Adaptation Transforms of Auto-Associative Neural Networks as Features for Speaker Verification

Samuel Thomas[1], *Sri Harish Mallidi*[1], *Sriram Ganapathy*[1] *and Hynek Hermansky*[1,2]

[1]Center for Language and Speech Processing,
Department of Electrical and Computer Engineering,
[2]Human Language Technology Center of Excellence,
The Johns Hopkins University, Baltimore, USA.
{samuel, mallidi, ganapathy, hynek}@jhu.edu

Abstract

We present a new approach of using Auto-Associative Neural Networks (AANNs) in the conventional GMM speaker verification framework with i-vector feature extraction and PLDA modeling. In this technique, an i-vector feature extractor is trained using adaptation parameters from a mixture of AANNs. In order to model parts of each speaker's acoustic space, a training objective function based on posterior probabilities of broad phonetic classes is used. The AANN based i-vectors are fused with GMM based i-vectors and a joint PLDA model is trained. The proposed approach provides promising results and significant gains when combined with baseline systems on the telephone conditions of NIST SRE 2010 and the recently concluded IARPA BEST 2011 speaker evaluations.

1. Introduction

State-of-the-art speaker verification systems use different kinds of features to capture information that is useful in discriminating between speakers. Conventional short-term features extract information from the spectrum of speech modeled in short analysis windows spanning few milliseconds [1, 2]. These features typically describe the differences in speech production between speakers. However many speaker specific cues useful in characterizing speakers appear also in the manner of speaking. One class of features that model these *higher order* differences between speakers are prosodic features. These features capture variations in syllable length, loudness, pitch and energy at different time resolutions in analysis windows spanning several hundreds of milliseconds. Examples of this approach include simple prosodic features which model the trajectory of pitch and energy [3] and more complex syllable-based, non-uniform extraction regions features (SNERFs) [4]. A second class of such higher order features are adaptation transform based features which attempt to capture speaker differences in terms of speaker-dependent and speaker-independent speech recognition models. In [5], features from an adaptation transform used in speech recognition - the Maximum Likelihood Linear Regression (MLLR) speaker adaptation transform, are used. Similar to this approach, transformation weights derived from adaptation parameters applied to Multi-layer Perceptrons (MLPs) in a connectionist speech recognizer have also been proposed [6]. Although short-term features continue to be dominantly used, most systems typically now employ a combination of both short-term and long-term features for better modeling of speakers [7, 8].

The research presented in this paper was partially funded by the IARPA BEST program under contract Z857701, the DARPA RATS program under D10PC20015 and the JHU Human Language Technology Center of Excellence. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the IARPA or DARPA or JHU HLTCOE.

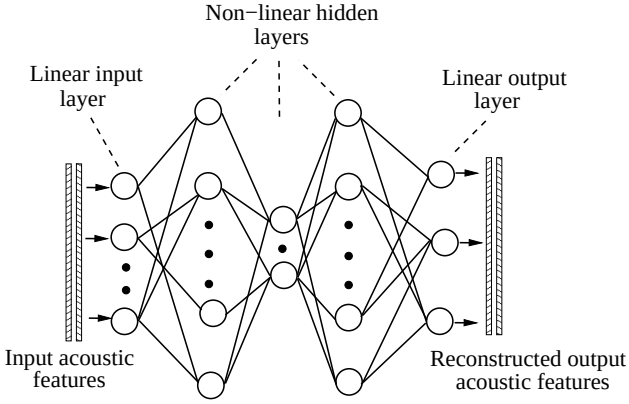

Figure 1: Auto-associative neural network with 5 hidden layers.

Apart from choosing good representative sets of features to model speakers, state-of-the-art systems also use various techniques to remove the effects of channel and session variability. In a simplified variant of the Joint Factor Analysis (JFA) technique [9] for Gaussian mixture model (GMM) based speaker verification [10], a single subspace covering the total variability of channels and speakers is first trained [11]. This model serves as an intermediate feature extractor to derive compact low-dimensional features called i-vectors. Probabilistic Linear Discriminant Analysis (PLDA) [12] is finally applied on the i-vectors to derive likelihood ratios for every trial [13].

Auto-Associative Neural Networks (AANNs) have been proposed as an alternative to Gaussian Mixture Models (GMMs) for modeling the distribution of data [14]. An AANN is a feed-forward neural network trained to reconstruct its input at its output through a hidden compression layer [15]. In our experiments we use AANNs with 5 layers as shown in Fig. 1. This architecture consists of three non-linear hidden layers between the linear input and output layers. The second hidden layer contains fewer nodes than the input layer, and is known as

the compression layer. AANNs have several advantages compared to the GMMs, for example, they relax the assumption of feature vectors to be locally normal. In [14, 16], AANN have been used as an alternative to GMMs for speaker verification. We extend this approach further by using a mixture of AANNs in [17].

In this paper we develop a new *higher order feature* based on Auto-Associative Neural Networks (AANNs). We use speaker specific adaptation weights from the mixture of AANNs as features, instead of using the AANNs directly as speaker models. This is done by first training an i-vector extractor on AANN adaptation parameters. I-vectors from the AANN system are concatenated with i-vectors from a conventional GMM based framework similar to the approach used in [18] with prosodic features. A joint PLDA model is finally trained to obtain likelihood ratios for trials. Our approach is different from other adaptation transform features since we now use adaptation parameters from models trained specifically for speaker verification rather than for speech recognition as in [5, 6].

The rest of the paper is organized as follows. In the next section we review how AANNs can be trained and used as models for speaker verification. Section 3 talks about how a lower dimensional i-vector front end can be trained from adaptation parameters from these AANNs. These i-vectors are then combined with i-vectors trained using conventional cepstral features to train a joint PLDA system discussed in Section 4. In Section 5 we describe how we use the proposed approach to build speaker verification systems. Experiments and results on different SRE tasks are presented in Section 6. The paper draws final conclusions in Section 7.

2. AANN Models for Speaker Verification

2.1. Modeling Speaker Data

As described earlier, AANNs are feed-forward neural networks with several layers trained to reconstruct the input at its output through a hidden compression layer. This is typically done by modifying the parameters of the network using the back-propagation algorithm such that the average squared error between the input and output is minimized over the entire training data. More formally, for an input vector $\mathbf{x}$, the network produces an output $\hat{\mathbf{x}}(\mathbf{x}, \mathcal{W})$ which depends both on the input $\mathbf{x}$ and the parameters $\mathcal{W}$ of the network (the set of weights and biases). For simplicity, we denote the network output as $\hat{\mathbf{x}}(\mathcal{W})$. The training process then adjusts the parameters such that -

$$\min_{\{\mathcal{W}\}} \mathbb{E}\left[\|\mathbf{x} - \hat{\mathbf{x}}(\mathcal{W})\|^2\right]. \tag{1}$$

This method of training ensures that for a well trained network, the average reconstruction error of input vectors that are drawn from the distribution of the training data will be small compared to vectors drawn from a different distribution [14]. The likelihood of the data $\mathbf{x}$ given the model can then be linked to the error as -

$$p(\mathbf{x}; \mathcal{W}) \propto \exp(-\mathbb{E}\left[\|\mathbf{x} - \hat{\mathbf{x}}(\mathcal{W})\|^2\right]). \tag{2}$$

In [14, 16], these properties have been used to model acoustic data for speaker verification. A single AANN is first trained as a universal background model (UBM) on acoustic features from large amounts of data containing multiple speakers. Since data from many speakers are used, the AANN model learns a speaker independent distribution of the acoustic vectors. For each speaker in the enrollment set, the UBM-AANN is then adapted to learn speaker dependent distributions by retraining the entire network using each speaker's enrollment data. During the test phase, the average reconstruction error of the test data is computed using both the UBM-AANN and the claimed speaker AANN model. In an ideal case, if the claim is true, the average reconstruction error under the speaker specific model will be smaller than under the UBM-AANN and vice versa if false.

This approach is similar to conventional UBM-GMM techniques [10] except for the maximum a posteriori probability (MAP) adaptation to obtain speaker specific models. In the MAP adaptation of GMMs, only those components that are well represented in the adaptation data get significantly modified. However in the case of neural networks, there is no similar mechanism by which only parts of the model can be adapted. This limits the ability of a single AANN to capture the distribution of acoustic vectors especially when the space of speakers is large. To address this issue, we introduce a mixture of AANNs as described in the following section.

2.2. Mixture of AANNs

A mixture of AANNs is composed of several independent AANNs each modeling a separate part of the acoustic feature space. In our experiments we partition the acoustic space into 5 classes corresponding to the broad phoneme classes of speech - vowels, fricatives, nasals, stops and silence. The assignment of a feature vector to one of these classes is done using posterior probabilities of these classes estimated using a separate multilayer perceptron (MLP). This additional information is incorporated into the objective function in Eqn. (1) as -

$$\sum_{j=1}^{c} \min_{\{\mathcal{W}_j\}} \mathbb{E}\left[P\left(\mathcal{C}_j/\mathbf{x}\right)\|\mathbf{x} - \hat{\mathbf{x}}(\mathcal{W}_j)\|^2\right] \tag{3}$$

where c denotes the number of mixture components or number of broad phoneme classes, and the set $\mathcal{W}_j$ consists of parameters of the j^{th} AANN of the mixture. $P\left(\mathcal{C}_j/\mathbf{x}\right)$ is the posterior probability of j^{th} broad phonetic class $\mathcal{C}_j$ given $\mathbf{x}$ estimated using the MLP. During back propagation training, since the error is weighted with class posterior probabilities, each mixture component is trained only on frames corresponding to a particular broad phonetic class.

Similar to the single AANN case, a UBM-AANN is first trained on large amounts of data. For each speaker in the enrollment, the UBM is then adapted using speaker specific enrollment data. Broad class phoneme posteriors are used in both these cases to guide the training of each class specific mixture component on appropriate set of frames. This approach helps to alleviate the limitation of a single AANN model described earlier since only parts of the UBM-AANN are now adapted based on the speaker data.

Using the mixture of AANNs, the average reconstruction error of data $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is given by

$$e\left(\mathcal{D}; \mathcal{W}_1, \ldots, \mathcal{W}_c\right) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{c} P\left(\mathcal{C}_j/\mathbf{x}_i\right)\|\mathbf{x}_i - \hat{\mathbf{x}}_i(\mathcal{W}_j)\|^2}{n}. \tag{4}$$

During the test phase, likelihood scores based on reconstruction errors from both the UBM-AANN and the claimed speaker models are used to make a decision as shown in Fig. 2. In our experiments, since the amount of adaptation data is usually limited, we adapt only the last layer weights of each AANN component. We also restrict the number of nodes of the third hidden layer to the size of the output layer.

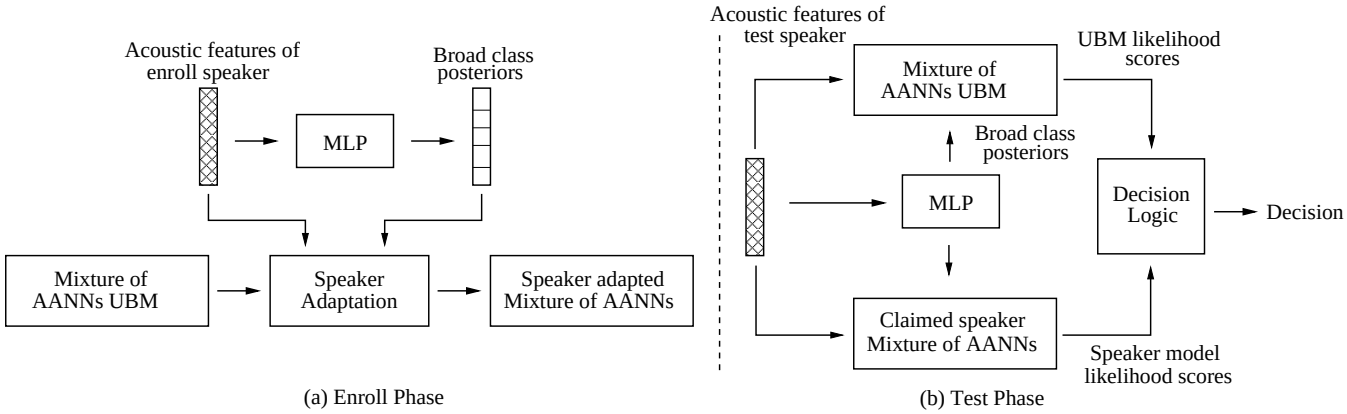

Figure 2: Enrollment and testing using a mixture of AANNs

3. Converting AANN Adaptation Transforms to Features

As discussed earlier, speaker specific mixture of AANN models are created by adapting the last layer weights of a speaker independently trained UBM-AANN. Since the adaptation is constrained to the set of last layer weights, we hypothesize these parameters as useful speaker specific adaptation transforms. By using a mixture of AANNs, speaker specific transforms are also trained separately across different phoneme classes. In this section, we discuss how these individual transforms can be modeled in a lower dimensional sub-space and then used as features.

In the Joint Factor Analysis (JFA) [9] for Gaussian mixture model (GMM) based speaker verification, a supervector of GMM means $\mathbf{s}$ is modeled in terms of two low-dimensional latent variables $\mathbf{x}$ and $\mathbf{y}$ corresponding to separate speaker and channel subspaces as -

$$\mathbf{s} = \mathbf{m} + \mathbf{V}\mathbf{y} + \mathbf{U}\mathbf{x}, \tag{5}$$

where $\mathbf{m}$ is a speaker and channel independent supervector, and $\mathbf{V}$ and $\mathbf{U}$ are subspaces capturing speaker and channel variability in the GMM supervector space. In a recent approach, a single subspace covering the total variability across supervectors of GMM means has been used [11]. Under this model a supervector of GMM means $\mathbf{s}$ is described in terms of a speaker and channel independent supervector $\mathbf{m}$ and a single subspace $\mathbf{T}$ as -

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{x}. \tag{6}$$

The low-dimensional latent variable $\mathbf{x}$ are also known as i-vectors.

Similar to these subspace modeling techniques, we model the adaptation parameters of mixture of AANNs in a lower dimensional subspace that captures both speaker and channel variabilities. The subspace learning problem is formulated as a regularized weighted least squares problem. We compare our update equations with that of total variability space training of GMMs [11]. The current work draws motivation from [19] in which the joint factor analysis (JFA) of GMMs is reinterpreted as signal coding using overcomplete dictionaries.

The subspace modeling of adaptation parameters with a development set of m speakers is formulated as follows. The mixture of AANN-UBM with c components is first adapted separately for each $h(s)$ session of a speaker s. We denote the vectorized last layer weights of j^{th} AANN component of this mixture model as $\boldsymbol{w}^j_{s,h}$. The following quantities are then computed -

(a) the number of points (soft) aligned with each mixture component, denoted as $n^j_{s,h}$. This count is obtained by summing the corresponding j^{th} MLP posteriors of the s^{th} speaker and h^{th} session along time (frames).

(b) the weight supervector obtained by concatenating the vectorized weights $\boldsymbol{w}^j_{s,h}$ as shown below-

$$\boldsymbol{w}_{s,h} = \begin{bmatrix} \boldsymbol{w}^1_{s,h} \\ . \\ . \\ \boldsymbol{w}^c_{s,h} \end{bmatrix}.$$

(c) the mean and covariance matrix of the weight supervector. These are computed using the soft counts as:

$$\boldsymbol{w} = \left(\sum_{s=1}^{m} \sum_{h=1}^{h(s)} \boldsymbol{N}_{s,h} \right)^{-1} \sum_{s=1}^{m} \sum_{h=1}^{h(s)} \boldsymbol{N}_{s,h}\ \boldsymbol{w}_{s,h}$$

$$\boldsymbol{\Sigma} = \left(\sum_{s=1}^{m} \sum_{h=1}^{h(s)} \boldsymbol{N}_{s,h} \right)^{-1} \left(\left[\sum_{s=1}^{m} \sum_{h=1}^{h(s)} \boldsymbol{N}_{s,h}\ (\boldsymbol{w}_{s,h} - \boldsymbol{w})\,(\boldsymbol{w}_{s,h} - \boldsymbol{w})^T \right] \right)$$

where,

$$\boldsymbol{N}_{s,h} = \begin{bmatrix} n^1_{s,h}\boldsymbol{I}_d & & \mathbf{0} \\ & n^2_{s,h}\boldsymbol{I}_d & \\ & & \\ \mathbf{0} & & n^c_{s,h}\boldsymbol{I}_d \end{bmatrix},$$

and $\boldsymbol{I}_d$ is a $d \times d$ identity matrix with d being the cardinality of $\boldsymbol{w}^1_{s,h}$.

We model the supervector $\boldsymbol{w}_{s,h}$ using a lower dimensional affine subspace parameterized by $\boldsymbol{T}$ i.e., $\boldsymbol{w}_{s,h} = \boldsymbol{w} + \boldsymbol{T}\boldsymbol{q}_{s,h}$, where $\boldsymbol{q}_{s,h}$ represents the unknown i-vector associated with the h^{th} session of s^{th} speaker. To find $\boldsymbol{T}$, the following weighted

least squares cost function is minimized with respect to its arguments:

$$\begin{aligned} & L\left(\boldsymbol{T}, \boldsymbol{q}_{1,1}, \ldots, \boldsymbol{q}_{m,h(m)}\right) \\ = & \sum_{s=1}^{m}\sum_{h=1}^{h(s)} \| \left[\boldsymbol{w}_{s,h} - \left(\boldsymbol{w} + \boldsymbol{T}\boldsymbol{q}_{s,h}\right)\right] \|^2_{\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}} \\ + & \underbrace{\sum_{s=1}^{m}\sum_{h=1}^{h(s)} \left(\lambda \operatorname{tr}\left(\boldsymbol{T}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\boldsymbol{T}\right) + \boldsymbol{q}_{s,h}^T\boldsymbol{q}_{s,h}\right)}_{regularization\ term} \end{aligned} \quad (7)$$

where $\|.\|_{\boldsymbol{A}}$ denotes a norm given by $\|\boldsymbol{x}\|^2_{\boldsymbol{A}} = \boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}$, and λ is a small positive constant.

Differentiating (7) with respect to $\boldsymbol{q}_{s,h}$ and setting equal to zero yields,

$$\begin{aligned} \frac{\partial L}{\partial \boldsymbol{q}_{s,h}} = \boldsymbol{0} \Rightarrow \\ -\boldsymbol{T}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\left[\boldsymbol{w}_{s,h} - \left(\boldsymbol{w} + \boldsymbol{T}\boldsymbol{q}_{s,h}\right)\right] + \boldsymbol{q}_{s,h} = \boldsymbol{0} \Rightarrow \\ \boldsymbol{q}_{s,h} = \left(\boldsymbol{I} + \boldsymbol{T}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\boldsymbol{T}\right)^{-1}\boldsymbol{T}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\left(\boldsymbol{w}_{s,h} - \boldsymbol{w}\right) \end{aligned} \quad (8)$$

Differentiating (7) with respect to $\boldsymbol{T}$ and setting it equal to zero yields,

$$\begin{aligned} \frac{\partial L}{\partial \boldsymbol{T}} = \boldsymbol{0} \Rightarrow \sum_{s=1}^{m}\sum_{h=1}^{h(s)} \{\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\boldsymbol{T}\boldsymbol{q}_{s,h}\boldsymbol{q}_{s,h}^T - \\ \boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\left[\boldsymbol{w}_{s,h} - \boldsymbol{w}\right]\boldsymbol{q}_{s,h}^T + \lambda\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\boldsymbol{T}\} = \boldsymbol{0} \\ \Rightarrow \sum_{s=1}^{m}\sum_{h=1}^{h(s)} \boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\boldsymbol{T}\left(\lambda\boldsymbol{I} + \boldsymbol{q}_{s,h}\boldsymbol{q}_{s,h}^T\right) = \\ \sum_{s=1}^{m}\sum_{h=1}^{h(s)} \boldsymbol{\Sigma}^{-1}\boldsymbol{N}_{s,h}\left[\boldsymbol{w}_{s,h} - \boldsymbol{w}\right]\boldsymbol{q}_{s,h}^T \end{aligned} \quad (9)$$

To obtain $\boldsymbol{T}$, we iterate between (8) and (9). In other words, for a given $\boldsymbol{T}$, we first find the i-vectors $\{\boldsymbol{q}_{1,1}, \ldots, \boldsymbol{q}_{m,h(m)}\}$ using (8). In the next step, we solve for $\boldsymbol{T}$ in (9) using the i-vectors $\{\boldsymbol{q}_{1,1}, \ldots, \boldsymbol{q}_{m,h(m)}\}$ found in the previous step. This procedure is repeated for several times until convergence. The above update equations can be compared with the total variability space training of GMMs [11]. Note that (8) and (9) resemble the maximum likelihood (ML) update equations in [11], except for the $\lambda\boldsymbol{I}$ term in (9).

Given a new speaker utterance, lower dimensional i-vectors are now extracted in two steps -
(a) Derive speaker specific adaptation transforms by adapting the mixture of AANN-UBM. This is done by retraining only the last layer weights of the UBM with acoustic features and broad class phoneme posteriors as input. The adaptation transforms are then vectorized to form a supervector $\hat{\boldsymbol{w}}$ by concatenating the weights from each mixture component.
(b) Apply the trained i-vector extractor with the subspace model parameters $\boldsymbol{w}$ and $\boldsymbol{T}$ to generate i-vectors $\boldsymbol{q}$ using Eqn. 8.

4. Joint Modeling of Cepstral and AANN based Features

In the i-vector training procedure described above, a single subspace is used to model both the speaker and channel variability while performing dimensionality reduction. To counter the effects of channel variability which have not yet been removed, a second level of modeling is hence employed. For a particular speaker, i-vectors $\boldsymbol{x_s} : \{s = 1, \ldots, S\}$ corresponding to multiple utterances from the speaker are assumed to be generated from a probabilistic model [12] such that $\boldsymbol{x_s}$ can be decomposed as

$$\boldsymbol{x_s} = \underbrace{\boldsymbol{m} + \boldsymbol{F}\boldsymbol{y}}_{A} + \underbrace{\boldsymbol{G}\boldsymbol{z_s} + \boldsymbol{\epsilon_s}}_{B}. \quad (10)$$

Part A corresponds to a speaker-specific part expressed in terms of a global offset $\boldsymbol{m}$, a speaker-specific subspace $\boldsymbol{F}$ and a latent identity variable $\boldsymbol{y}$ with a normal distribution. Part B on the other hand is utterance dependent and describes a channel subspace in terms of $\boldsymbol{G}$ and a normally distributed latent variable $\boldsymbol{z_s}$ along with a residual term $\boldsymbol{\epsilon_s}$ assumed to be Gaussian with zero mean and diagonal covariance $\boldsymbol{\Sigma}$. All the latent variables are assumed to be statistically independent.

Since the i-vectors that we derive from the speaker specific AANN transforms have a very low dimension (80 dimensions) we include the following steps in our modeling process-
(a) Fuse the AANN based i-vectors with conventional i-vectors from a GMM based system trained using cepstral features, similar to the approach in [18],
(b) Incorporate the modifications proposed in [13, 20] and use a simpler PLDA model expressed as

$$\boldsymbol{x_s} = \boldsymbol{m} + \boldsymbol{F}\boldsymbol{y} + \boldsymbol{\epsilon_s}. \quad (11)$$

The maximum likelihood estimates of the model parameters $\{\boldsymbol{m}, \boldsymbol{F}, \boldsymbol{\Sigma}\}$ are obtained from a large collection of development data using an EM algorithm formulated in [12].

Once the joint model has been trained, the model can be used to evaluate the log-likelihood ratio for the hypothesis test of whether a pair of i-vectors belongs to the same speaker or different speakers. This can be efficiently computed in closed-form as described in [13].

5. Speaker Verification Systems

Fig. 3 illustrates the different building blocks that are used in the proposed system. Each of the blocks are explained in detail below.

5.1. AANN Subsystem

A. *Acoustic features* - The acoustic features used in our experiments are 39 dimensional FDLP features [21]. In this technique, sub-band temporal envelopes of speech are first estimated in narrow sub-bands (96 linear bands). These sub-band envelopes are then gain normalized to remove reverberation and channel artifacts. After normalization, the frequency axis is warped to the mel scale. We use 37 Mel bands in the frequency range of 125-3800 Hz to derive a gain normalized mel scale energy representation of speech similar to the mel spectrogram obtained in conventional MFCC feature extraction. These mel band energies are converted to cepstral coefficients by using a log operation followed by DCT. We use 13 cepstral coefficients along with derivative and acceleration components yielding 39 dimensional features. These features are VAD processed and feature warped [22].

B. *MLP based posteriors* - Posterior features are used at multiple stages of the AANN based subsystem. These

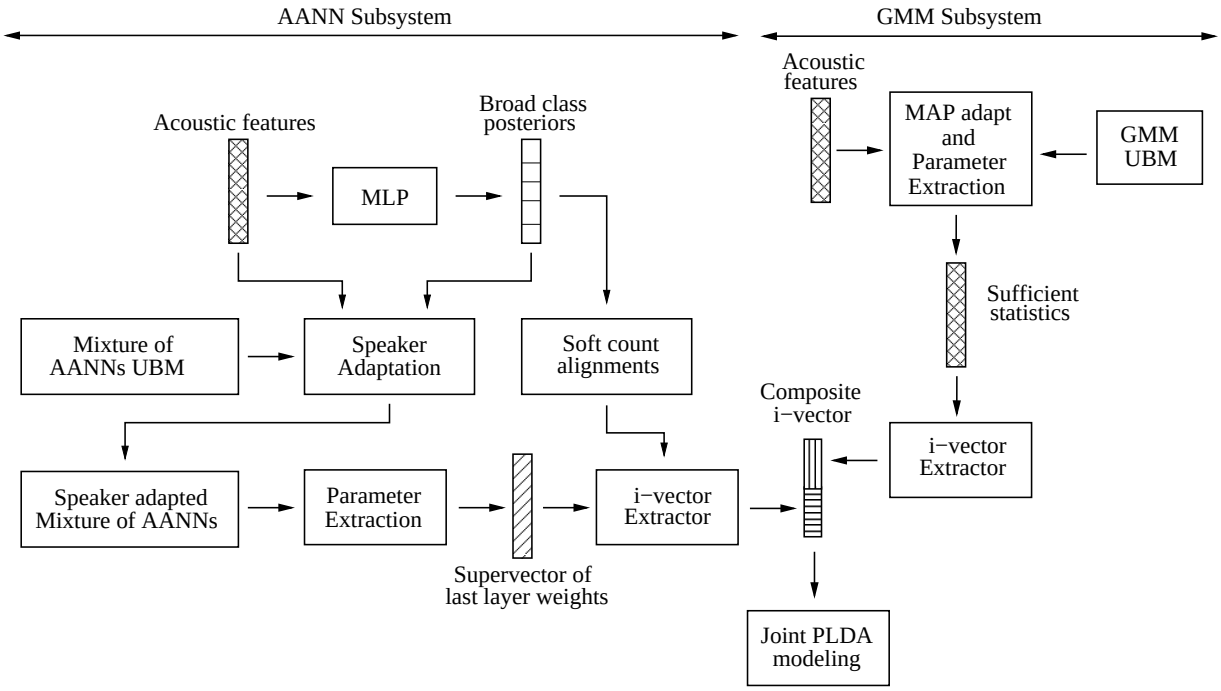

Figure 3: Creating AANN and GMM based i-vectors for joint PLDA modeling

posteriors are derived from an MLP trained on 300 hours of conversational telephone speech (CTS) [23]. The 45 phoneme posteriors are combined appropriately to obtain 5 broad phonetic class posteriors corresponding to vowels, fricatives, plosives, nasals and silence. In this paper we focus only on telephone channel conditions for which this MLP can be used to derive reliable posteriors.

C. *Mixture of AANNs UBM* - We train gender specific AANN UBMs using a telephone development data set consisting of audio from the NIST 2004 speaker recognition database, the Switchboard II Phase III corpora and the NIST 2006 speaker recognition database. We use 400 male and 400 female utterances to train the UBMs. Broad-class posteriors for these utterances are derived using the MLP described above. Each AANN component of the UBM has a linear input and a linear output layer along with three nonlinear (tanh nonlinearity) hidden layers. Both input and output layers have 39 nodes corresponding to the dimensionality of the input FDLP features. We use 160 nodes in the first hidden layer, 20 nodes in the compression layer and 39 nodes in the third hidden layer. The training procedure described in Sec 2.2 is used to train the gender specific mixture of AANNs.

D. *Speaker adaptation* - Speaker specific models are obtained by adapting (retraining) only the last layer weights (39x39 parameters) of each AANN.

E. *I-vector extractor* - Gender specific 80 dimensional total variability subspaces are trained as described in Sec. 3. A development data set of 8750 male and 10500 female utterances drawn from Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2 and NIST 2004-2005 SRE are used to train these subspaces. Soft counts of points that align with each mixture component are derived from MLP posteriors. The weight supervectors are extracted from speaker adapted models corresponding to each of these utterances.

5.2. GMM Subsystem

A. *Acoustic features* - We use the same 39 dimensional FDLP features that we use with the neural networks to train gender specific GMM models.

B. *GMM UBM* - 1024 component gender specific GMMs are trained on a development set from NIST SRE 2004, Switchboard II Phase III, NIST SRE 2006 and the NIST SRE 2008 interview development set. This data collection contains close to 4300 male and 5500 female utterances. We use lesser amounts of data for the neural networks to avoid any over training since they have much fewer number of free parameters.

C. *I-vector extractor* - Once the UBM is trained, the mixture component means are MAP adapted and concatenated to form supervectors. We use the i-vector based factor analysis technique [11] on these supervectors in a gender dependent manner. For the factor analysis subspaces, we use a development set made of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2, NIST SRE 2004, NIST SRE 2005 and extended NIST SRE 2008 far-field data. Close to 17000 male recordings and 21300 female recordings are used to extract 450 dimensional gender specific i-vectors.

5.3. Joint PLDA modeling

For the joint PLDA modeling, i-vectors from the AANN subsystem and the GMM subsystem are fused together to form a 530 dimensional i-vector. After length normalization [13], a 250 dimensional PLDA sub-space is trained on the same telephone training set used for training AANN subspace i-vector extractor. The joint PLDA system is used to generate scores for

Table 1: *EER (%)and DCF values for NIST SRE 2010 extended core data set, on telephone conditions 5, 6 and 8.*

Condition	Number of Trials		GMM Baseline System		AANN based System		Score fusion 0.7-0.3	
	TGT	NTGT	EER	DCF	EER	DCF	EER	DCF
C5 (Tel. speech, Normal Vocal Effort in train and test)	7,169	408,950	2.97	0.4895	4.13	0.5875	2.83	0.4471
C6 (Tel. speech, Normal Vocal Effort in train, High Vocal Effort in test)	4,137	461,438	4.41	0.7566	5.95	0.7872	4.19	0.7120
C8 (Tel. speech, Normal Vocal Effort in train, Low Vocal Effort in test)	3,821	404,848	1.89	0.4633	2.88	0.5030	1.65	0.4285

Table 2: *BEST Evaluation metric values for the IARPA BEST 2011 evaluation, on conditions involving telephone data.*

Condition	Number of Trials	GMM Baseline System (P_{FA} at P_{Miss}=10%)	AANN based System (P_{FA} at P_{Miss}=10%)	Score Fusion 0.7-0.3 (P_{FA} at P_{Miss}=10%)
Conversational tel. speech in training and test (Tel-phonecall-Tel-phonecall)	7,128,892	1.337	3.045	1.332
Training with conver. tel. test speech recorded over a room mic. channel and conver. tel. speech in test (Mic-phonecall-Tel-phonecall)	3,433,346	7.531	8.091	6.31
Interview training speech and conv. tel. test speech (Mic-interview-Tel-phonecall)	3,918,542	3.259	4.189	2.538

the proposed system.

5.4. Baseline GMM based system

To evaluate system performances, we build a separate gender specific GMM based baseline system using MFCC features. The architecture of this system is identical to the FDLP based GMM subsystem described above with exception to a separate PLDA system that generates the final scores. This 250 dimensional PLDA sub-space uses both telephone and microphone data drawn from Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2, NIST SRE 2004, NIST SRE 2005 and the extended NIST SRE 2008 far-field data.

6. Speaker Verification Experiments and Results

We evaluate the proposed systems on two speaker verification evaluations. Our first evaluation is conducted on telephone conditions 5, 6 and 8 of the core extended-trial NIST SRE 2010 evaluation [24]. Equal error rate (EER) and 2010 detection cost function (DCF) values for the baseline and proposed system are shown in Table 1. The proposed system with AANN features is slightly worse in performance to the baseline GMM system. However when scores from both the systems are combined, an improvement of close to 7% in both EER and DCF scores over all conditions is observed.

The same systems used for the SRE 2010 evaluations were deployed for the recently concluded IARPA BEST 2011 evaluation [25]. Table 2 shows the results on conditions involving telephone data using the BEST evaluation metric. This metric is defined as the value of P_{FA} at the decision threshold (operating point) for which P_{Miss} = 10%. We observe very similar trends on this very large evaluation set. A significant relative improvement of 17% is obtained over all conditions with score fusion. We use the same choice of fusion weights (0.7-0.3) for both evaluations.

In their current form the AANN based systems do not outperform the conventional GMM systems although they provide very useful complimentary information. This could probably be due to the differences in the amount of data used to train the systems. The GMM baseline has been trained on close to twice the amount of data with both telephone and microphone utterances. However only telephone data has been used with the AANN system. We are encouraged by these results and hope to extend the proposed systems further. We are currently training MLPs with microphone and interview data to allow the proposed approach to used with these kinds of data as well.

7. Conclusions

In this paper we have proposed adaptation transforms of auto-associative neural networks as new features for speaker verification. Unlike earlier approaches which use speaker transforms from speech recognition systems, we use speaker transforms from models that are specifically trained for speaker verification. The paper describes how the speaker specific models are trained. Supervectors of last layer weights derived from these models are then used along with a subspace modeling technique. We show how these transforms can be integrated with conventional GMM based speaker verification systems. Experiments on two evaluation tasks with several millions of trials show the usefulness of the proposed technique.

8. Acknowledgments

The authors would like to thank G.S.V.S. Sivaram, X. Zhou, D. Zotkin, D. Garcia-Romero, O. Glembek and members of the speech group at BUT for their help in setting up the systems.

9. References

[1] H. Hermansky, "Perceptual Linear Predictive (PLP) Analysis of Speech", Journal of the Acoustic Society of America, 1990.

[2] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences", IEEE Transactions on Acoustics, Speech, and Signal Processing, 1980.

[3] C.-Y. Lin and H.-C. Wang, "Language identification using pitch contour information", IEEE ICASSP, 2005.

[4] E. Shriberg, L. Ferrer, S. Kajarekar, A. Venkataraman, and A. Stolcke, "Modeling prosodic feature sequences for speaker recognition", Speech Communication, 2005.

[5] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg and A. Venkataraman, "MLLR transforms as features in speaker recognition", ISCA Eurospeech, 2005.

[6] A. Abad and J. Luque, "Connectionist Transformation Network Features for Speaker Recognition", ISCA Odyssey, 2010.

[7] L. Burget, M. Fapšo, V. Hubeika, O. Glembek, M. Karafiát, M. Kockmann, P. Matějka, P. Schwarz and J. Černockỳ, "BUT system for NIST 2008 speaker recognition evaluation", ISCA Interspeech, 2009.

[8] S. Kajarekar, N. Scheffer, M. Graciarena, E. Shriberg, A. Stolcke, L. Ferrer and T. Bocklet, "The SRI NIST 2008 speaker recognition evaluation system", IEEE ICASSP, 2009.

[9] P. Kenny, G. Boulianne, P. Oullet and P. Dumouchel, "Joint factor analysis versus eigenchannes in speaker recognition", IEEE Transactions on Audio, Speech, and Language Processing, 2007.

[10] D. Reynolds, T. Quatieri and R. Dunn, "Speaker verification using adapted Gaussian mixture models", Digital Signal Processing, 2000.

[11] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel and P. Ouellet, "Front-end factor analysis for speaker verification", IEEE Transactions on Audio, Speech, and Language Processing, 2010.

[12] S.J.D. Prince and J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity", IEEE ICCV, 2007.

[13] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of I-vector Length Normalization in Speaker Recognition Systems", ISCA Interspeech, 2011.

[14] B. Yegnanarayana and S. Kishore, "AANN: an alternative to GMM for pattern recognition", Neural Networks, 2002.

[15] M.A. Kramer, "Nonlinear principal component analysis using auto-associative neural networks", AICHE Journal, 1991.

[16] K.S.R. Murty and B. Yegnanarayana, "Combining evidence from residual phase and MFCC features for speaker recognition", IEEE Signal Processing Letters, 2005.

[17] G.S.V.S. Sivaram, S. Thomas and H. Hermansky, "Mixture of Auto-Associative Neural Networks for Speaker Verification", ISCA Interspeech, 2011.

[18] M. Kockmann, L. Ferrer, L. Burget and J. Černockỳ, "iVector Fusion of Prosodic and Cepstral Features for Speaker Verification", ISCA Interspeech, 2011.

[19] D. Garcia-Romero and C.Y. Espy-Wilson, "Joint Factor Analysis for Speaker Recognition Reinterpreted as Signal Coding Using Overcomplete Dictionaries", ISCA Odyssey, 2010.

[20] P. Kenny, "Bayesian Speaker Verification with Heavy-Tailed Priors", ISCA Odyssey, 2010.

[21] S. Ganapathy, J. Pelecanos and M.K. Omar, "Feature Normalization for Speaker Verification in Room Reverberation", IEEE ICASSP 2011.

[22] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification", ISCA Odyssey, 2001.

[23] S. Ganapathy, S. Thomas and H. Hermansky, "Static and Dynamic Modulation Spectrum for Speech Recognition", ISCA Interspeech, 2009.

[24] NIST SRE 2010, http://www.nist.gov/itl/iad/mig/sre10.cfm

[25] IARPA BEST 2011, http://www.nist.gov/itl/iad/mig/best.cfm