Model-Based Approaches to Speaker and Environment Adaptation

Mark Gales

April 2009

Cambridge University Engineering Department

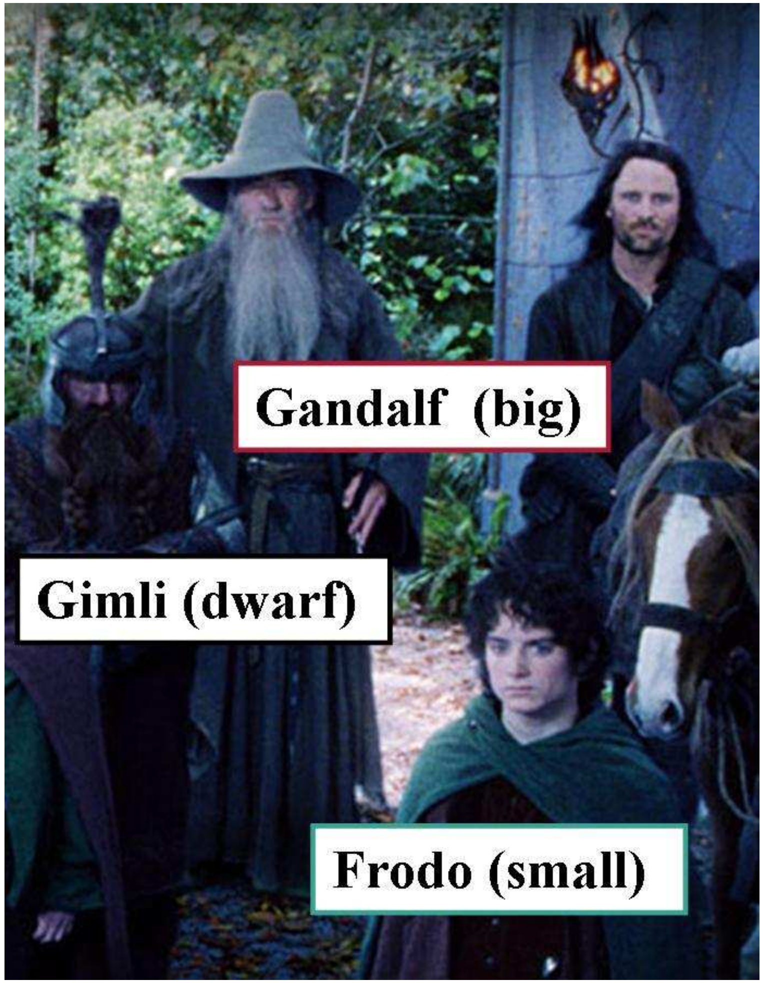
Overview

- Speaker Adaptation - “Adaptive”
  - linear transform-based adaptation / adaptive training
- Extensions to Linear-Transform Approaches
  - Bayesian adaptive training and inference
  - discriminative mapping functions
  - noisy constrained MLLR
- Noise Robust Speech Recognition - “Predictive”
  - model-based approaches / ML noise estimation
- Extensions to Model-Based Approaches
  - joint uncertainty decoding
  - predictive linear transforms
  - adaptive training / incremental adaptation

Speaker Adaptation

Speaker Adaptation

- Large differences between speakers
- Linguistic Differences e.g.
  - Accents
    tomato in RP/American English
  - Speaker idiosyncrasies
    either in English
  - non-native speaker
- Physiological Differences e.g.
  - physical attributes - gender, length of vocal tract
  - transitory effects
    cold/stress/public speaking



Adaptation Modes

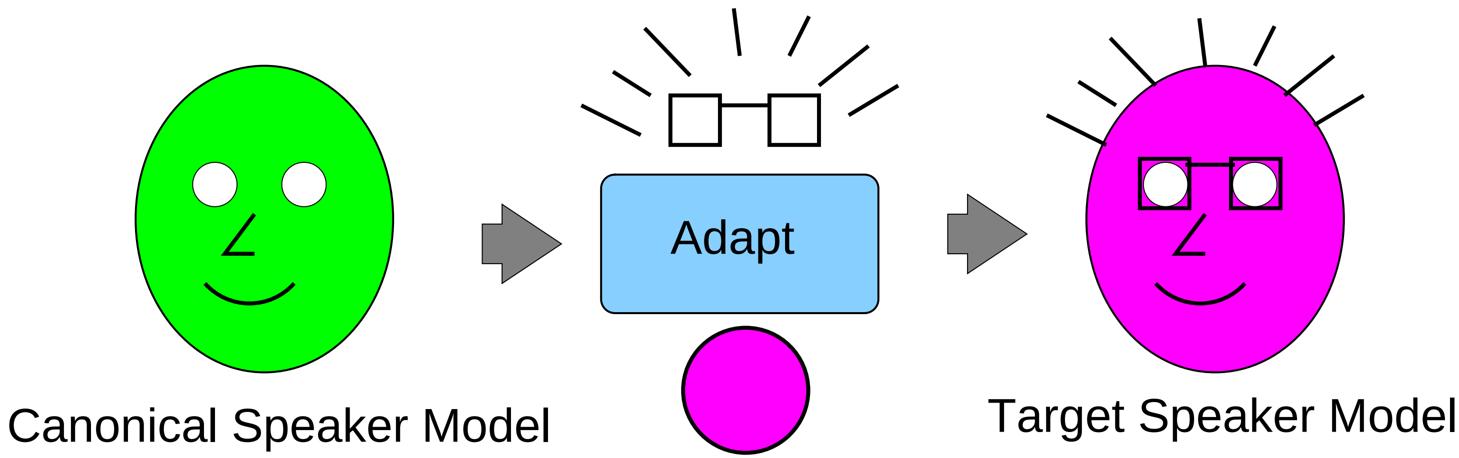
- Speaker/environment adaptation is an essential part of LVCSR systems
  - obtain the performance of a Speaker/Environment dependent system with orders-of-magnitude less data (30 seconds vs 2000 hours!)
- The mode of adaptation depends on the task being investigated
  - incremental: results are required causally, the adaptation data is not all available in one block - dictation tasks, car navigation
  - batch: all the data is available (or can be used) in one block - BN transcription, CTS transcription

In addition for batch adaptation the adaptation data may be

- supervised: the correct transcription of the adaptation data is known (dictation enrolment)
- unsupervised: no transcribed adaptation data available, transcription must be hypothesised (BN transcription)

General Adaptation Process

- Aim: Modify a "canonical" model to represent a target speaker
  - transformation should require minimal data from the target speaker
  - adapted model should accurately represent target speaker



- Need to determine
  - nature (and complexity) of the speaker transform
  - how to train the "canonical" model that is adapted

Form of the Adaptation Transform

- There are a number of standard forms in the literature [1].
- Maximum A-Posteriori MAP [2] adaptation: general “robust” estimation
  - in simplest form only adapts “seen” components
- Speaker Clustering: Gender-dependent (GD) models are the simplest from:
  - often estimated using MAP adaptation with speaker-independent priors

  EigenVoices[3], CAT [4] are more complex forms.
- Vocal Tract Length Normalisation: motivated from physiological perspective
- Linear Transform Adaptation: dominant form for LVCSR
  - will be the focus of this part of the talk

Form of the Adaptation Transform

- Dominant form for LVCSR are ML-based linear transformations
  - MLLR adaptation of the means [5]

$$\boldsymbol{\mu}^{(s)} = \mathbf{A}^{(s)}\boldsymbol{\mu} + \mathbf{b}^{(s)}$$

  - MLLR adaptation of the covariance matrices [6, 7]

$$\boldsymbol{\Sigma}^{(s)} = \mathbf{H}^{(s)}\boldsymbol{\Sigma}\mathbf{H}^{(s)\mathsf{T}}$$

  - Constrained MLLR adaptation [7]

$$\boldsymbol{\mu}^{(s)} = \mathbf{A}^{(s)}\boldsymbol{\mu} + \mathbf{b}^{(s)}; \quad \boldsymbol{\Sigma}^{(s)} = \mathbf{A}^{(s)}\boldsymbol{\Sigma}\mathbf{A}^{(s)\mathsf{T}}$$

- Forms may be combined into a hierarchy [8] e.g.

$$\texttt{CMLLR} \longrightarrow \texttt{MLLRMEAN}$$

ML and MAP Linear Transforms

- Transforms often estimated using ML (with hypothesis $\mathcal{H}$)

$$\mathbf{W}^{(s)}_{\texttt{ml}} = \arg\max_{\mathbf{W}} \left\{ p(\mathbf{O}^{(s)}|\mathcal{H};\mathbf{W}) \right\}$$

  - where $\mathbf{W}^{(s)}_{\texttt{ml}} = \left[\mathbf{A}^{(s)}_{\texttt{ml}} \quad \mathbf{b}^{(s)}_{\texttt{ml}} \right]$
  - however not robust to limited training data

- Including transform prior, $p(\mathbf{W})$, to get MAP estimate [9]

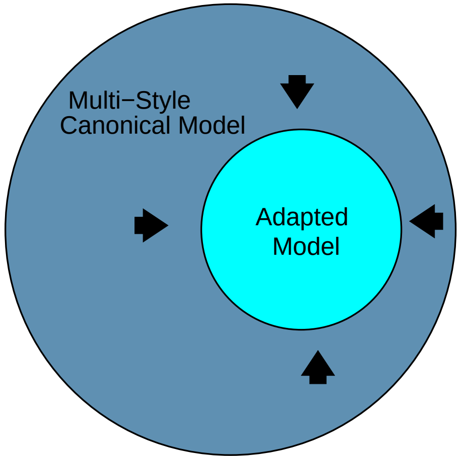
$$\mathbf{W}^{(s)}_{\texttt{map}} = \arg\max_{\mathbf{W}} \left\{ p(\mathbf{O}^{(s)}|\mathcal{H};\mathbf{W})p(\mathbf{W}) \right\}$$

  - for MLLR Gaussian is a Gaussian prior for the auxiliary function
  - CMLLR prior more challenging ...

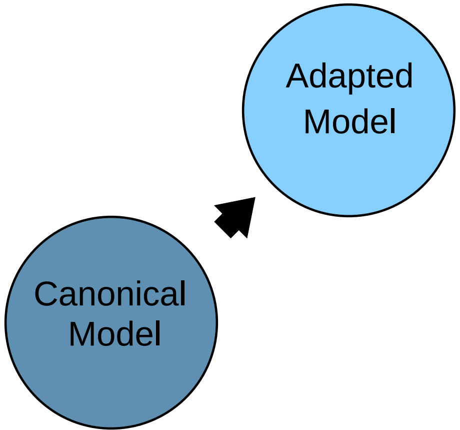
- Both approaches rely on expectation-maximisation (EM)

Training a "Good" Canonical Model

- Standard "multi-style" canonical model
  - treats all the data as a single "homogeneous" block
  - model represents acoustic realisation of phones/words (desired)
  - and acoustic environment, speaker, speaking style variations (unwanted)



(a) Multi-Style System

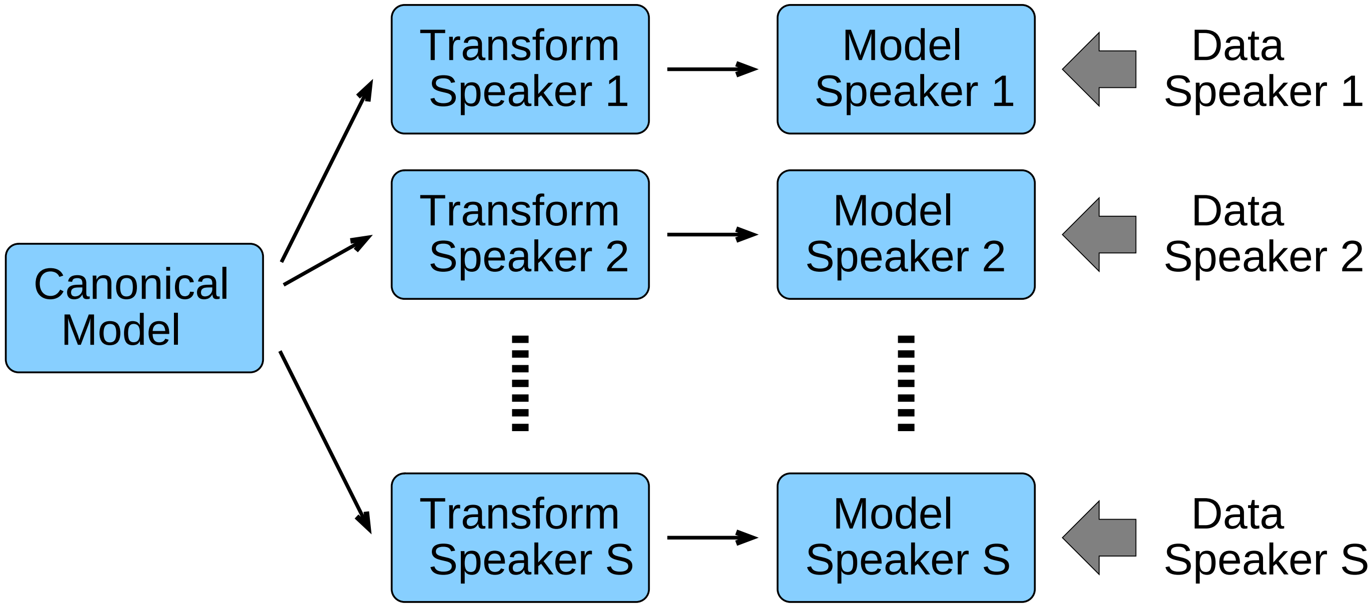


(b) Adaptive System

Two different forms of canonical model:

- Multi-Style: adaptation converts a general system to a specific condition;
- Adaptive: adaptation converts a "neutral" system to a specific condition [10, 7]

Adaptive Training



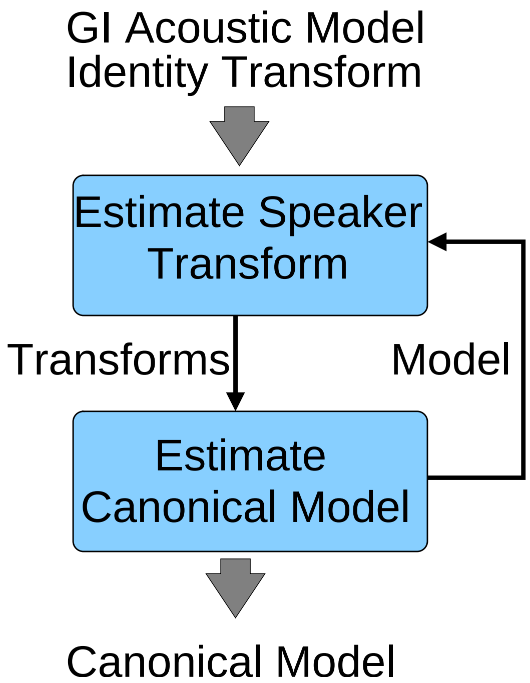
- In adaptive training the training corpus is split into "homogeneous" blocks
  - use adaptation transforms to represent unwanted acoustic factors
  - canonical model only represents desired variability
- All forms of linear transform can be used for adaptive training
  - CMLLR adaptive training highly efficient

CMLLR Adaptive Training

- The CMLLR likelihood may be expressed as [7]:

$$\mathcal{N}(\boldsymbol{o}_t; \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{T}}) = \frac{1}{|\mathbf{A}|}\mathcal{N}(\mathbf{A}^{-1}\boldsymbol{o}_t - \mathbf{A}^{-1}\mathbf{b}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

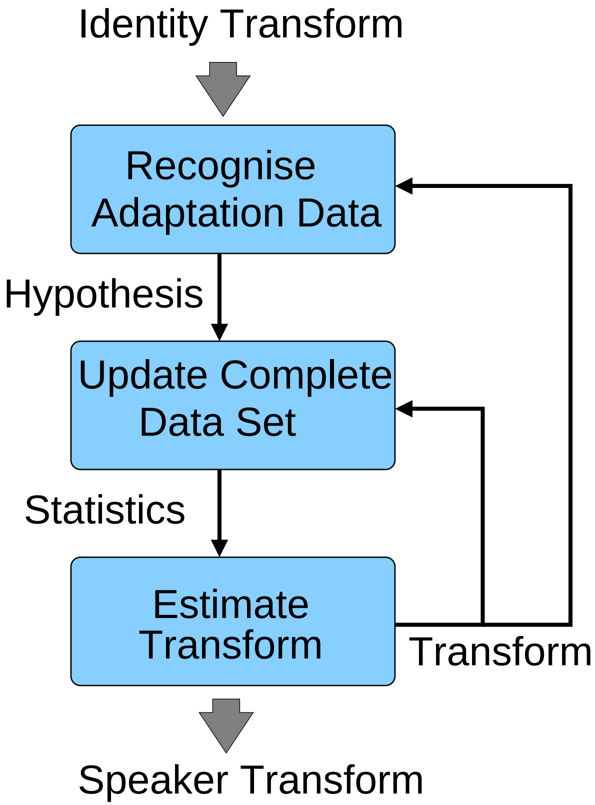
same as feature normalisation - simply train model in transformed space



- Interleave Model and transform estimation
- Adaptive canonical model not suited for unadapted initial decode
  - GI model used for initial hypothesis
- MLLR less efficient, but still reasonable

Unsupervised Linear Transformation Estimation

- Estimation of all the transforms is based on EM:
  - requires the transcription/hypothesis of the adaptation data
  - iterative process using "current" transform to estimate new transform

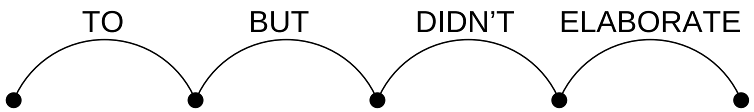


- Two iterative loops for estimation:
  1. estimate hypothesis given transform
  2. update complete-dataset given transform and hypothesis

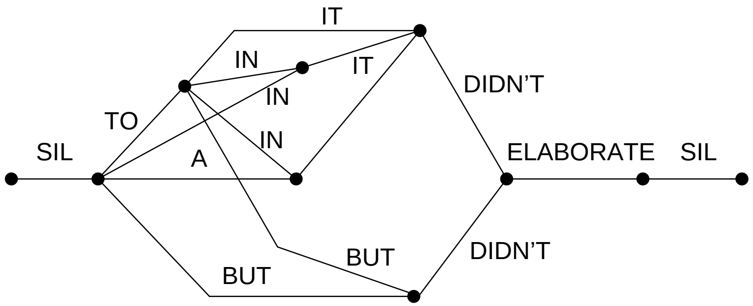
  referred to as Iterative MLLR [11]
- For supervised training hypothesis is known
- Confidence-scores can also be used
  - confidence-based MLLR [12]

Lattice-Based MLLR

- For unsupervised adaptation hypothesis will be error-full
- Rather than using the 1-best transcription and iterative/confidence MLLR
  - generate a lattice when recognising the adaptation data [12]
  - accumulate statistics over the lattice (Lattice-MLLR)



1-best transcription



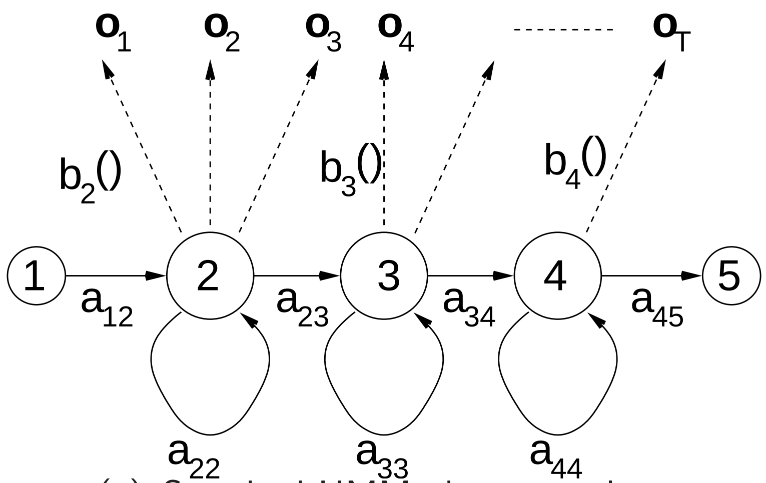
Word lattice

- The accumulation of statistics is closely related to obtaining denominator statistics for discriminative training
- No need to re-recognise the data
  - iterate over the transform estimation using the same lattice

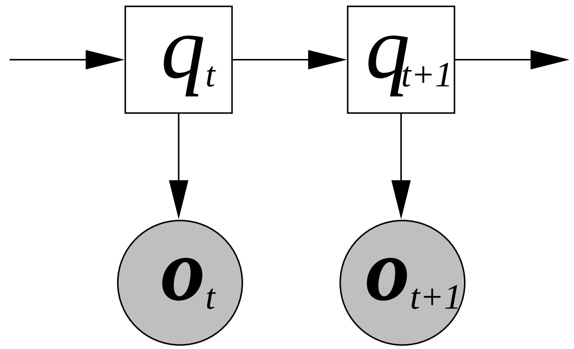
Extensions to Linear Transform Approaches

- Bayesian Adaptive Training and Inference:
  - HMMs as a dynamic Bayesian network
  - transform parameters embedded in acoustic model
  - integrated (instantaneous) adaptation and recognition
- Discriminative Mapping Transforms:
  - efficient and robust approach to obtaining discriminative linear transforms
- Noisy Constrained MLLR:
  - ML-estimated transform suitable for both noise and speaker adaptation
  - integration into adaptive training framework

Hidden Markov Model - A Dynamic Bayesian Network



(c) Standard HMM phone topology



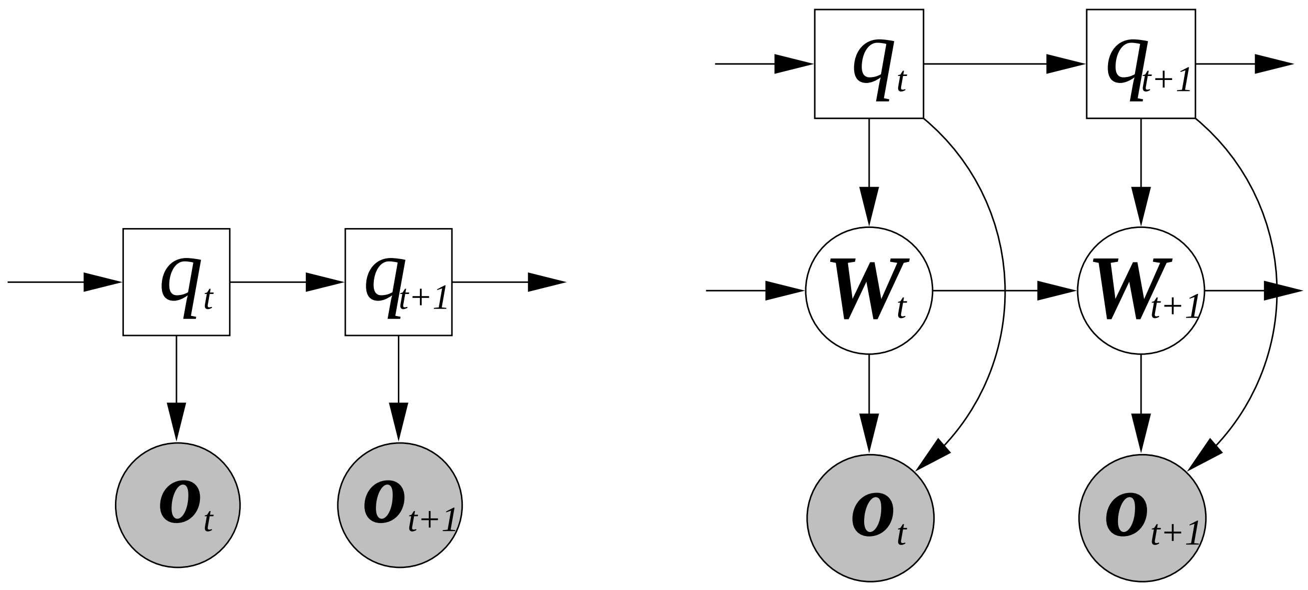
(d) HMM Dynamic Bayesian Network

- Notation for DBNs:

circles - continuous variables	shaded - observed variables
squares - discrete variables	non-shaded - unobserved variables

- Observations conditionally independent of other observations given state.
- States conditionally independent of other states given previous states.
- Poor model of the speech process - piecewise constant state-space.

Adaptive Training From Bayesian Perspective



(e) Standard HMM (f) Adaptive HMM

- Observation additionally dependent on transform $\mathbf{W}_t$ [13]
  - transform same for each homogeneous block ($\mathbf{W}_t = \mathbf{W}_{t+1}$)
  - adaptation integrated into acoustic model - instantaneous adaptation
- Need to known the prior transform distribution $p(\mathbf{W})$ (as in MAP scheme)

Inference with Adaptive HMMs

- Acoustic score - marginal likelihood of the whole sequence, $\mathbf{O} = \mathbf{o}_1, \ldots, \mathbf{o}_T$
  - still depends on the hypothesis $\mathcal{H}$
  - point-estimate canonical parameters (standard complexity control schemes)

$$
\begin{aligned}
p(\mathbf{O}|\mathcal{H}) &= \int_{\mathbf{W}} p(\mathbf{O}|\mathcal{H}, \mathbf{W}) p(\mathbf{W})\, d\mathbf{W} \\
&= \int_{\mathbf{W}} \sum_{\mathbf{q} \in \mathbf{Q}^{(\mathcal{H})}} P(\mathbf{q}) \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{o}_t; \mathbf{A}\boldsymbol{\mu}^{(q_t)} + \mathbf{b}, \boldsymbol{\Sigma}^{(q_t)}) p(\mathbf{W})\, d\mathbf{W}
\end{aligned}
$$

- Latent variables makes exact inference impractical
  - need to sum over all possible state-sequences explicitly
  - Viterbi decoding not possible to find best hypothesis
- Need schemes to handle both these problems [13]
  - variational Bayes/MAP N-best supervision/adaptation/rescoring

Utterance Level Bayesian Adaptation

- Initial evaluations on English Conversational Speech recognition task

Bayesian Approx	ML Train SI	ML Train SAT
—	32.8	—
ML	35.5	35.2
MAP	32.2	31.8
VB	31.8	31.5

- All experiments use N-best supervision
  - ML adaptation much worse than SI - insufficient adaptation data
  - VB yields additional gains over MAP
  - Note: N-best supervision better than 1-best (0.5% for VB-SAT)
- SAT performance better than SI performance
  - gains from adaptive HMM 1.3% absolute over SI baseline
  - integrated adaptation seems to be useful (though implementation an issue)

Discriminative Linear Transforms

- Linear transforms can be trained using discriminative criteria [14, 15]
  - estimation using minimum phone error (MPE) training

$$\mathbf{W}_{\mathrm{d}}^{(s)} = \arg\min_{\mathbf{W}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\mathbf{O}^{(s)};\mathbf{W})\mathcal{L}(\mathcal{H},\mathcal{H}^{(s)}) \right\}.$$

- For unsupervised adaptation discriminative linear transforms (DLTs) not used
  - estimation highly sensitive to errors in supervision hypothesis
  - more costly to estimate transform than ML training
- Not used for discriminative SAT [16], standard procedure
  1. perform standard ML-training (ML-SI)
  2. perform ML SAT training updating models and transforms (ML-SAT)
  3. estimate MPE-models given the ML-transforms (MPE-SAT)

Discriminative Mapping Functions

- Would like to get aspects of discriminative transform without the problems:
  - train all speaker-specific parameters using ML training
  - train speaker-independent parameters using MPE training
- Applying this to linear transforms yields (as one option) [17]

$$\begin{aligned}\boldsymbol{\mu}^{(s)} &= \mathbf{A}_{\mathtt{d}}\left(\mathbf{A}_{\mathtt{ml}}^{(s)}\boldsymbol{\mu} + \mathbf{b}_{\mathtt{ml}}^{(s)}\right) + \mathbf{b}_{\mathtt{d}} \\ &= \mathbf{A}_{\mathtt{d}}\boldsymbol{\mu}_{\mathtt{ml}}^{(s)} + \mathbf{b}_{\mathtt{d}}\end{aligned}$$

  - $\mathbf{W}_{\mathtt{ml}}^{(s)} = \left[\mathbf{A}_{\mathtt{ml}}^{(s)} \quad \mathbf{b}_{\mathtt{ml}}^{(s)}\right]$ - speaker-specific ML transform
  - $\mathbf{W}_{\mathtt{d}} = \left[\mathbf{A}_{\mathtt{d}} \quad \mathbf{b}_{\mathtt{d}}\right]$ - speaker-independent MPE transform
- Yields a composite discriminative-like transform

$$\mathbf{A}_{\mathtt{d}}^{(s)} = \mathbf{A}_{\mathtt{d}}\mathbf{A}_{\mathtt{ml}}^{(s)}; \quad \mathbf{b}_{\mathtt{d}}^{(s)} = \mathbf{A}_{\mathtt{d}}\mathbf{b}_{\mathtt{ml}}^{(s)} + \mathbf{b}_{\mathtt{d}}$$

Training DMTs

- This form of DMT results in the following estimation criterion

$$\mathbf{W}_{\tt d} = \arg\min_{\mathbf{W}} \left\{ \sum_{s} \sum_{\mathcal{H}} P(\mathcal{H}|\mathbf{O}^{(s)}; \mathbf{W}, \mathbf{W}_{\tt ml}^{(s)}) \mathcal{L}(\mathcal{H}, \mathcal{H}^{(s)}) \right\}.$$

  - posterior $P(\mathcal{H}|\mathbf{O}^{(s)}; \mathbf{W}, \mathbf{W}_{\tt ml}^{(s)})$ based on speaker ML-adapted models
  - supervised training of discriminative transform
- Standard DLT update formulae can be used
- Quantity of training data vast compared to available speaker-specific data
  - use large number of base-classes
  - in these experiments 1000 base-classes used
- Can also be used for adaptive training [18]
  - closer to full discriminative adaptive training

Discriminative Adaptive Training with DMTs

- Initial evaluations on English Conversational Speech recognition task

Training Scheme	Transform Training	Transform Testing	WER (%) eval03
SI	—	—	29.2
SI	—	MLLR	27.0
		MLLR+DMT	26.2
DSAT	MLLR	MLLR	26.4
	MLLR	MLLR+DMT	25.6
	DLT	DLT	28.1
	MLLR+DMT	MLLR+DMT	25.3

- All systems trained using MPE (both multi-style and adaptive)
- As expected adaptation helps with the multi-style trained system
  - DMTs help with the multi-style trained system (0.8% absolute)
  - DMTs help with adaptively trained system (1.1% absolute)

Noisy CMLLR

- Linear transforms described are general
  - hierarchies allow very complex forms to be used
  - interesting to examine forms aimed at particular tasks
- Noisy CMLLR is aimed at noise-robust speech [19] recognition

$$p(\boldsymbol{o}_t; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)}, \mathbf{A}, \mathbf{b}, \boldsymbol{\Sigma}_{\mathtt{b}}) = |\mathbf{A}|\mathcal{N}(\mathbf{A}\boldsymbol{o}_t + \mathbf{b}; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)} + \boldsymbol{\Sigma}_{\mathtt{b}})$$

  - has the same form as a model-based compensation scheme (JUD)
- Similar to CMLLR, but with an additional bias on the variance
  - CMLLR can be viewed as estimating the "neutral" speech
  - the variance bias, $\boldsymbol{\Sigma}_{\mathtt{b}}$, a level of uncertainty
- Form can be used in an adaptive training/discriminative fashion as well

Noisy CMLLR and Factor Analysis

- The estimation of/adaptive training of NCMLLR related to:
  - shared factor analysis approach for covariance matrix modelling [20]
  - EM-based VTS adaptive training for canonical model estimation [21]
- All treat "clean" speech as a latent variable
  - posterior distribution depends on the form being examined
  - update for canonical models:

$$\hat{\boldsymbol{\mu}}^{(m)} = \frac{\sum_{h=1}^{H}\sum_{t=1}^{T}\gamma_t^{(mh)}\mathcal{E}\left\{\mathbf{s}_t|\mathbf{o}_t, m\right\}}{\sum_{h=1}^{H}\sum_{t=1}^{T}\gamma_t^{(mh)}}$$

- Discriminative adaptive training also considered [19]

$$\hat{\boldsymbol{\mu}}^{(m)} = \frac{\sum_{h=1}^{H}\sum_{t=1}^{T}(\gamma_{\texttt{num}t}^{(mh)} - \gamma_{\texttt{den}t}^{(mh)})\mathcal{E}\left\{\mathbf{s}_t|\mathbf{o}_t, m\right\} + D_m\boldsymbol{\mu}^{(m)} + \tau_{\texttt{p}}\boldsymbol{\mu}_{\texttt{p}}^{(m)}}{\sum_{h=1}^{H}\sum_{t=1}^{T}(\gamma_{\texttt{num}t}^{(mh)} - \gamma_{\texttt{den},t}^{(mh)}) + D_m + \tau_{\texttt{p}}}$$

Noisy CMLLR vs CMLLR Performance

- Evaluated on engine-on/highway noise condition from the Toshiba data
  - phone-numbers task (unknown digit length sequences)
  - see next section for test data configuration, here run at speaker level
  - simplified training data set-up trained on noise-corrupted WSJ SI-284

System	Adapt (diag)	ENON		HWY	
		ML	MPE	ML	MPE
Multi-style	—	1.2	0.8	6.7	5.0
	CMLLR	0.3	0.3	2.4	2.0
	NCMLLR	0.5	0.6	2.1	1.9
Adaptive Training	CMLLR	0.3	0.2	2.1	1.5
	NCMLLR	0.3	0.2	1.8	1.2

- Adaptive training again shows gains over multi-style training
  - NCMLLR out-performs CMLLR at low SNR conditions
  - MPE gains larger when using adaptive training

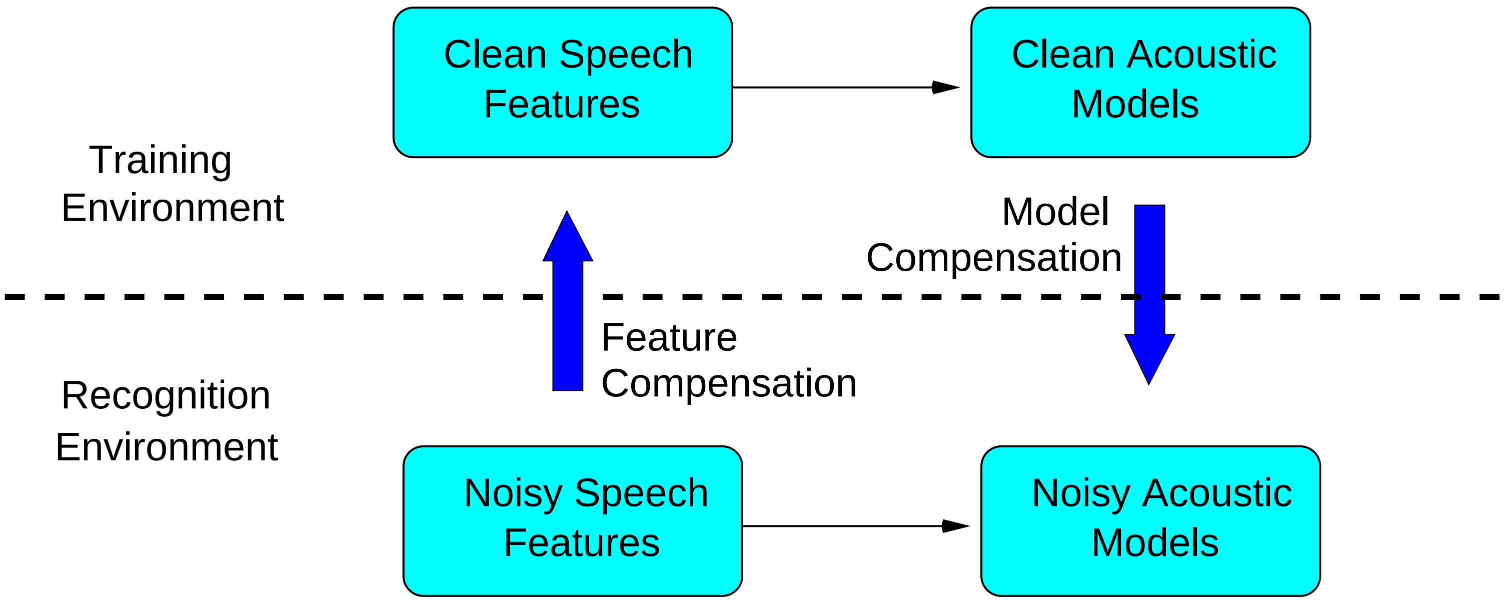
Speaker Adaptation Summary

- Speaker adaptation an important part of speech recognition systems
- Linear transform-based adaptation still dominant form for LVCSR adaptation
  - extensively used in CU-HTK and other evaluation systems
  - needs to be able to handle errors in the hypotheses
  - need to be able to discriminatively estimated transforms
- Adaptive training a theoretically very interesting extension
  - use adaptation transforms during training
  - allows a “neutral” speaker model to be generated
- Gains for speaker adaptive training still disappointing ...
- Though simple (just a linear transform) still issues to be addressed
  - e.g. integrating adaptation into acoustic model efficiently ...

Noise-Robust ASR

Noise Robust ASR - In-Car Navigation

Noise Compensation Approaches



- Two main approaches:
  - feature compensation: "clean" the noisy features
  - model compensation: "corrupt" the clean models
- This work concentrates on model compensation approaches
  - VTS and JUD examples, predictive model compensation schemes

Mismatch Functions

- Speech data is normally parameterised in the Cepstral domain, thus

$$\boldsymbol{y}_t^{\mathrm{s}} = \frac{1}{2}\mathbf{C}\log\left(\exp(2\mathbf{C}^{\text{-}1}\boldsymbol{x}_t^{\mathrm{s}} + 2\mathbf{C}^{\text{-}1}\boldsymbol{h}^{\mathrm{s}}) + \exp(2\mathbf{C}^{\text{-}1}\boldsymbol{n}_t^{\mathrm{s}})\right) = \boldsymbol{x}_t^{\mathrm{s}} + \boldsymbol{h}^{\mathrm{s}} + f(\boldsymbol{x}_t^{\mathrm{s}}, \boldsymbol{n}_t^{\mathrm{s}}, \boldsymbol{h}^{\mathrm{s}})$$

  $\mathbf{C}$ is the DCT, magnitude-based Cepstra

  - non-linear relationship between the clean speech, noise and corrupted speech
  - not possible to get simple expression for all parameterisations

- This has assumed sufficient smoothing to remove all "cross" terms

  - some sites use interaction likelihoods or phase-sensitive functions [22, 23]
  - given $\boldsymbol{x}_t^{\mathrm{s}}, \boldsymbol{h}^{\mathrm{s}}$ and $\boldsymbol{n}_t^{\mathrm{s}}$ there is a distribution

$$\boldsymbol{y}_t^{\mathrm{s}} \sim \mathcal{N}\left(\boldsymbol{x}_t^{\mathrm{s}} + \boldsymbol{h}_t^{\mathrm{s}} + f(\boldsymbol{x}_t^{\mathrm{s}}, \boldsymbol{n}_t^{\mathrm{s}}, \boldsymbol{h}^{\mathrm{s}}), \boldsymbol{\Phi}\right)$$

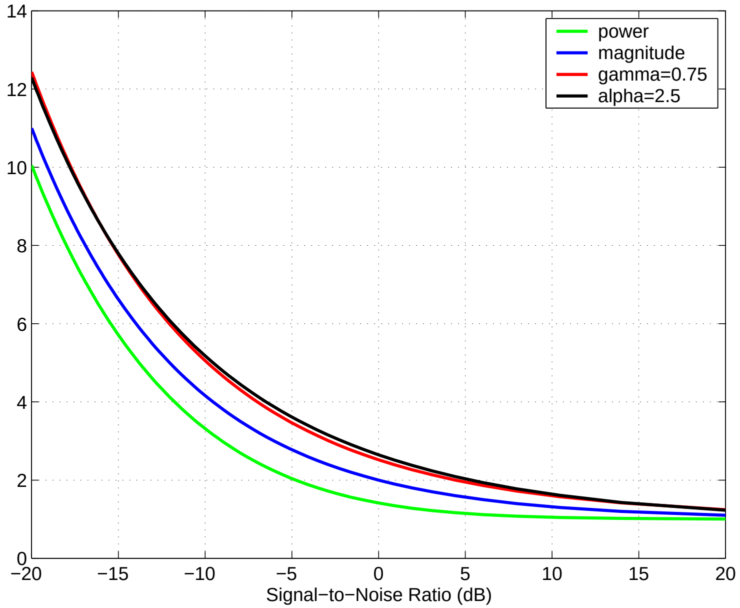
Mismatch function optimisation

- The mismatch function is only an approximation - variations possible
  - γ-optimised - tunable parameter γ, ignoring $\boldsymbol{h}^{\text{s}}$

$$\boldsymbol{y}_t^{\text{s}} = \boldsymbol{x}_t^{\text{s}} + \frac{1}{\gamma}\mathbf{C}\log\left(1+\exp\left(\gamma\mathbf{C}^{-1}(\boldsymbol{n}_t^{\text{s}} - \boldsymbol{x}_t^{\text{s}})\right)\right)$$

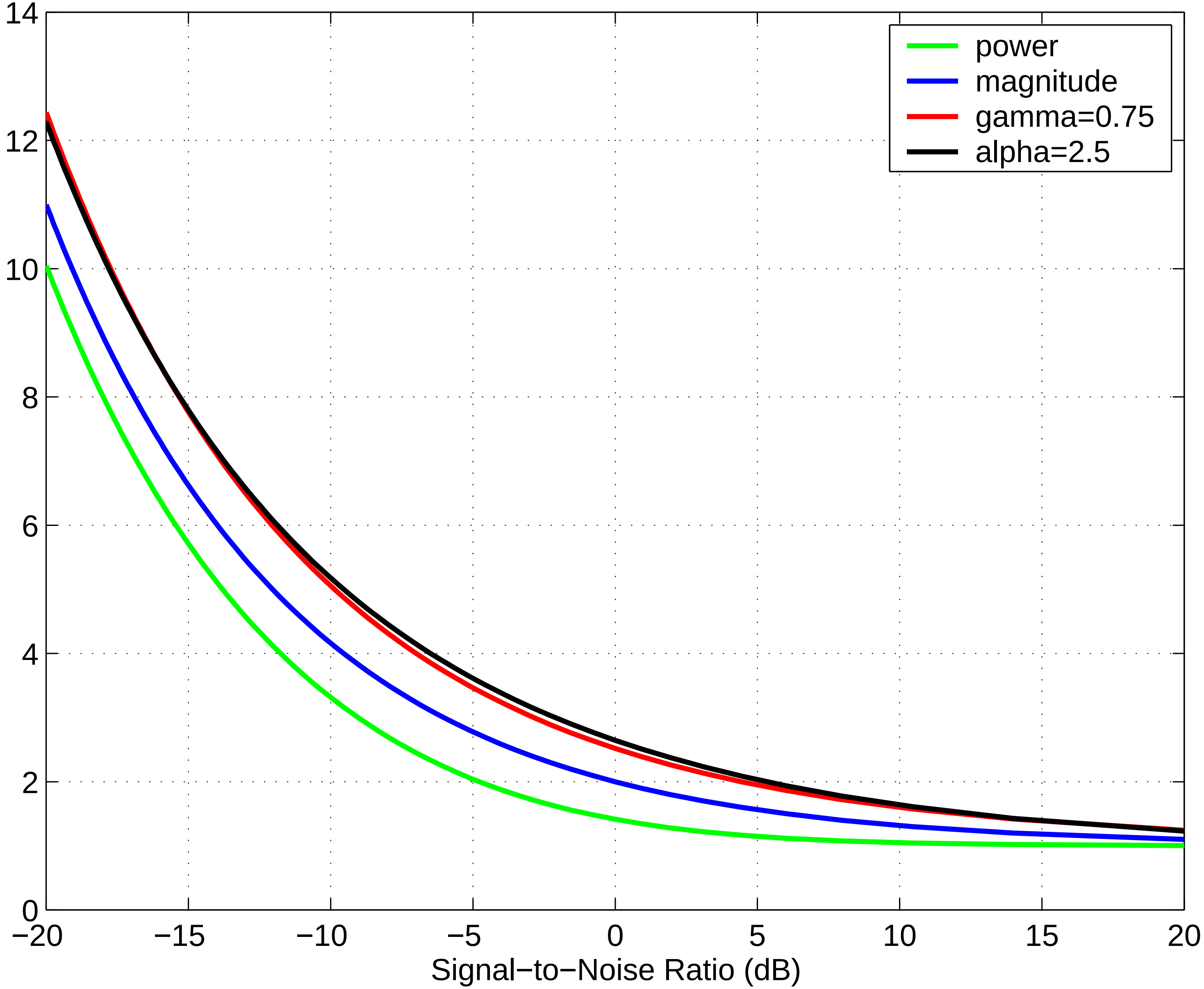
  - Phase-sensitive - tunable parameter α, in theory $-1 \leq \alpha \leq 1$

$$\boldsymbol{y}_t^{\text{s}} = \boldsymbol{x}_t^{\text{s}} + \frac{1}{2}\mathbf{C}\log\left(1+\exp\left(2\mathbf{C}^{-1}(\boldsymbol{n}_t^{\text{s}} - \boldsymbol{x}_t^{\text{s}})\right) + 2\alpha\exp\left(\mathbf{C}^{-1}(\boldsymbol{n}_t^{\text{s}} - \boldsymbol{x}_t^{\text{s}})\right)\right)$$



Ratio of corrupted speech magnitude to clean speech magnitude

- magnitude ($\alpha = 1$, $\gamma = 1$)
- power ($\alpha = 0$, $\gamma = 2$)
- $\alpha = 2.5$ (AURORA tuned [24])
- $\gamma = 0.75$ (AURORA tuned [25])
- $\gamma = 1.0$ used in this work



power
magnitude
gamma=0.75
alpha=2.5
0
2
4
6
8
10
12
14
−20
−15
−10
−5
0
5
10
15
20
Signal−to−Noise Ratio (dB)

Delta and Delta-Delta Parameters

- Aim to 'reduce' HMM conditional independence assumptions
  - standard to add delta and delta-delta [26] parameters

$$\boldsymbol{y}_t = \begin{bmatrix} \boldsymbol{y}_t^{\text{s}} \\ \boldsymbol{\Delta y}_t^{\text{s}} \\ \boldsymbol{\Delta}^2 \boldsymbol{y}_t^{\text{s}} \end{bmatrix}; \quad \boldsymbol{\Delta y}_t^{\text{s}} = \frac{\sum_{i=1}^{n} w_i \left(\boldsymbol{y}_{t+i}^{\text{s}} - \boldsymbol{y}_{t-i}^{\text{s}} \right)}{2 \sum_{i=1}^{n} w_i^2}$$

- Two versions used to represent the impact of noise on these [27, 28]

$$\boldsymbol{\Delta y}_t^{\text{s}} \approx \frac{\partial \boldsymbol{y}_t^{\text{s}}}{\partial t} \quad \text{OR} \quad \boldsymbol{\Delta y}_t^{\text{s}} = \mathbf{D} \begin{bmatrix} \boldsymbol{y}_{t-1}^{\text{s}} \\ \boldsymbol{y}_t^{\text{s}} \\ \boldsymbol{y}_{t+1}^{\text{s}} \end{bmatrix}$$

  - the second is more accurate, but more statistics required to be stored
  - need to compensate all model parameters for best performance
- For enhancement can simply base deltas on static "clean" features

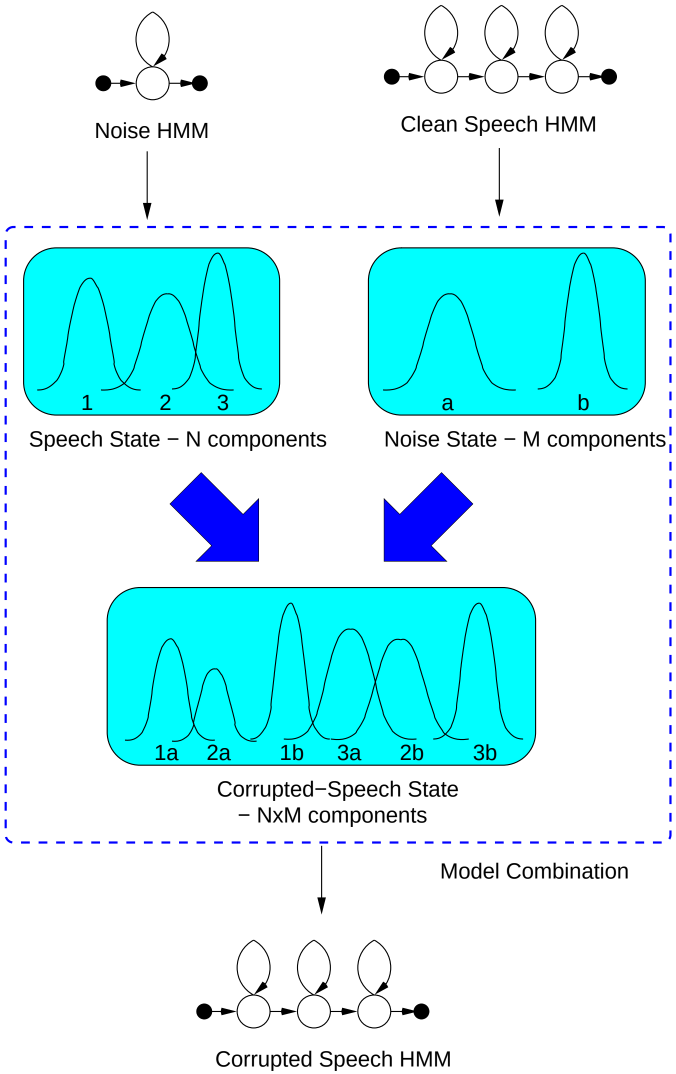
Model-Based Compensation

- Could retrain system using noise-corrupted training data
  - need to have all training data available and corrupt it with noise
  - slow - single-pass retraining [29] a faster approximation
- Model-based compensation approximates SPR [29]

$$\boldsymbol{\mu}_{\mathrm{y}}^{(m)} = \mathcal{E}\{\boldsymbol{y}|m\}; \qquad \boldsymbol{\Sigma}_{\mathrm{y}}^{(m)} = \mathrm{diag}\left(\mathcal{E}\{\boldsymbol{y}\boldsymbol{y}^{\mathsf{T}}|m\} - \boldsymbol{\mu}_{\mathrm{y}}^{(m)}\boldsymbol{\mu}_{\mathrm{y}}^{(m)\mathsf{T}}\right)$$

- Due to non-linearities no closed form solution - approximations required [29]
  - Monte-Carlo-style: generate "speech" and "noise" observations and combine
  - Log-Add: only transform the mean
  - Log-Normal: sum of two log-normal variables approximately log-normal
  - Vector Taylor series: first or higher order expansions used [30]
- Referred to here as predictive schemes - model parameters implicitly found
  - contrast to adaptive speaker transforms - explicit parameter estimation

Model-Based Compensation Procedure



- Each speech/noise pair considered
  - yields final component
- Also multiple-states possible
  - 3-D Viterbi decoding [31]
- Iterative schemes also possible:
  - iterative PMC [29]
  - Algonquin [22]
- Commonly used configuration:
  - single state
  - single component

Vector Taylor Series

- Vector Taylor Series (VTS) one popular approximation [32, 30]
  - Taylor series expansion about "current" parameter values
  - for these expression ignore impact of convolutional distortion
  - mismatch function approximated using first order series

$$\boldsymbol{y}_t^{\mathrm{s}} \approx \boldsymbol{\mu}_{\mathrm{x}}^{\mathrm{s}} + f(\boldsymbol{\mu}_{\mathrm{x}}^{\mathrm{s}}, \boldsymbol{\mu}_{\mathrm{n}}^{\mathrm{s}}) + \boldsymbol{\nabla}_{\mathrm{x}} f(\boldsymbol{x}, \boldsymbol{n})|_{\mu_{\mathrm{x}}^{\mathrm{s}}, \mu_{\mathrm{n}}^{\mathrm{s}}}(\boldsymbol{x}_t^{\mathrm{s}} - \boldsymbol{\mu}_{\mathrm{x}}^{\mathrm{s}}) + \boldsymbol{\nabla}_{\mathrm{n}} f(\boldsymbol{x}, \boldsymbol{n})|_{\mu_{\mathrm{x}}^{\mathrm{s}}, \mu_{\mathrm{n}}^{\mathrm{s}}}(\boldsymbol{n}_t^{\mathrm{s}} - \boldsymbol{\mu}_{\mathrm{n}}^{\mathrm{s}})$$

where $f(\boldsymbol{x}, \boldsymbol{n})$ is the mismatch function from previous slide (ignoring $\boldsymbol{h}^{\mathrm{s}}$)

- Gives simple approach to estimating noise parameters

$$\boldsymbol{\mu}_{\mathrm{y}}^{(m)\mathrm{s}} = \mathcal{E}\{\boldsymbol{y}_t^{\mathrm{s}}|m\} \approx \boldsymbol{\mu}_{\mathrm{x}}^{(m)\mathrm{s}} + f(\boldsymbol{\mu}_{\mathrm{x}}^{(m)\mathrm{s}}, \boldsymbol{\mu}_{\mathrm{n}}^{\mathrm{s}})$$

$$\boldsymbol{\Sigma}_{\mathrm{y}}^{(m)\mathrm{s}} \approx \mathbf{A}\boldsymbol{\Sigma}_{\mathrm{x}}^{(m)\mathrm{s}}\mathbf{A}^{\mathsf{T}} + (\mathbf{I} - \mathbf{A})\boldsymbol{\Sigma}_{\mathrm{n}}^{(m)\mathrm{s}}(\mathbf{I} - \mathbf{A})^{\mathsf{T}}; \quad \mathbf{A} = \frac{\partial \boldsymbol{y}^{\mathrm{s}}}{\partial \boldsymbol{x}^{\mathrm{s}}}$$

Noise Parameter Estimation

- In practice the noise model parameters, $\boldsymbol{\mu}_{\rm n}, \boldsymbol{\mu}_{\rm h}, \boldsymbol{\Sigma}_{\rm n}$, are not known
  - need to be estimated from test data
  - simplest approach - use VAD and start/end frames to estimate noise
- Also possible to use ML estimation [32, 33, 24]

$$\left\{\hat{\boldsymbol{\mu}}_{\rm n}, \hat{\boldsymbol{\mu}}_{\rm h}, \hat{\boldsymbol{\Sigma}}_{\rm n}\right\} = \underset{\boldsymbol{\mu}_{\rm n},\boldsymbol{\mu}_{\rm h},\boldsymbol{\Sigma}_{\rm n}}{\rm argmax}\left\{p(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_T|\boldsymbol{\mu}_{\rm n}, \boldsymbol{\mu}_{\rm h}, \boldsymbol{\Sigma}_{\rm n}; \boldsymbol{\lambda}_{\rm x})\right\}$$

- VTS approximation yields simple approach to find $\boldsymbol{\mu}_{\rm n}, \boldsymbol{\mu}_{\rm h}$
  - first/second-order approaches to find $\boldsymbol{\Sigma}_{\rm n}$
  - simple statistics for auxiliary function
- Parameters estimated in the same fashion as unsupervised adaptation
  - need to have hypothesis $\mathcal{H}$

Extensions to Model-Based Approaches

- Joint Uncertainty Decoding:
  - derived from joint clean/corrupted speech modelling
  - attempts to speed up model compensation process
- Predictive Linear Transforms:
  - efficiently handles changes in the feature-vector correlations
- Adaptive Training using VTS/JUD:
  - training systems with a wide-range of back-ground noise conditions
- Incremental Adaptation:
  - using model-based related schemes in a causal fashion

Minimum Mean-Square Error Estimates

- Estimate the clean speech $\hat{\boldsymbol{x}}_t$ given the corrupted speech $\boldsymbol{y}_t$
  - to handle non-linearity partition space using an R-component GMM, then

$$\hat{\boldsymbol{x}}_t = \mathcal{E}\{\boldsymbol{x}_t|\boldsymbol{y}_t\} = \sum_{r=1}^{R} P(r|\boldsymbol{y}_t)\mathcal{E}\{\boldsymbol{x}_t|\boldsymbol{y}_t, r\}$$

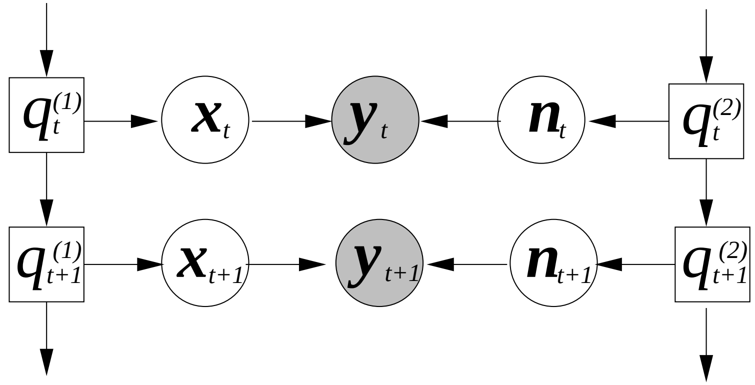
- Model the joint-distribution for each component, then [34]

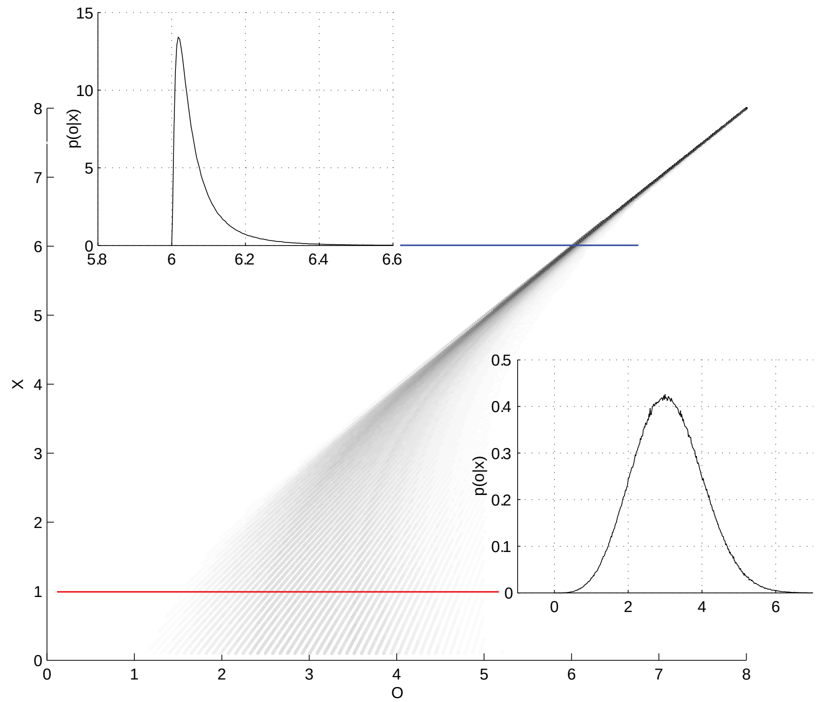
$$\left.\left[\begin{array}{c} \boldsymbol{y}_t \\ \boldsymbol{x}_t \end{array}\right]\right| r \sim \mathcal{N}\left(\left[\begin{array}{c} \boldsymbol{\mu}_{\mathrm{y}}^{(r)} \\ \boldsymbol{\mu}_{\mathrm{x}}^{(r)} \end{array}\right], \left[\begin{array}{cc} \boldsymbol{\Sigma}_{\mathrm{yy}}^{(r)} &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

$$\mathcal{E}\{\boldsymbol{x}_t|\boldsymbol{y}_t, r\} = \boldsymbol{\mu}_{\mathrm{x}}^{(r)} + \boldsymbol{\Sigma}_{\mathrm{xy}}^{(r)}\boldsymbol{\Sigma}_{\mathrm{yy}}^{(r)\text{-}1}(\boldsymbol{y}_t - \boldsymbol{\mu}_{\mathrm{y}}^{(r)}) = \mathbf{A}^{(r)}\boldsymbol{y}_t + \mathbf{b}^{(r)}$$

  - joint distribution estimated using stereo data
    can be estimated using model-based compensation schemes [32, 35]
  - various forms/variants possible: SPLICE [36], POF[37], VTS-based [32, 38]

Uncertainty Decoding





$$p(\boldsymbol{y}_t) = \int p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{n}_t)p(\boldsymbol{x}_t)p(\boldsymbol{n}_t)d\boldsymbol{n}_t d\boldsymbol{x}_t$$

- All the model-based approaches are computationally expensive
  - scales linearly with # components (100K+ for LVCSR systems)
- Need to model the conditional distribution $p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{n}_t)$ [39, 22, 33]
  - select form to allow efficient compensation/decoding (if possible)

Joint Uncertainty Decoding

- Rather than model $p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{n}_t)$ use [33]

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t) = \int p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{n}_t)p(\boldsymbol{n}_t)d\boldsymbol{n}_t$$

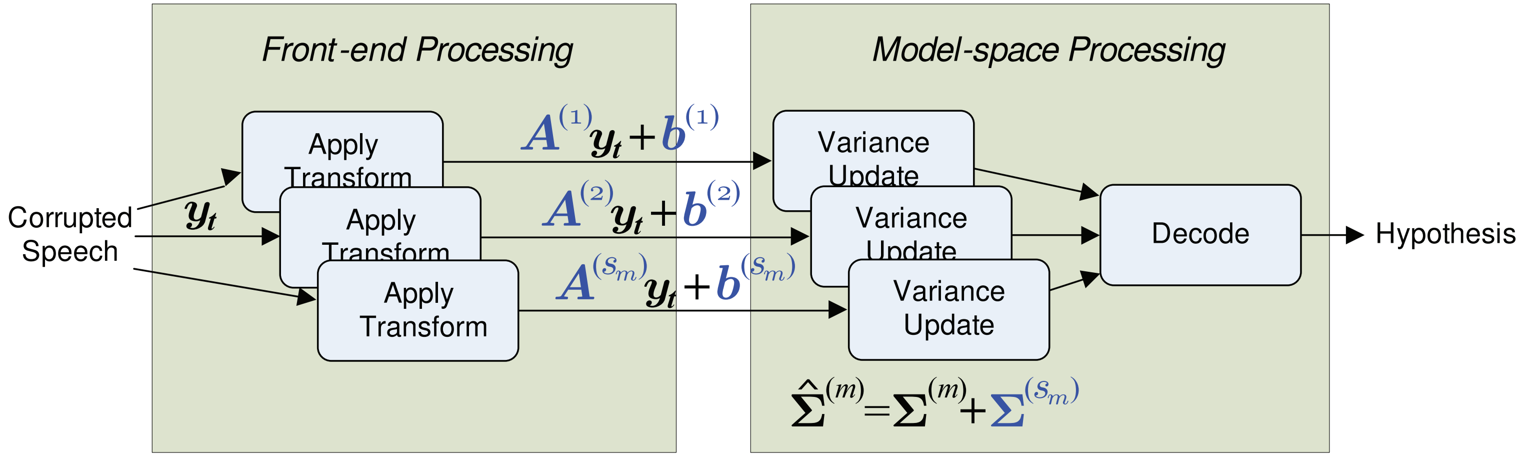
- Simplest approach is to assume $\boldsymbol{y}_t$ and $\boldsymbol{x}_t$ jointly Gaussian (again)
  - to handle changes with acoustic-space make dependent on r
  - simple to derive conditional distribution $p(\boldsymbol{y}_t|\boldsymbol{x}_t, r)$
  - contrast to MMSE where $p(\boldsymbol{x}_t|\boldsymbol{y}_t, r)$ modelled
  - joint distribution estimated using VTS/PMC (stereo data can also be used)
- Product of Gaussians is an un-normalised Gaussian, so

$$p(\boldsymbol{y}_t|m, r) = |\mathbf{A}^{(r)}|\mathcal{N}(\mathbf{A}^{(r)}\boldsymbol{y}_t + \mathbf{b}^{(r)}; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)} + \boldsymbol{\Sigma}_{\mathsf{b}}^{(r)})$$

  - r is normally determined by the component m [40]
  - contrast to MMSE where GMM built in acoustic space to determine r

JUD versus (N)CMLLR

- For JUD compensation, PMC/VTS only required at regression class level
  - $\mathbf{A}^{(r)}, \mathbf{b}^{(r)}$ and $\boldsymbol{\Sigma}_{\mathrm{b}}^{(r)}$ functions of noise parameters $\boldsymbol{\mu}_{\mathrm{n}}, \boldsymbol{\mu}_{\mathrm{h}}, \boldsymbol{\Sigma}_{\mathrm{n}}$



- Similar to CMLLR however
  - JUD parameters estimated using noise models derived from data
  - CMLLR directly uses data to estimate parameters
  - JUD has a bias variance, found to be important for noise estimation
  - same form as NCMLLR, but estimated in a predictive fashion

Predictive Linear Transforms

- Consider a GMM, the corrupted/adapted distributions are

$$p(\boldsymbol{y}) = \sum_{m=1}^{M} c_{\text{y}}^{(m)} \mathcal{N}(\boldsymbol{y}; \boldsymbol{\mu}_{\text{y}}^{(m)}, \boldsymbol{\Sigma}_{\text{y}}^{(m)}); \quad \tilde{p}(\boldsymbol{y}) = \sum_{m=1}^{M} c_{\text{x}}^{(m)} |\mathbf{A}| \mathcal{N}(\mathbf{A}\boldsymbol{y} + \mathbf{b}; \boldsymbol{\mu}_{\text{x}}^{(m)}, \boldsymbol{\Sigma}_{\text{x}}^{(m)})$$

  - how to estimate the "best" linear transform?

- Estimate should be based on minimising the KL-divergence

$$\mathcal{KL}(p||\tilde{p}) = \int p(\boldsymbol{y}) \log \left(\frac{p(\boldsymbol{y})}{\tilde{p}(\boldsymbol{y})} \right) d\boldsymbol{y}$$

  - using the matched-bound approximation (K terms independent of $\mathbf{A}, \mathbf{b}$)

$$\mathcal{KL}(p||\tilde{p}) \leq -\sum_{m=1}^{M} c_{\text{y}}^{(m)} \int \mathcal{N}(\boldsymbol{y}; \boldsymbol{\mu}_{\text{y}}^{(m)}, \boldsymbol{\Sigma}_{\text{y}}^{(m)}) \log \left(\mathcal{N}(\mathbf{A}\boldsymbol{y} + \mathbf{b}; \boldsymbol{\mu}_{\text{x}}^{(m)}, \boldsymbol{\Sigma}_{\text{x}}^{(m)}) \right) d\boldsymbol{y} + K$$

  - a framework for estimating "predictive" linear transforms [41]

Predictive CMLLR

- For schemes like CMLLR required the "predictive" statistics are e.g.:

$$\mathbf{k}^{(r)}_{\mathtt{pc}i} = \sum_{m \in \mathtt{r}_r} \frac{\gamma^{(m)} \mu^{(m)}_{\mathtt{x}i}}{\sigma^{(m)2}_{\mathtt{x}i}} \begin{bmatrix} 1 \\ \mathcal{E}\{\boldsymbol{y}|m\} \end{bmatrix}$$

  - normally the expectations obtained from observed data (adaptive)
  - could also use model-compensation schemes to obtain values (predictive)
  - $\gamma^{(m)}$ either based on observations $\gamma^{(m)}_{\mathtt{y}t}$ or training data counts $\gamma^{(m)}_{\mathtt{x}}$

$$\mathcal{E}\{\boldsymbol{y}|m\} = \frac{\sum_t \gamma^{(m)}_{\mathtt{y}t} \boldsymbol{y}_t}{\sum_t \gamma^{(m)}_{\mathtt{y}t}} \quad \textbf{or} \quad \mathcal{E}\{\boldsymbol{y}|m\} = \boldsymbol{\mu}^{(m)}_{\mathtt{y}}$$

- Schemes such as JUD can be used to efficiently obtain "pseudo" statistics

$$\sum_{m \in \mathtt{r}_r} \frac{\gamma^{(m)}_{\mathtt{x}}}{\sigma^{(m)2}_{\mathtt{x}i}} \mathcal{E}\{\boldsymbol{y}|m\} = \mathbf{A}^{(r)\text{-}1} \left(\sum_{m \in \mathtt{r}_r} \frac{\gamma^{(m)}_{\mathtt{x}} \boldsymbol{\mu}^{(m)}_{\mathtt{x}}}{\sigma^{(m)2}_{\mathtt{x}i}} \right) - \mathbf{A}^{(r)\text{-}1} \mathbf{b}^{(r)} \left(\sum_{m \in \mathtt{r}_r} \frac{\gamma^{(m)}_{\mathtt{x}}}{\sigma^{(m)2}_{\mathtt{x}i}} \right)$$

"Adaptive" vs "Predictive" Schemes

- Adaptive and predictive schemes complementary to one another

Adaptive	Predictive
general approach	applicable to noise
linear assumption - use many linear transforms	mismatch function required - may be inaccurate
transform parameters estimated - large numbers of parameters	noise model estimated - small number of parameters

- Obvious approach is to combine the two in a fashion similar to MAP [42]:
  - limited data predictive approaches used
  - increased data adaptive approaches used
- Count smoothing simple approach to use (parent transforms also possible)

$$\mathbf{k}^{(r)}_{\mathtt{pa}i} = \frac{\mathbf{k}^{(r)}_{\mathtt{pc}i}}{\sum_{m\in\mathtt{r}_r}\gamma^{(m)}_{\mathtt{x}}} + \tau_{\mathsf{sm}}\mathbf{k}^{(r)}_i$$

PCMLLR vs MMSE Schemes

- Both MMSE and PCMLLR yield liner transforms of the feature-space

$$\hat{\boldsymbol{x}}_t = \mathbf{A}\boldsymbol{y}_t + \mathbf{b}$$

  - both make use of the joint distribution between clean and corrupted speech

- However motivation for the two approaches very different
  - MMSE is the expected value of the clean speech
  - PCMLLR is the linear transform that minimises the KL-divergence

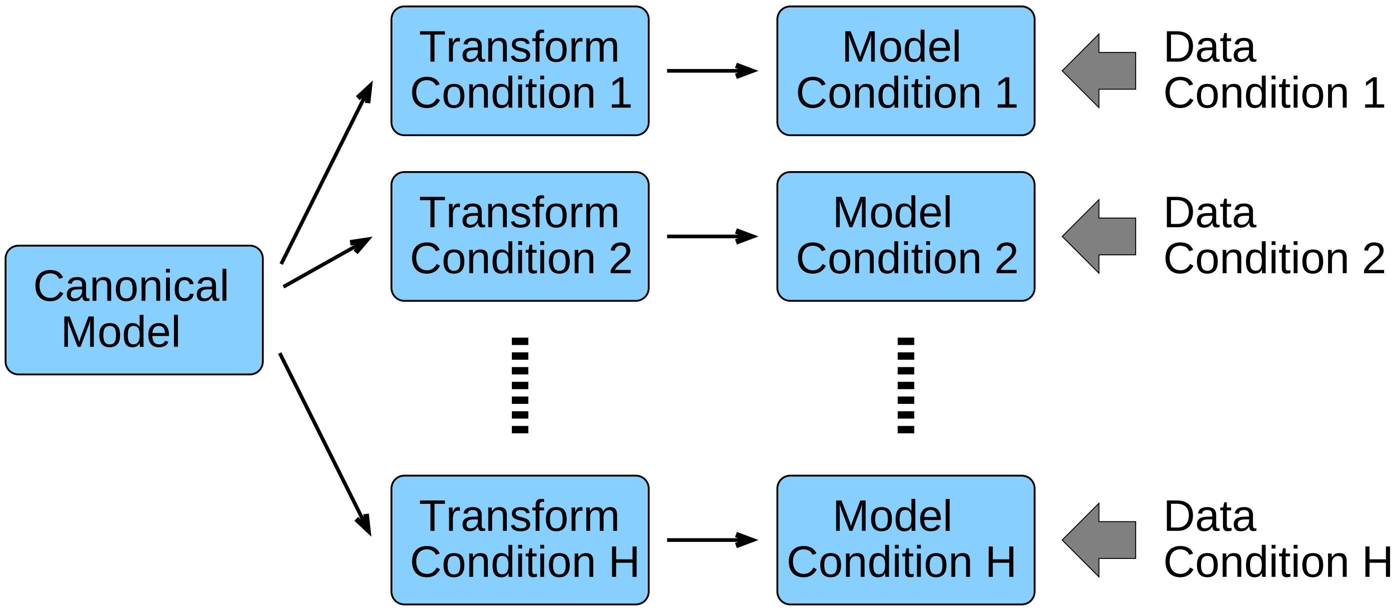
- Theoretically should use:
  - MMSE: when enhancing data for additional processing
  - PCMLLR: when transformed data directly for recognition

- Initial AURORA results show that PCMLLR out-performs MMSE

Adaptive Training

- In practice training data comes from multiple sources
  - various levels and sources of background noise
  - various speakers and channel conditions
- Multistyle (multi-environment) models required to represent all variabilities
  - for wide-range of noises models become very "broad"
  - previously seen issues with applying VTS/JUD to multi-style models
- Adaptive training one approach to handling this
  - adaptive training with various transforms previously investigated
  - generic transforms: MLLR, CMLLR, CAT
  - noise targeted transform: Noisy CMLLR
- Perform adaptive training with VTS and JUD
  - Joint Adaptive Training examined on Broadcast News transcription [43]
  - interested in applying VTS-adaptive training/JAT in lower SNR conditions

VTS/JOINT based Adaptive Training



- Same general framework as other adaptive training schemes
  - partition data into H homogeneous subsets
  - interleave updates of transform and canonical model
- Canonical model update (more) interesting with VTS/JUD transforms

VTS/JOINT Adaptive Training

- System trained by interleaving transform (VTS/JUD) and HMM estimates
  - VTS/JUD estimates usual approach using current canonical model
  - canonical model estimation based on second-order optimisation [43]
  - also possible to use EM-based [21]
- Derivative wrt to mean of canonical model given by

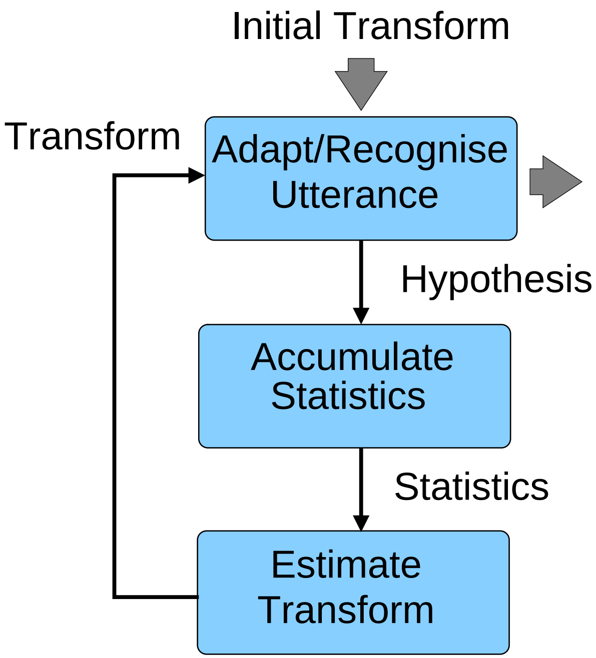
$$\frac{\partial \mathcal{Q}_{\mathrm{j}}}{\partial \mu_{\mathrm{x}i}^{(m)}} = \sum_{h=1}^{H} \sum_{t=1}^{T^{(h)}} \gamma_{\mathrm{yt}}^{(m)} \left(\frac{\hat{\boldsymbol{a}}_i^{(rh)} \boldsymbol{y}_t + \hat{b}_i^{(rh)} - \mu_{\mathrm{x}i}^{(m)}}{\sigma_{\mathrm{x}i}^{(m)2} + \hat{\sigma}_{\mathrm{b}i}^{(rh)2}} \right)$$

  - $\hat{\sigma}_{\mathrm{b}i}^{(rh)2}$ is larger in low-SNR regions
  - impact of observation on derivative decreases for low-SNR
  - agrees with intuition
- VAT implementation is JAT with #regression classes = #system components

Incremental Noise Estimation

- Batch-mode adaptation introduces latency into decoding process
  - for some tasks, e.g. in-car command/control, need to minimise latency
  - many tasks require multiple interactions over a short period

- Incremental adaptation introduces no latency-



- generate hypothesis using current transform
- accumulate statistics $\mathcal{O}_i^{(m)}$ using hypothesis

$$\mathcal{O}_i^{(m)} = \sum_t \gamma_{\text{y}t}^{(m)} \boldsymbol{y}_t^{(i)} + \alpha\ \mathcal{O}_{i-1}^{(m)}$$

  - $0 \Rightarrow$ no smoothing
  - $1 \Rightarrow$ "complete" smoothing

- estimate transform for next utterance

Combined Incremental/Adaptive Processing

- Adaptive training requires a test-condition transform for good performance
  - normally requires a multi-pass system
  - multiple models may be required (where to get initial htypothesis)
  - sensitivity to initial hypothesis/transform
- Noise parameters can be estimated using a single utterance
- Incremental adaptation a good framework for VTS/JUD adaptive systems
  - no need for multiple-passes
  - output can be generated in a causal fashion
  - hypothesis/initial transforn may be "good" (depending on form of noise)
  - only an adaptively trained system needed
- BUT still need an initial transform for first utterance to get things started

Toshiba In-Car Task

- TREL-CRL04 small/medium sized recognition task
  - Speech collected in the office and in vehicles (enon, city, highway)
  - `phone numbers` (PH) task used for initial evaluations
    * 30 English speakers (15 male, 15 female) uttering 30 sentences each
    * 35, 18 SNR averages for the enon, highway condition, respectively
  - `4 digits` (4D), `command & control` (CC) and `city names` (CN) also used

#	PH	4D	CC	CN
utt	861	757	1916	958
words/utt	9.5	4.0	5.5	1.2
secs/utt	6.9	3.6	4.7	3.2
vocabulary	11	11	119	544

averaged between enon/hway (no city)

- Range of lengths (total not speech) and vocabularies

System Configuration

- Acoustic training data - 486 hours of data
  - mixture of real in-car data and clean data artificially corrupted
  - approx 283 hours artificially corrupted data (from WSJ data)
  - approx 203 hours "real" data
- Acoustic model characteristics
  - MFCC-parameters plus delta/delta-delta features (39-dimensional)
  - $\approx$650 states, 12 components/state, $\approx$7800 components
  - acoustic models: decision tree clustered states, cross-word triphones
- compact system (embedded market is possible target domain)
- Both multi-style (multi) and adaptively trained (adapt) system built

Batch Multi-Style vs VTS Adaptive Training (phone-numbers)

Iteration	ENON		HWY	
	multi	adapt	multi	adapt
0	1.1		4.5	
1	1.5	0.5	3.2	1.6
2	1.4	0.5	2.4	1.5

- VTS adaptation applied to both multi-style and adaptively trained systems
  - update hypothesis and transform at each iteration
- Multi-style performance degraded by VTS for high SNR conditions
  - mismatch function not suitable for multi-style training
- VTS Adaptive training consistent gains
  - better than multi-style training, worked at higher SNR conditions

Adaptively-Trained VTS Performance Summary

		ENON				HWY				Avg
		PH	4D	CC	CN	PH	4D	CC	CN	
iter	0	1.1	1.0	0.9	3.9	4.5	3.0	2.0	14.5	3.86
	1	0.5	0.3	0.7	3.6	1.6	1.5	1.4	13.6	2.90
	2	0.5	0.2	0.7	3.7	1.5	1.3	1.3	11.0	2.53
ETSI-adv		1.2	1.1	0.9	4.5	3.1	1.8	1.2	8.2	2.75

- Batch VTS adaptation evaluated on full range of tasks
  - compared to ETSI-advanced front-end with same training data
- For ENON consistent gains for all conditions
- For HWY mixed results are more mixed
  - City-Names (CN) very poor performance
  - related to sensitivity to initial hypothesis/noise estimate
  - can get similar performance to ETSI advanced front-end eventually

Incremental vs Batch with Adaptively Trained System

System	ENON				HWY				Avg
	PH	4D	CC	CN	PH	4D	CC	CN	
VAT-batch	0.5	0.2	0.7	3.7	1.5	1.3	1.3	11.0	2.53
VAT-INC	0.6	0.3	0.8	3.8	2.0	1.9	1.5	6.8	2.21
JAT-INC	1.2	0.7	0.8	3.6	2.5	2.3	1.7	6.9	2.46
ETSI-adv	1.2	1.1	0.9	4.5	3.1	1.8	1.2	8.2	2.75

- Incremental adaptation applied to adaptively trained systems
  - smoothing factor of $\alpha = 0.6$, 2-iteration batch results
  - also used JAT - highly efficient noise estimation/adaptation
  - initial hypothesis from multi-style system ...
- Slight degradation from batch to incremental for ENON
  - issues with City-Names addressed for HWY
- Incremental adaptation yields good overall performance

Predictive and Adaptive Incremental Adaptation

System	HWY			
	PH	4D	CC	CN
VAT-INC	2.0	1.9	1.5	6.8
+CMLLR	1.7	2.0	1.1	6.7
ETSI-adv	3.1	1.8	1.2	8.2

- Predictive with Adaptive incremental adaptation
  - use VTS compensated models to get prior statistics for CMLLR
- Combination of predictive and adaptive provides gains
  - problem with `command & controls` task "fixed"
  - `Phone numbers` also improved using combined compensation approach
- Only preliminary results - need real data to test schemes

Summary Model-Based Noise Robustness

- Model-based compensation approaches
  - good theoretical motivation
  - but requires a mismatch function
  - slow compared to feature-enhancement
- Range of extensions to standard approaches
  - joint uncertainty decoding for faster compensation
  - predictive linear transforms - additional flexibility
  - adaptive training - handle multi-style data
  - incremental adaptation
- Interesting area - still problems
  - efficiency still needs to be improved
  - improved compensation (full covariance matrices)
  - improved use of adaptive training

References

[1] P. C. Woodland, "Speaker adaptation for continuous density HMMs: A review," in *ISCA Adaptation Workshop.*, 2001.

[2] J. L. Gauvain and C.-H. Lee, "Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[3] R. Kuhn, P. Nguyen, J.-C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, "Eigenvoices for speaker adaptation," in *Proceedings ICSLP*, 1998, pp. 1771–1774.

[4] M. J. F. Gales, "Cluster adaptive training of hidden Markov models," *IEEE Transactions Speech and Audio Processing*, vol. 8, pp. 417–428, 2000.

[5] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[6] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Languages*, vol. 10, pp. 249–264, 1996.

[7] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[8] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, X. Liu, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book, version 3.4*. Cambridge, UK: Cambridge University Engineering Department, 2006.

[9] W. Chou, "Maximum a-posterior linear regression with elliptical symmetic matrix variate priors," in *Proceedings Eurospeech*, 1999, pp. 1–4.

[10] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proceedings ICSLP*, 1996, pp. 1137–1140.

[11] P. C. Woodland, D. Pye, and M. J. F. Gales, "Iterative unsupervised adaptation using maximum likelihood linear regression," in *Proc. Int. Conf. Spoken Lang. Process.*, Philadelphia, 1996, pp. 1133–1136.

[12] L. F. Uebel and P. C. Woodland, "Speaker adaptation using lattice-based MLLR," in *Proc. ITRW on Adaptation Methods for Speech Recognition*, 2001.

[13] K. Yu and M. Gales, "Bayesian adaptive inference and adaptive training," *IEEE Transactions Speech and Audio Processing*, vol. 15, no. 6, pp. 1932–1943, August 2007.

[14] F. Wallhof, D. Willett, and G. Rigoll, "Frame-discriminative and Confidence-driven Adaptation for LVCSR," in *Proc ICASSP'00*, Istanbul, 2000, pp. 1835–1838.

[15] S. Tsakalidis, V. Doumpiotis, and W. Byrne, "Discriminative Linear Transforms for Feature Normalisation and Speaker Adaptation in HMM Estimation," *IEEE Trans Speech and Audio Processing*, vol. 13, no. 3, pp. 367–376, 2005.

[16] L. Wang and P. Woodland, "Discriminative Adaptive Training using the MPE Criterion," in *Proc ASRU*, St Thomas, US Virgin Islands, 2003.

[17] K. Yu, M. Gales, and P. Woodland, "Unsupervised discriminative adaptation using discriminative mapping transforms," in *Proc. ICASSP*, 2008.

[18] C. Raut, K. Yu, and M. Gales, "Adaptive training using discriminative mapping transforms," in *Proc. InterSPeech*, 2008.

[19] D. Kim and M. Gales, "Noisy CMLLR for noise-robust speech recognition," University of Cambridge, Tech. Rep. TR611, February 2009, available from http://mi.eng.cam.ac.uk/∼mjfg/.

[20] R. Gopinath, B. Ramabhadran, and S. Dharanipragada, "Factor analysis invariant to linear transformations of data," in *Proc. Int. Conf. Spoken Lang. Process.*, 1998, pp. 397–400.

[21] Y. Hu and Q. Huo, "Irrelevant variability normalization based HMM training using VTS approximation of an explicit model of environmental distortions," in *Proc. InterSPeech*, 2007.

[22] T. Kristjansson, "Speech recognition in adverse environments: a probabilistic approach," Ph.D. dissertation, University of Waterloo, Waterloo, Canada, 2002.

[23] L. Deng, J. Droppo, and A. Acero, "Enhancement of log mel power spectra of speech using a phase sensitive model the acoustic environemnt and sequential estimation of the corrupting noise," *Proc. IEEE Transactions on Speech and Audio Processing*, 2004.

[24] J. Li, L. Deng, Y. Gong, and A. Acero, "HMM adaptation with joint compensation of additive and convolutive distortions via vector Taylor series," in *ASRU 2007*, Kyoto, Japan, 2007.

[25] M. Gales and F. Flego, "Discriminative classifiers with generative kernels for noise robust speech recognition," Cambridge University, Tech. Rep. CUED/F-INFENG/TR605, August 2008, available from: http://mi.eng.cam.ac.uk/∼mjfg.

[26] S. Furui, "Speaker independent isolated word recognition using dynamic features of speech spectrum," *IEEE Transactions ASSP*, vol. 34, pp. 52–59, 1986.

[27] R. A. Gopinath, M. J. F. Gales, P. S. Gopalakrishnan, S. Balakrishnan-Aiyer, and M. A. Picheny, "Robust speech recognition in noise - performance of the IBM continuous speech recognizer on the ARPA noise spoke task," in *ARPA Workshop on Spoken Language System Technology*, 1995, pp. 127–130.

[28] R. van Dalen and M. Gales, “Extended VTS for noise-robust speech recognition,” in *Proc ICASSP*, 2009.

[29] M. Gales, “Model-based Techniques for Noise Robust Speech Recognition,” Ph.D. dissertation, Cambridge University, 1995.

[30] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, “HMM Adaptation using Vector Taylor Series for Noisy Speech Recognition,” in *Proc ICSLP*, Beijing, China, 2000.

[31] A. Varga and R. Moore, “Hidden Markov model decomposition of speech and noise,” in *Proc ICASSP*, 1990, pp. 845–848.

[32] P. Moreno, “Speech Recognition in Noisy Environments,” Ph.D. dissertation, Carnegie Mellon University, 1996.

[33] H. Liao, “Uncertainty Decoding For Noise Robust Speech Recognition,” Ph.D. dissertation, Cambridge University, 2007.

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

[35] V. Stouten, H. V. Hamme, and P. Wambacq, “Accounting for the uncertainty of speech estimates in the context of model-based feature enhancement,” in *Proc. ICSLP*, vol. 1, Jeju Island, Korea, Oct. 2004, pp. 105–108.

[36] J. Droppo, L. Deng, and A. Acero, “Evaluation of the SPLICE Algorithm on the Aurora 2 Database,” in *Proc. Eurospeech*, Aalberg, Denmark, 2001, pp. 217–220.

[37] L. Neumeyer and M. Weintraub, “Probabilistic Optimum Filtering for Robust Speech Recognition,” in *Proc. ICASSP*, Adelaide, 1994.

[38] V. Stouten, “Robust speech recognition in time-varying environments,” Ph.D. dissertation, Universitet Leuven, 2006.

[39] J. Droppo, A. Acero, and L. Deng, “Uncertainty decoding with SPLICE for Noise Robust Speech Recognition,” in *Proc ICASSP 02*, Orlando, Florida, 2002.

[40] H. Liao and M. Gales, “Issues with Uncertainty Decoding for Noise Robust Speech Recognition,” in *Proc. ICSLP*, Pittsburgh, PA, 2006.

[41] M. J. F. Gales and R. C. van Dalen, “Predictive linear transforms for noise robust speech recognition,” in *Proc. ASRU*, 2007, pp. 59–64.

[42] F. Flego and M. Gales, “Incremental predictive and adaptive noise compensation,” in *Proc ICASSP*, 2009.

[43] H. Liao and M. Gales, “Adaptive training with joint uncertainty decoding for robust recognition of noisy data,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2007.