Model-Based Approaches for Degraded Channel Modelling in Robust ASR

M.J.F. Gales and F. Flego

Cambridge University Engineering Department, Trumpington Street, Cambridge, UK
mjfg@eng.cam.ac.uk, ff257@eng.cam.ac.uk

Abstract

Speech is usually observed after passing through some form of "channel" that results in distortions. For some scenarios it is possible to build explicit models of this channel distortion and hence compensate the acoustic models. However the accuracy of the distortion model is sometimes poor and more general adaptation approaches are required. This paper investigates these model-based approaches for communication channel, link, modelling. In particular the paper examines the interaction of link models with speaker adaptation and adaptive training. CMLLR link models with multiple transforms can yield multiple inconsistent feature-spaces When combined with speaker adaptation with very few transforms this inconsistency can limit adaptation performance gains. In contrast using a front-end CMLLR (FE-CMLLR) transform yields a consistent space for speaker adaptation. These schemes are compared on communication channel distorted dialect Arabic conversational speech. Preliminary results on this task indicate the benefits of performing adaptation in a consistent feature-space.

Index Terms: acoustic model adaptation, adaptive training.

1. Introduction

For most practical applications of speech recognition it is not possible to observe the "clean" speech signal. There are for example distortions caused by background noise and channel effects, as well as differences due to speaker. There are a number of approaches that can address this problem. This paper concentrates on model-based approaches which can be split into: *predictive* schemes, such as vector Taylor series (VTS) compensation [1]; and *adaptive* schemes such as Maximum Likelihood Linear Regression (MLLR) [2] and Constrained MLLR (CMLLR) [3]. When multiple distinct distortions, factors, affect the signal it is important to consider how the various forms of compensation interact with each other. This is of particular interest when adaptive training approaches [4, 3] are being used.

This paper considers one particular scenario with multiple factors: speech from multiple speakers is transmitted down a range of communication channels. Thus it is necessary to consider both the communication channel and the speaker. An interesting aspect of this scenario is that there is usually a large amount of data available for the channel model compared to limited data for the speaker models. Thus when using linear transforms, multiple linear transforms (regression classes) are used for the channel, compared to a limited number for the speaker. For the standard adaptive-training transform, CMLLR, this can cause issues when estimating and using the speaker transforms [5]. In particular CMLLR independently transforms the features for a particular regression class. This could yield dramatically different transformed feature spaces across the regression classes. Using one or two speaker transforms estimated on these inconsistent features can yield limited performance gains. It is possible to modify the speaker transform estimation scheme to compensate for this. However in this paper an alternative scheme is investigated front-end CMLLR (FE-CMLLR) [6]. This scheme interpolates multiple transforms together and yields a consistent space for speaker adaptation.

This work was partially supported by DARPA under the RATS program. The paper does not necessarily reflect the position or the policy of US Government and no official endorsement should be inferred.

2. Model-Based Adaptation

This section briefly discusses the attributes and examples of the two main forms of model-based adaptation, predictive and adaptive schemes.

In predictive approaches an explicit model of the channel distortion is used. A common approach is VTS compensation where the distortion is assumed to be of the form [1] for the distorted d-dimensional speech observation $\boldsymbol{y}_t$

$$\begin{aligned}\boldsymbol{y}_t &= \mathbf{C}\log\left(\exp(\mathbf{C}^{-1}(\boldsymbol{x}_t+\boldsymbol{h}_t))+\exp(\mathbf{C}^{-1}(\boldsymbol{n}_t))\right)\\ &= \boldsymbol{f}(\boldsymbol{x}_t,\boldsymbol{h}_t,\boldsymbol{n}_t)\end{aligned} \quad (1)$$

where $\boldsymbol{x}_t$ are the clean speech features, $\mathbf{C}$ is the DCT matrix, $\boldsymbol{h}_t$ the convolutional distortion, and $\boldsymbol{n}_t$ is the additive noise vector. Applying a first-order VTS approximation to (1) for a particular component m yields the following compensated mean $\boldsymbol{\mu}_{\mathrm{y}}^{(m)}$ for clean mean $\boldsymbol{\mu}_{\mathrm{x}}^{(m)}$

$$\boldsymbol{\mu}_{\mathrm{y}}^{(m)} = \boldsymbol{f}(\boldsymbol{\mu}_{\mathrm{x}}^{(m)},\boldsymbol{\mu}_{\mathrm{h}},\boldsymbol{\mu}_{\mathrm{n}}) \quad (2)$$

where $\boldsymbol{\mu}_{\mathrm{n}}$ and $\boldsymbol{\mu}_{\mathrm{h}}$ are the additive noise and convolutional distortion means. One of the strengths of VTS and related approaches is that the number of parameters to be estimated from the adaptation data is very small, whilst at the same time yielding a highly non-linear compensation process. However the approach relies on the mismatch function (1) being a reasonable estimate of the distortions being applied to the channel.

Initial experiments using discriminative VTS adaptive training [7] were carried out using the data described in section 5. Though gains were observed over an equivalent system using just CMN, the standard PLP-based system out-performed the VTS adaptive training scheme when using HLDA. In terms of the distortions caused by the communication links, the mismatch function in (1) does not appear to be an appropriate form to use, and is not investigated further.

Rather than using an explicit representation for the distortion of the channel, it is possible to use general adaptation approaches to model the differences. Currently linear transformation schemes are popular for adaptation, for example MLLR and CMLLR. To model complex distortions multiple linear transforms can be used. The usual approach is to construct a regression class tree and associate a linear transform with each of

the leaves of the tree [2]. Each component, m, of the system is assigned to one of the R regression classes, r_m indicates the regression class. For CMLLR-based models ($\boldsymbol{\Sigma}_{\mathtt{x}}^{(m)}$ is diagonal)

$$p(\boldsymbol{y}_t|m,\mathcal{M}_{\mathtt{x}},\mathcal{M}_{\mathtt{lnk}}) = |\mathbf{A}_{\mathtt{lnk}}^{(r_m)}|\mathcal{N}\left(\mathbf{A}_{\mathtt{lnk}}^{(r_m)}\boldsymbol{y}_t + \mathbf{b}_{\mathtt{lnk}}^{(r_m)};\boldsymbol{\mu}_{\mathtt{x}}^{(m)},\boldsymbol{\Sigma}_{\mathtt{x}}^{(m)}\right) \quad (3)$$

$\mathcal{M}_{\mathtt{lnk}}$ is the communication channel, or link, model. The parameters of the R link transforms, $\mathbf{W}_{\mathtt{lnk}}^{(r)} = [\mathbf{A}_{\mathtt{lnk}}^{(r)}\ \mathbf{b}_{\mathtt{lnk}}^{(r)}]$, can then be found using the following auxiliary function

$$\mathcal{Q}(\mathcal{M}_{\mathtt{lnk}},\hat{\mathcal{M}}_{\mathtt{lnk}}) = \sum_{r=1}^{R}\left[\sum_{m:r_m=r}\sum_{t=1}^{T}\gamma_t^{(m)}\log(|\mathbf{A}_{\mathtt{lnk}}^{(r)}|) - \frac{1}{2}\sum_{j=1}^{d}\left(\mathbf{w}_{\mathtt{lnk}j}^{(r)}\mathbf{G}^{(rj)}\mathbf{w}_{\mathtt{lnk}j}^{(r)\mathsf{T}} - 2\mathbf{w}_{\mathtt{lnk}j}^{(r)}\mathbf{k}^{(rj)\mathsf{T}}\right)\right] \quad (4)$$

where $\mathbf{w}_{\mathtt{lnk}j}^{(r)}$ is the j^{th} *row* of $\mathbf{W}_{\mathtt{lnk}}^{(r)}$, $\gamma_t^{(m)}$ is the posterior probability of component m generating the observation at time t using the current link model $\hat{\mathcal{M}}_{\mathtt{lnk}}$, $\mathcal{M}_{\mathtt{lnk}}$ is to be estimated and

$$\mathbf{G}^{(rj)} = \sum_{m:r_m=r}\sum_{t=1}^{T}\frac{\gamma_t^{(m)}}{\sigma_{\mathtt{x}j}^{(m)2}}\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}} \quad (5)$$

$$\mathbf{k}^{(rj)} = \sum_{m:r_m=r}\sum_{t=1}^{T}\frac{\gamma_t^{(m)}\mu_{\mathtt{x}j}^{(m)}}{\sigma_{\mathtt{x}j}^{(m)2}}\boldsymbol{\zeta}_t^{\mathsf{T}} \quad (6)$$

$\boldsymbol{\zeta}_t^{\mathsf{T}} = [\boldsymbol{y}_t^{\mathsf{T}}\ 1]$. Given these statistics and the occupancy counts, an iterative estimate of the transforms can be obtained [3].

One of the advantages of CMLLR adaptation is that it is very efficient to use in adaptive training [3] as the transforms act on the features. Adaptive training can be applied at the channel (link) level, link adaptive training (LAT), or speaker level SAT.

3. Front-End CMLLR

An alternative form of feature transformation is the Front-End CMLLR (FE-CMLLR) [6]. Here the use of transforms is determined by component posteriors from a GMM, rather than regression classes. For an FE-CMLLR transform, $\mathcal{M}_{\mathtt{c}}$,

$$\hat{\boldsymbol{x}}_t = \sum_{c=1}^{C}\gamma_{\mathtt{c}t}^{(c)}\left(\mathbf{A}_{\mathtt{c}}^{(c)}\boldsymbol{y}_t + \mathbf{b}_{\mathtt{c}}^{(c)}\right) = \mathbf{A}_{\mathtt{c}t}\boldsymbol{y}_t + \mathbf{b}_{\mathtt{c}t} \quad (7)$$

where $\gamma_{\mathtt{c}t}^{(c)} = P(c|\boldsymbol{y}_t,\mathcal{M}_{\mathtt{c}})$ is obtained from a GMM and

$$\mathbf{A}_{\mathtt{c}t} = \sum_{c=1}^{C}\gamma_{\mathtt{c}t}^{(c)}\mathbf{A}_{\mathtt{c}}^{(c)};\quad \mathbf{b}_{\mathtt{c}t} = \sum_{c=1}^{C}\gamma_{\mathtt{c}t}^{(c)}\mathbf{b}_{\mathtt{c}}^{(c)}; \quad (8)$$

The number of components in the front-end GMM, C, determines the number of transforms. As the transform alters the feature-spaces a likelihood normalisation term is required, thus

$$p(\boldsymbol{y}_t|m,\mathcal{M}_{\mathtt{c}},\mathcal{M}_{\mathtt{x}}) = |\mathbf{A}_{\mathtt{c}t}|\mathcal{N}(\mathbf{A}_{\mathtt{c}t}\boldsymbol{y}_t + \mathbf{b}_{\mathtt{c}t};\boldsymbol{\mu}_{\mathtt{x}}^{(m)},\boldsymbol{\Sigma}_{\mathtt{x}}^{(m)}) \quad (9)$$

Since the normalisation term is not a function of the component (and hence state or word) it does not impact the rank-ordering of the hypotheses, thus it can be ignored for decoding. The following auxiliary function can used to estimate the transform

$$\mathcal{Q}(\mathcal{M}_{\mathtt{c}},\hat{\mathcal{M}}_{\mathtt{c}}) = \sum_{t=1}^{T}\log(|\mathbf{A}_{\mathtt{c}t}|) - \frac{1}{2}\sum_{c=1}^{C}\sum_{j=1}^{d}\left(\mathbf{w}_{\mathtt{c}j}^{(c)}\left(\sum_{e=1}^{C}\mathbf{G}^{(cej)}\mathbf{w}_{\mathtt{c}j}^{(e)\mathsf{T}}\right) - 2\mathbf{w}_{\mathtt{c}j}^{(c)}\mathbf{k}^{(cj)\mathsf{T}}\right) \quad (10)$$

where $\mathbf{W}_{\mathtt{c}}^{(c)} = [\mathbf{A}_{\mathtt{c}}^{(c)}\ \mathbf{b}_{\mathtt{c}}^{(c)}]$

$$\mathbf{G}^{(cej)} = \sum_{m=1}^{M}\sum_{t=1}^{T}\frac{\gamma_{\mathtt{c}t}^{(c)}\gamma_{\mathtt{c}t}^{(e)}\gamma_t^{(m)}}{\sigma_{\mathtt{x}j}^{(m)2}}\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}} \quad (11)$$

$$\mathbf{k}^{(cj)} = \sum_{m=1}^{M}\sum_{t=1}^{T}\frac{\gamma_{\mathtt{c}t}^{(c)}\gamma_t^{(m)}\mu_{\mathtt{x}j}^{(m)}}{\sigma_{\mathtt{x}j}^{(m)2}}\boldsymbol{\zeta}_t^{\mathsf{T}} \quad (12)$$

and summation for m is over all M ASR HMM components.

It is possible to use gradient-descent based techniques to directly optimise (10). However the memory requirement becomes impractical as the number of transforms C increases. Two simplifications are possible during training: only use the transform with the highest GMM posterior; use the posteriors to "weight" the observation contributions to C independent transform estimations. The second approximation does not guarantee an increase in likelihood, unlike the maximum approximation, but was found to perform best in preliminary experiments. Thus

$$\mathcal{Q}(\mathcal{M}_{\mathtt{c}},\hat{\mathcal{M}}_{\mathtt{c}}) = \sum_{c=1}^{C}\left[\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_{\mathtt{c}t}^{(c)}\gamma_t^{(m)}\log(|\mathbf{A}_{\mathtt{c}}^{(c)}|) - \frac{1}{2}\sum_{j=1}^{d}\left(\mathbf{w}_{\mathtt{c}j}^{(c)}\mathbf{G}^{(cj)}\mathbf{w}_{\mathtt{c}j}^{(c)\mathsf{T}} - 2\mathbf{w}_{\mathtt{c}j}^{(c)}\mathbf{k}^{(cj)\mathsf{T}}\right)\right] \quad (13)$$

where $\mathbf{k}^{(cj)}$ is given by (12) and

$$\mathbf{G}^{(cj)} = \sum_{m=1}^{M}\sum_{t=1}^{T}\frac{\gamma_{\mathtt{c}t}^{(c)}\gamma_t^{(m)}}{\sigma_{\mathtt{x}j}^{(m)2}}\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}} \quad (14)$$

This form is similar to (4), except that statistics are weighted by $\gamma_{\mathtt{c}t}^{(c)}$, and the standard CMLLR update formulae can be used.

FE-CMLLR can be compared to stereo-mapping MMSE approaches [8, 9]. A front-end GMM is again used and the estimate of the clean speech is given by (7). However MMSE, not ML, estimation based on a joint Gaussian distribution is used.

$$\mathbf{A}_{\mathtt{c}}^{(c)} = \boldsymbol{\Sigma}_{\mathtt{xy}}^{(c)}\boldsymbol{\Sigma}_{\mathtt{y}}^{(c)\text{-}1};\quad \mathbf{b}_{\mathtt{c}}^{(c)} = \boldsymbol{\mu}_{\mathtt{x}}^{(c)} - \boldsymbol{\Sigma}_{\mathtt{xy}}^{(c)}\boldsymbol{\Sigma}_{\mathtt{y}}^{(c)\text{-}1}\boldsymbol{\mu}_{\mathtt{y}}^{(c)} \quad (15)$$

The joint Gaussian distribution is estimated using stereo data

$$\left[\begin{array}{c}\boldsymbol{x}_t\\ \boldsymbol{y}_t\end{array}\right]\Bigg| c \sim \mathcal{N}\left(\left[\begin{array}{c}\boldsymbol{\mu}_{\mathtt{x}}^{(c)}\\ \boldsymbol{\mu}_{\mathtt{y}}^{(c)}\end{array}\right],\left[\begin{array}{cc}\boldsymbol{\Sigma}_{\mathtt{xx}}^{(c)} & \boldsymbol{\Sigma}_{\mathtt{xy}}^{(c)}\\ \boldsymbol{\Sigma}_{\mathtt{yx}}^{(c)} & \boldsymbol{\Sigma}_{\mathtt{y}}^{(c)}\end{array}\right]\right) \quad (16)$$

This form of enhancement yields the same general form as FE-CMLLR. However there are two important differences:

1. *training criterion*: for FE-CMLLR an ML-based approach is used, rather than an MMSE estimate.
2. *stereo data*: for MMSE estimates stereo data is required. In general this form of data is not available. In contrast FE-CMLLR just needs data from the channel.

In this work clean and noisy (retransmitted) data is available, but it is not synced. Thus MMSE approaches were not used.

4. Channel and Speaker Adaptation

Given the large amounts of data typically available for each of the channels it is possible to robustly train individual channel, or link, models with large numbers of transforms, in this case 128. However if these link models are to be combined with speaker level adaptation, either for adaptive training or test, then the interaction of the two transforms needs to be considered.

In CMLLR individual components are uniquely assigned to regression classes, thus there are no constraints that different transforms should yield consistent feature-spaces. If the number of speaker transforms is greater than the number of link transforms this is not a problem. However, if this is not the case the form of link adaptation can significantly influence the speaker adaptation. In this section the options for this interaction are described in terms of CMLLR transforms, MLLR speaker transforms act similarly.

1) Normalised-space adaptation: here transforms are estimated in the transformed space [5]. Thus

$$p(\boldsymbol{y}_t|s,m,\mathcal{M}_{\mathtt{x}},\mathcal{M}_{\mathtt{lnk}},\mathcal{M}_{\mathtt{spk}}) = |\mathbf{A}^{(s)}_{\mathtt{spk}}||\mathbf{A}^{(r_m)}_{\mathtt{lnk}}|\times \quad (17)$$
$$\mathcal{N}\left(\mathbf{A}^{(s)}_{\mathtt{spk}}(\mathbf{A}^{(r_m)}_{\mathtt{lnk}}\boldsymbol{y}_t + \mathbf{b}^{(r_m)}_{\mathtt{lnk}}) + \mathbf{b}^{(s)}_{\mathtt{spk}}; \boldsymbol{\mu}^{(m)}_{\mathtt{x}}, \boldsymbol{\Sigma}^{(m)}_{\mathtt{x}}\right)$$

This is computationally efficient. However the resulting transformed features are not co-ordinated, the feature-spaces for each regression class are estimated independently.

2) Full-covariance adaptation: here CMLLR is explicitly applied to the models, rather than implicitly via transforming the features. This yields

$$p(\boldsymbol{y}_t|s,m,\mathcal{M}_{\mathtt{x}},\mathcal{M}_{\mathtt{lnk}},\mathcal{M}_{\mathtt{spk}}) = |\mathbf{A}^{(s)}_{\mathtt{spk}}||\mathbf{A}^{(r_m)}_{\mathtt{lnk}}|\times \quad (18)$$
$$\mathcal{N}\left(\mathbf{A}^{(r_m)}_{\mathtt{lnk}}(\mathbf{A}^{(s)}_{\mathtt{spk}}\boldsymbol{y}_t + \mathbf{b}^{(s)}_{\mathtt{spk}}) + \mathbf{b}^{(r_m)}_{\mathtt{lnk}}; \boldsymbol{\mu}^{(m)}_{\mathtt{x}}, \boldsymbol{\Sigma}^{(m)}_{\mathtt{x}}\right)$$

where the speaker transforms $\{\mathbf{A}^{(s)}_{\mathtt{spk}}, \mathbf{b}^{(s)}_{\mathtt{spk}}\}$ are estimated based on the model-transformed distributions for each component m

$$\mathcal{N}\left(\mathbf{A}^{(r_m)\text{-}1}_{\mathtt{lnk}}(\boldsymbol{\mu}^{(m)}_{\mathtt{x}} - \mathbf{b}^{(r_m)}_{\mathtt{lnk}}), \mathbf{A}^{(r_m)\text{-}1}_{\mathtt{lnk}}\boldsymbol{\Sigma}^{(m)}_{\mathtt{x}}\mathbf{A}^{(r_m)\text{-}\mathtt{T}}_{\mathtt{lnk}}\right) \quad (19)$$

The transforms are estimated based on the original-space, which is by definition consistent for all the regression classes. Though a consistent space for the transform estimation, the full covariance matrices in (19) make these transforms more computationally expensive to compute [10].

3) Implicit Co-ordination: rather than having a unique assignment of components, if components "contribute" to more than one transform it is possible to co-ordinate the resulting spaces. The simplest implementation of this is to smooth the regression class specific counts with global regression class counts. Though this improves the feature-space co-ordination, the transforms will become "smoother". This may limit the benefits of large numbers of transforms.

Methods (1) and (2) yield the same result when the speaker and link regression-classes are the same. Both methods (2) and (3) above offer approaches that help to co-ordinate the estimated degraded channel transforms, but both have drawbacks. Note if MLLR, rather than CMLLR is used for channel modelling then there is no feature-space transformation, so the space will be consistent. However MLLR is significantly more expensive to use for adaptive training [4].

Alternatively if FE-CMLLR transforms are used the space should be consistent (co-ordinated). The transform is determined by the posteriors of the front-end GMM, rather than the component. This means that the same component will make use of multiple transforms, ensuring consistency over the transform spaces. It is thus possible to directly estimate the speaker transforms in the space defined by the FE-CMLLR transform.

$$p(\boldsymbol{y}_t|s,m,\mathcal{M}_{\mathtt{x}},\mathcal{M}_{\mathtt{c}},\mathcal{M}_{\mathtt{spk}}) \propto |\mathbf{A}^{(s)}_{\mathtt{spk}}|\times \quad (20)$$
$$\mathcal{N}\left(\mathbf{A}^{(s)}_{\mathtt{spk}}(\mathbf{A}_{ct}\boldsymbol{y}_t + \mathbf{b}_{ct}) + \mathbf{b}^{(s)}_{\mathtt{spk}}; \boldsymbol{\mu}^{(m)}_{\mathtt{x}}, \boldsymbol{\Sigma}^{(m)}_{\mathtt{x}}\right)$$

where $\mathbf{A}_{ct}$ and $\mathbf{b}_{ct}$ are given by (8). FE-CMLLR allows efficient speaker adaptation in a space that is co-ordinated. This should yield better speaker adaptation than the normalised-space adaptation, avoids the computational cost of full-covariance adaptation, and avoids the smoothing effects of the implicit co-ordination.

5. Preliminary Experiments

The training and test data was taken from Robust Automatic Transcription of Speech (RATS) program for Arabic keyword-spotting. The data was collected by retransmitting Levantine Arabic conversation telephone speech data [1] down eight communication channels (links) labelled A to H. These links have a range of distortions associated with them. Only a subset of the original data was retransmitted. The total data for training the systems from all eight channels, plus the original clean speech, was 173 hours. Part of the retransmitted data was held-out as a test-set, dev1. For each of the channels there was between 2 and 2.5 hours of test data, depending how much data passed the quality assurance tests. As the test data was Levantine Arabic, there is little appropriate data for training language models. For this work all the clean Levantine Arabic transcriptions, excluding the dev1 test data, was used. This gave 1.6 million words for training. A trigram language model was trained using Kneser-Ney discounting.

The acoustic data was parametrised using 39-dimensional PLP-based front-end, including C0, with delta, delta-deltas and triples projected from 52 to 39 dimensions using HLDA. Speaker (side) based cepstral mean normalisation was applied. Word-based graphemic systems, incorporating word boundary information, were built. Though performance gains can be obtained with phonetic systems and MADA decomposition, the word context dependence of the decomposition and pronunciations make their use in a keyword spotting system (the final task) more complicated. State-clustered cross-word triphone HMM acoustic models were trained using MPE. A total of about 3K distinct states were used with an average of 36 components per state. All adaptation transforms were full in nature. For the link representations 128 CMLLR transforms were used, both for CMLLR and FE-CMLLR, with link adaptive training (LAT). A single GMM for all channels was used in FE-CMLLR. For speaker adaptation speech and silence CMLLR transforms were used, similarly in SAT. The CMLLR speaker transforms were estimated using normalised-space adaptation, method (1).

For the test data the link was assumed to be known, as well as the segmentation of the utterances. To save space three representative links in terms of distortions are given in the tables: link A (high); link C (medium); and link G (low). In addition the overall average A-H is also quoted. All results are based on confusion network (CN) decoding.

Table 1: *Unadapted (for speaker) Decoding Experiments.*

System	Link XForm	WER%			
		A	C	G	Avg
SI	—	80.0	70.6	61.5	71.9
LAT	CMLLR	73.6	69.1	61.7	70.1
LAT	FE-CMLLR	73.9	69.1	60.7	70.1

[1] The transcription of dialectic Arabic is highly challenging. The error rates on non-degraded conversational telephone speech are typically in the range 40%-50%, for example see [11].

Table 1 shows the unadapted performance of multi-style speaker-independent (SI), and CMLLR and FE-CMLLR LAT systems. As expected the overall performance is poor, reflecting the challenging nature of this task. Even the low distortion link (G) had an error rate of around 60%. For the more distorted links the use of LAT, CMLLR or FE-CMLLR based, yields good gains over the SI system. Both LAT systems (CMLLR or FE-CMLLR based) produced similar average performance.

Table 2: *MLLR Speaker Adaptation Experiments.*

System		Link XForm	WER% A	C	G	Avg
P2a	SI	—	77.7	70.1	61.0	71.2
P2b	LAT	CMLLR	73.6	68.7	61.1	69.8
P2c	LAT	FE-CMLLR	73.6	68.2	60.1	69.6

Table 2 shows the performance of the baseline, CMLLR and FE-CMLLR systems using unsupervised speaker-level MLLR adaptation at test-time. The overall performance gains of adaptation are small, reflecting the low accuracy of the supervision. However the performance of FE-CMLLR is slightly better than CMLLR. This indicates that the FE-CMLLR space may be more co-ordinated than the standard CMLLR system.

Table 3: *CMLLR Speaker Adaptive Training Experiments.* $\otimes$ *indicates CNC, supervision for adaptation taken from Table 2.*

System Supervision		Link XForm	WER% A	C	G	Avg
SAT	P2a	—	73.5	67.6	58.6	68.8
SAT	P2b	—	71.9	67.2	58.6	68.4
LAT	P2b	CMLLR	72.5	68.2	59.8	69.1
LSAT	P2b	CMLLR	71.1	67.2	58.3	67.9
LSAT	P2c	FE-CMLLR	71.1	66.2	56.6	67.0
LSAT-P2b $\otimes$ SAT-P2b			70.5	66.2	57.3	67.1
LSAT-P2c $\otimes$ LSAT-P2b			70.0	65.3	56.1	66.2

If the FE-CMLLR space is more appropriate for speaker adaptation, this should be more clearly shown when combined with speaker adaptive training (SAT). A range of SAT and link and speaker adaptively trained systems (LSAT) were built. The supervision hypotheses were based on the output from the systems in Table 2 and speech and silence CMLLR transforms used in both training and test. The baseline SAT system using the SI system supervision (P2a) yielded gains over the SI system in Table 2. Here, the SAT system models some of the attributes of the links. Improving the supervision (P2b) yields additional small gains. The performance of this SAT P2b system is better than the LAT P2b system, indicating the usefulness of speaker specific transforms in training for this task. The LSAT system based on CMLLR LSAT P2b gave gains over the SAT system. This shows that the limited number of transforms that can be robustly estimated for each speaker do not fully model the link distortions. However the impact of the lack of a co-ordinated space for adaptation is demonstrated by comparing with the LSAT P2c system where FE-CMLLR was used. The gain of using LSAT over SAT for the low distortion link is increased from 0.3% absolute (CMLLR) to 2.0% absolute (FE-CMLLR). The average performance gain (links A-H) was 1.8% absolute compared to the baseline SAT P2a system.

In addition it is possible to combine various combinations of the output from multiple SAT and LSAT systems. The results for this using CN combination (CNC) are shown in the bottom two lines of Table 3. By combining the two LSAT systems additional gains of 0.5%-1.1% and an overall gain of 0.8% absolute.

6. Conclusions

This paper has discussed the possibilities and issues with modelling distorted channels with predictive and adaptive approaches. When combined with speaker adaptation it is important that the form of the link modelling yields a consistent, co-ordinated, space for adaptation. For standard CMLLR adaptation this is not the case, however using a front-end version of this scheme, FE-CMLLR, a consistent space from the link model is obtained. The performance of the system was evaluated on a highly challenging task, degraded communication channels with dialect (Levantine) Arabic conversational telephone speech. Though the overall performance of the systems is poor, 60%-70% WER, the trends are consistent with the theory. In particular for the low distortion link (G) the advantages of using FE-CMLLR compared to CMLLR are more pronounced.

This paper has only considered one version of FE-CMLLR. It is possible to extend this in a number of ways: link-specific front-end GMMs; acoustic de-weighting on the GMM likelihoods to yield smoother spaces; and the use of discriminative training (which deals with the normalisation issues). It is interesting to note that discriminative FE-CMLLR transforms are related to region dependent transforms (RDTs) [12]. However the transforms now become region and link dependent. ML FE-CMLLR transforms should yield an efficient approach for initialising these region and link discriminative transforms.

7. References

[1] A. Acero, L. Deng, T. T. Kristjansson, and J. Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," in *Proc. ICSLP*, Beijing, China, Oct. 2000, pp. 869–872.

[2] C. J. Legetter, "Improved acoustic modelling for HMMs using linear transformations," Ph.D. dissertation, Cambridge Univ., 1995.

[3] M. J. F. Gales, "Maximum Likelihood Linear Transformations For HMM-Based Speech Recognition," *Computer Speech and Language*, vol. 12, Jan. 1998.

[4] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A Compact Model for Speaker Adaptive Training," in *ICSLP'96*, Philadelphia, 1996.

[5] M. J. F. Gales, "Adapting semi-tied full-covariance HMMs," Cambridge University, Tech. Rep. CUED/F-INFENG/TR298, 1997, available via anonymous ftp from: svr-ftp.eng.cam.ac.uk.

[6] H. Liao and M. J. F. Gales, "Joint uncertainty decoding for noise robust speech recognition," in *Proc. Interspeech*, 2005.

[7] F. Flego and M. J. F. Gales, "Discriminative adaptive training with VTS and JUD," in *Proc. ASRU*, 2009.

[8] X. D. Huang, A. Acero, and H. W. Hon, *Spoken Language Processing*. Prentice Hall, 2001.

[9] M. Afify, X. Cui, and Y. Gao, "Stereo-based stochastic mapping for robust speech recognition," in *Proc. ICASSP*, 2007.

[10] K. C. Sim and M. J. F. Gales, "Adaptation of precision matrix models on large vocabulary continuous speech recognition," in *ICASSP*, 2005.

[11] D. Vergyri, K. Kirchhoff, R. Gadde, A. Stolcke, and J. Zheng, "Development of a conversational telephone speech recognizer for Levantine Arabic," in *Proc. Interspeech*, 2005.

[12] B. Zhang, S. Matsoukas, and R. Schwartz, "Discriminatively trained region dependent transforms for speech recognition," in *Proc. ICASSP*, 2006.