequently the NMF-based speaker adaptation predominantly adapts the acoustic model to the speaker gender. In other words, in (Duchateau et al., 2008), the NMF-based speaker adaptation process mainly plays the role of VTLN, while in our setup, any observed improvement is an improvement in addition to VTLN.

Figure 5: Distribution of the normalized $\mathbf{V}$ and $\mathbf{W}$ values in the NMF algorithm. The x-coordinate shows the fraction of the sorted weight values. The y-coordinate is the sorted weight value, which is between 0 and 1.

Both the *SI+NMF* and *SAT+NMF* experiments show that there is no substantial improvement when the number of latent speaker vectors is increased from 10 to 40, even when sufficient enrollment data are available. Preliminary experiments to determine the optimal range for the number of latent speaker vectors, were consistent with this observation. The reason could be that the NMF decomposition could not extract more than 10 different distinct speaker characteristics from the training data (284 speakers). This will be further analyzed in section 6.2.3.

Switching to the combination SAT+NMF shows that improvements can be obtained with the NMF-based weight adaptation scheme. The fact that NMF-adaptation also works when only a single enrollment sentence is used (3 to 8 s of adaptation speech) and hence fMLLR-adaptation is bypassed (as stated in section 6.1, $\mathbf{M}_{\text{SAT}}$ is used instead of $\mathbf{M}_e$ in this case), shows that the difference in behaviour between the SI+NMF and SAT+NMF setup is to be attributed primarily to an improved latent vector matrix $\mathbf{W}$. In fact, replacing the $\mathbf{W}$ matrix in the SI+NMF setup with the $\mathbf{W}$ matrix from the SAT setup resulted in a significant improvements as well. In other words, NMF-based weight adaptation works well if high quality latent speaker vectors can be learned. As was stated earlier, the key assumption underlying the NMF adaptation technique is that relevant Gaussian usage patterns can be learned from the training data. With SAT, each emission probability density of the speaker-dependent HMM states will have more active Gaussians, hence providing richer information to the NMF decomposition. This results in better latent vectors. This is illustrated in figure 5. The dash-dot

line in figure 5a shows that with SAT there are more active Gaussians in the $\mathbf{V}$ matrix and after NMF decomposition, more active Gaussians are kept in the $\mathbf{W}$ matrix, which is represented by the dash-dot line in figure 5b. Given the non-negative nature of the values in $\mathbf{W}$, the only way systematic Gaussian usage patterns can be encoded is by having positive values for the respective Gaussians. Hence, having more non-zero elements is a strong indicator that more (relevant) usage patterns are recorded in $\mathbf{V}$. It is also shown in figure 5b that without VTLN the recorded correlations are further reduced to those related to the speaker’s gender.

The results obtained with Gaussian exponentiation (*SI+GE+NMF*) are consistent with this conclusion. As can be seen from the thin solid colored lines in figure 5, applying the Gaussian exponentiation lowers the number of Gaussians that show no activity at all for certain speakers in the train database. Hence, the NMF decomposition can learn more relevant usage patterns which results in a higher quality $\mathbf{W}$ matrix and a decrease in WER compared to the *SI+NMF* setup.

The NMF-based weight adaptation is also compared with fMLLR speaker adaptation. Comparing the different setups shows that fMLLR is sensitive to the amount of adaptation data. With sufficient amounts of enrollment data, fMLLR lowers the SI WER on average by 6%. However, when the amount of adaptation data is limited to 3 s of data, it is no longer possible to reliably estimate the speaker dependent transformation matrix $\mathbf{M}_e$. In fact, fMLLR now significantly deteriorates the recognition performance. NMF-based weight adaptation on the other hand is very robust w.r.t. the amount of adaptation data. Even with only 3 s of adaptation speech, improvements similar to those obtained with fMLLR with 100+ s of data (5% relative) can be observed. When NMF-adaptation is combined with fMLLR, an additional improvement is observed with sufficient amounts of enrollment data.

The NMF-based weight adaptation algorithm and the eigenvoice algorithm alone show similar performance. However, the larger dimensionality of the eigenvoices could make the eigenvoices more susceptible to overfitting when only limited amounts of enrollment data are available. This may explain the increase in WER when using 40 instead of 10 eigenvoices with only 3 s of enrollment data for the eigenvoice system, whereas the weight adaptation (with lower dimensional latent vectors) shows no signs of overfitting. By combining these two methods together, there are consistent improvements over both the eigenvoice and the NMF-based weight adaptation schemes by themselves. For example, with 10 degrees of freedom and 3 s of enrollment data, the performance of the recognition system is improved by 1.3%

relatively over the best single adaptation method, resulting in a 7.9% improvement compared to the SAT baseline. When more degrees of freedom (40) and more enrollment data (100+) are available, this improvement increases to 5.8% compared to the eigenvoice approach and 10.7% compared to the SAT baseline. Furthermore, the combination with less degrees of freedom (*eigenvoice*[10]+*NMF*[10]) gives better results than either the eigenvoice or the NMF-based weight adaptation with more degrees of freedom (*eigenvoice*[40] and *NMF*[40] respectively). We observe that the eigenvoice adaptation keeps the Gaussians that are not needed for a certain training speaker close to their SI positions. This may create a source for undesirable overlap of state densities. The NMF weight adaptation can suppress these Gaussians by assigning a relative small value or zero to the corresponding Gaussian mixture weights. Compared with the *fMLLR*+*NMF*[10] combination scheme, the *eigenvoice*[10]+*NMF*[10] also shows better performance with less enrollment data. When more enrollment data (100+) are available, the *eigenvoice*[40]+*NMF*[40] with 80 degrees of freedom gives similar performance as *fMLLR*+*NMF*[10] with about twenty times more degrees of freedom.

These results show that NMF-based Gaussian mixture weight adaptation and eigenvoice-based mean adaptation are compatible with each other and that this combination can perform fast speaker adaptation. Both adaptation algorithms outperform fMLLR given less enrollment data. NMF-based weight adaptation is complementary to eigenvoice speaker adaptation and fMLLR.

The experiments with unsupervised adaptation show that NMF-adaptation is not affected by a small amount of transcription errors in the enrollment data (unsupervised adaptation data). Even when not removing the words with the lowest confidence score, NMF-adaptation still works admirable. This, in combination with the (very) fast adaptation provided by NMF (see next section) opens up the possibility to implement NMF-based speaker adaptation in a single pass low-latency recognition system.

6.2.2. Adaptation speed and convergence

An extra batch of experiments was conducted to investigate the adaptation speed and convergence of the proposed algorithm in more detail. Stretches of adaptation data ranging from 20 ms to 8 s (excluding silence) were selected. For each length, 40 disjunct segments (containing different phonetic content) of adaptation data were chosen randomly: 20 segments from the 5k sub-task and 20 segments from the 20k sub-task. Figure 6 shows the relation between the obtained WER and the amount of adaptation data.

Figure 6: WER (%) in function of the amount of adaptation data. The dash-dot line marks the WER of the SI model without any adaptation technique. The dashed and solid lines correspond to the supervised *SAT+NMF*[10] and *GE+NMF*[10] adaptation respectively. The thick lines and dots represent the average WER, the thin lines and downward-pointing triangles indicate the limit for the 98% confidence interval (single sided) on the WER.

To quantify the uncertainty on the WER due to choice of enrollment data, the 98% single sided confidence intervals on the word error rates are estimated. A standard normal distribution is used to model the variation on the WER:

$$p\left(\frac{\text{WER} - \mu_{\text{WER}}}{\sigma_{\text{WER}}} \leq \text{CR}_{\text{WER}}\right) = 0.98 \rightarrow \text{CR}_{\text{WER}} = 1.98 \quad (29)$$

with μ_{WER} and σ_{WER} being the mean and standard deviation of the WER over the set of 20 experiments and CR_{WER} the critical value, i.e. in less than 2%, the obtained WER will be higher than $\mu_{\text{WER}} + \text{CR}_{\text{WER}} \times \sigma_{\text{WER}}$.

It can be seen that if at least 0.5 s of adaptation data are available, the proposed algorithms will improve upon the SI (non-adapted) speech recognition system in at least 98% of random drawings of adaptation data. The proposed Gaussian mixture weight adaptation algorithm also converges

very fast: once 3 s or more data have been processed, the dependency of the WER on the phonetic content of the enrollment data or on the amount of enrollment data becomes small in comparison to the average absolute improvement. The plot also shows that NMF-based Gaussian mixture weight adaptation combined with SAT or Gaussian exponentiation gives similar performance.

The fast adaptation and convergence show that (1) NMF is not susceptible to overfitting when only small amounts of enrollment data are available, and (2) NMF generalizes well and hence can work with very small amounts of adaptation data.

6.2.3. Content of the latent speaker vectors

We also investigated the correlations between the latent speaker vectors and some observable speaker characteristics. The analysis was limited to the 200 WSJ1 training speakers for which speaker meta-data are available. The following speaker characteristics (with corresponding classes and number of speakers belonging to each class) were considered: *gender* (male: 100, female: 100), *age* (< 25: 26, < 35: 75, < 45: 55, < 55: 26, ≥ 55: 16), and *region* where the speaker went to primary school (West: 110, the Midlands: 27, Southern: 19, New England: 16, the Inland North: 9, New York City: 6, North Central: 5). The classification was done with a linear classifier ($\mathbf{U}_i$ in equations (30) and (31)) trained on the latent speaker coefficients $\mathbf{H}$ in a leaving-one-out scheme with NMF as described below.

To perform the classification with NMF, a grounding matrix $\mathbf{G}$ is formulated whose columns are binary unit vectors. Taking the gender as an example, the size of $\mathbf{G}$ is $2 \times R$. If speaker r is a female, the corresponding element in the first row of $\mathbf{G}$ is set to 1. Otherwise the element in the second row is set to 1. $\mathbf{H}_{i-}$ and $\mathbf{G}_{i-}$ are derived from $\mathbf{H}$ and $\mathbf{G}$ respectively by leaving the i th column (speaker) out. The gender prediction matrix $\mathbf{U}_i$ is estimated with NMF. In other words, one tries to predict the gender from the latent vector coefficients $\mathbf{H}_{i-}$.

$$\mathbf{G}_{i-} \approx \mathbf{U}_i \mathbf{H}_{i-} \quad (30)$$

The assessed gender vector $\hat{\mathbf{g}}_i$ for speaker i th is calculated as follows.

$$\hat{\mathbf{g}}_i = \mathbf{U}_i \mathbf{h}_i \quad (31)$$

The index of the maximum value in $\hat{\mathbf{g}}_i$ determines the estimated gender of the i th training speaker, i.e. the speaker left out in the leaving-one-out scheme.

meta-data	HMM providing the latent vectors				
	priors	SI, no VTLN	SI, VTLN	SAT, VTLN	GE, VTLN
gender	50	99	94	95	99
age	38	41	48	51	47
region	57	57	54	57	57

Table 3: Classification accuracy in (%) for different speaker characteristics.

Figure 7: Latent speaker coefficients in function of gender (*SAT* model).

The classification results are shown in table 3, with classification results based on the prior class distributions and based on the latent speaker coefficients for a SI model without VTLN added for reference. Figures 7, 8a and 8b depict the content of the latent speaker coefficient matrix $\mathbf{H}$ for NMF combined with SAT, with the speakers re-ordered according to some speaker characteristic. These plots show three of the more interesting results and also illustrate how the NMF-based classification system (equations (30) and (31)) operates.

From table 3, it can be concluded that without VTLN, *gender* is the only discernible speaker characteristic. Adding VTLN allows the NMF-decomposition to encode additional speaker characteristics such as *age* in

the latent vectors. However, *gender* remains by far the most prominent feature, indicating that neither VTLN nor fMLLR can completely compensate for the gender differences. The investigation whether the latent speaker vectors also reflect speaker *accent*, measured by means of the *region* where the speaker went to primary school, was inconclusive. The lack of any observed correlation could be caused by several factors: (i) *region* may not predict *accent* since a great amount of the speakers moved among different regions, (ii) other characteristics such as ethnicity may be more related to accent, and (iii) we work on read speech so most speakers adopt a relative standard English accent.

Similar conclusions can be drawn from a visual inspection of the latent vector coefficients. Figure 7 illustrates how the latent speaker vectors encode the gender information of the training speakers: female speakers are described by the latent vectors with index 2, 3, 5, 7 and 10, i.e. those vectors ($\mathbf{W}$ matrix) which have large coefficients ($\mathbf{H}$ matrix) for the female speakers; the remaining 5 latent vectors describe the male speakers. Figure 8 shows how VTLN and SAT improve the NMF decomposition so that additional speaker characteristics such as *age* start to be encoded. In order to visualize the effect age has on of the latent speaker vectors, the preponderant gender feature is removed by only showing the female speakers. The speakers are sorted according to age, with the left panel showing the relation between speaker index and age. Figure 8a shows that some systematic patterns of latent speaker vector usage can be observed when using VTLN and SAT: the latent speaker vectors 2, 3 and 5 which show significantly more activation in the left lower corner of the latent speaker coefficient figure, are mainly used to describe people younger than 25 years old; the latent speaker vectors mainly used to describe people older than 55 are 7 and 10 and these show more activation in the right top corner. Without VTLN and SAT (figure 8b) no such systematic patterns can be observed. Similarly, no systematic patterns could be discerned on visual inspection of the latent vector coefficients in function of the speaker’s *region*.

7. Conclusions and future research

This paper described a novel model space fast speaker adaptation scheme that adjusts the Gaussian mixture weights. The target speaker weights are expressed as a linear combination of latent speaker vectors. The latent speaker vectors encode systematic patterns of variation in Gaussian usage between speakers. The vectors are learned by means of NMF on statistics collected for all speakers that make up the training data.

Figure 8: Latent speaker coefficients in function of age for female speakers.

By relying on pre-learned correlations, the statistics for both the seen and unseen acoustic model parameters (phones) are updated, resulting in good generalization. Furthermore, expressing weights in function of a few latent vectors limits the number of free parameters. Constraining the model space avoids overfitting. Nevertheless, weight adaptation still allows complex non-linear redistributions of the probability mass. These properties combined make that the proposed Gaussian mixture weight adaptation algorithm requires very little adaptation data (improvement in accuracy with 98% confidence after 0.5 s of data, convergence after 3 s of data on our tests), which is important for speech recognition applications where the amount of enrollment data is limited.

The effectiveness of the weight adaptation is directly related to the quality of the latent speaker vectors. Deriving high quality latent vectors requires that sufficiently rich statistics are collected from the training data. We proposed two methods to achieve this. Speaker adaptive training results in more active Gaussians and hence a richer input. The Gaussian exponentiation scheme lowers the dynamic range of the Gaussian likelihoods, revealing more correlation between Gaussian activations. In both cases, by encoding more relevant active Gaussian relations, more speaker characteristics were encoded in the latent speaker vectors, which in turn lead to a more effective weight adaptation.

Using the NMF-based weight adaptation in combination with SAT or GE decreased the WER on the WSJ 92 and 93 tasks with 5% compared to the original speaker independent system, which already included VTLN. This improvement is comparable with what is achieved with fMLLR on 100+ s of adaptation data. We also noticed that the NMF based weight adaptation and fMLLR based mean and variance adaptation are complementary to each other. The combination of these two adaptation techniques yielded further improvements in recognition performance, and given that both NMF-based weight adaptation and fMLLR are ML techniques, the combination is straightforward. The NMF weight adaptation and eigenvoice mean adaptation techniques are also compatible with each other. The combination of these two techniques outperforms the eigenvoice speaker adaptation technique alone consistently for both large or small amounts of enrollment data. The combination of NMF weight adaptation and eigenvoice mean adaptation was also found to outperform adaptation designs with more degrees of freedom, such as NMF weight adaptation alone, eigenvoice mean adaptation alone and NMF combined with fMLLR.

Unsupervised adaptation works equally well as supervised adaptation, even without filtering out the words with a low confidence score. This, in

combination with the very fast adaptation opens up the possibility to include NMF-based speaker adaptation in a single pass low-latency recognition system.

Future research will focus on additional methods to improve the quality of the latent vectors. We also intend to apply hierarchical weight decomposition as to adjust the degrees of freedom in the NMF-adaptation to the amount of available adaptation data. Switching to a small number of base vectors (degrees of freedom) avoids the overfitting problem when little enrollment data are available. Increasing the number of base vectors with large amounts of enrollment data allows the system to get closer to the true speaker dependent model.

References

- Anastasakos, T., McDonough, J., Makhoul, J., 1997. Speaker adaptive training: a maximum likelihood approach to speaker normalization, in: Proc. International Conference on Acoustics, Speech and Signal Processing, pp. 1043–1046.
- Anastasakos, T., McDonough, J., Schwartz, R., Makhoul, J., 1996. A compact model for speaker-adaptive training, in: Proc. International Conference in Spoken Language Processing, pp. 1137–1140.
- Baum, L., 1972. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities* 3, 1–8.
- Chen, K.T., Liao, W.W., Wang, H.M., Lee, L.S., 2000. Fast speaker adaptation using eigenspace-based maximum likelihood linear regression, in: Proc. International Conference on Spoken Language Processing, pp. 742–745.
- Chen, S., Goodman, J., 1998. Orthogonal nonnegative matrix tri-factorizations for clustering. Technical Report. Center for Research in Computing Technology (Harvard University). Technical Report, TR-10-98.
- Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society, series B* 39, 1–38.

- Demuynck, K., 2001. Extracting, modelling and combining information in speech recognition. Ph.D. thesis. Katholieke Universiteit Leuven.
- Duchateau, J., Leroy, T., Demuynck, K., Van hamme, H., 2008. Fast speaker adaptation using non-negative matrix factorization, in: Proc. International Conference on Acoustics, Speech and Signal Processing, pp. 4269–4272.
- Duchateau, J., Wigham, M., Demuynck, K., Van hamme, H., 2006. A flexible recogniser architecture in a reading tutor for children, in: Proc. ISCA Tutorial and Research Workshop (ITRW) on Speech Recognition and Intrinsic Variation, pp. 59–64.
- Gales, M.J.F., 1998. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer Speech and Language* 12, 75–98.
- Gaussier, E., Goutte, C., 2005. Relation between PLSA and NMF and implications, in: Proc. ACM SIGIR conference on Research and development in information retrieval, pp. 601–602.
- Gauvain, J.L., Lee, C.H., 1994. Maximum a posteriori estimation for multivariate gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing* 2, 291–298.
- Hofmann, T., 1999a. Probabilistic latent semantic analysis, in: Proc. 15th conference on uncertainty in AI.
- Hofmann, T., 1999b. Probabilistic latent semantic indexing, in: Proc. ACM SIGIR Special Interest Group on Information Retrieval, pp. 50–57.
- Huang, X., Acero, A., Hon, H.W., 2001. Spoken language processing: a guide to theory, algorithm and system development. Prentice Hall PTR.
- Kuhn, R., Junqua, J.C., Nguyen, P., Niedzielski, N., 2000. Rapid speaker adaptation in eigenvoice space. *IEEE Transactions on Speech and Audio Processing* 8, 695–707.
- Lee, D.D., Seung, H.S., 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 788–791.
- Lee, D.D., Seung, H.S., 2001. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems* 13, 556–562.

- Leggetter, C.J., Woodland, P.C., 1994. Speaker adaptation using linear regression. Technical Report CUED/F-INFENG/TR.181. Cambridge University Engineering Department.
- Leggetter, C.J., Woodland, P.C., 1995. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language* 9, 171–185.
- Nguyen, P., 1998. Fast speaker adaptation. Industrial Thesis Report. Institut Eurécom.
- Smaragdis, P., Brown, J.C., 2003. Non-negative matrix factorization for polyphonic music transcription, in: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 177–180.
- Smaragdis, P., Raj, B., Shashanka, M., 2006. A probabilistic latent variable model for acoustic modeling, in: *Proc. Advances in Models for Acoustic Processing Workshop*.
- Tuerk, C., Robinson, T., 1993. A new frequency shift function for reducing inter-speaker variance, in: *Proc. Eurospeech*, pp. 351–354.
- Van hamme, H., 2008. Integration of asynchronous knowledge sources in a novel speech recognition framework, in: *Proc. ISCA Tutorial and Research Workshop (ITRW) on Speech Analysis and Processing for Knowledge Discovery*.
- Virtanen, T., 2007. Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria. *IEEE Transactions on Audio, Speech and Language Processing* 15, 291–298.
- Woodland, P.C., 2001. Speaker adaptation for continuous density HMMs: A review, in: *ISCA Tutorial and Research Workshop (ITRW) on Adaptation Methods for Speech Recognition*, pp. 11–19.
- Xu, W., Liu, X., Gong, Y., 2003. Document-clustering based on non-negative matrix factorization, in: *Proc. ACM SIGIR Special Interest Group on Information Retrieval*, pp. 267–273.
- Zavaliagkos, G., Schwartz, R., 1996. Maximum a posteriori adaptation for large scale HMM recognizers, in: *Proc. International Conference on Acoustics, Speech and Signal Processing*, pp. 725–728.

- Zhang, X., Demuynck, K., Van hamme, H., 2011. Rapid speaker adaptation with speaker adaptive training and non-negative matrix factorization, in: Proc. International Conference on Acoustics, Speech and Signal Processing, pp. 4456–4459.
- Zhang, X., Demuynck, K., Van hamme, H., 2012. Latent variable speaker adaptation of gaussian mixture weights and means, in: Proc. International Conference on Acoustics, Speech and Signal Processing, pp. 4349–4352.