Noise Robust Automatic Speech Recognition

Jasha Droppo
Microsoft Research

jdroppo@microsoft.com
http://research.microsoft.com/~jdroppo/

Additive Noise

- In controlled experiments, training and testing data can be made quite similar.
- In deployed systems, the test data is often corrupted in new and exciting ways.

Overview

- **Introduction**
  - **Standard noise robustness tasks**
  - **Overview of techniques to be covered**
  - **General design guidelines**
- Analysis of noisy speech features
- Feature-based Techniques
  - Normalization
  - Enhancement
- Model-based Techniques
  - Retraining
  - Adaptation
- Joint Techniques
  - Noise Adaptive Training
  - Joint front-end and back-end training
  - Uncertainty Decoding
  - Missing Feature Theory

Standard Noise-Robust Tasks

- Prove relative usefulness of different techniques
- Can include recipe for acoustic model building.
- Allow you to make sure your code is performing properly.
- Allow others to evaluate your new algorithms.

Standard Noise-Robust Tasks

- Aurora
  - Aurora2: Artificially Noisy English Digits
  - Aurora3: Digits recorded in four European languages, recorded inside cars.
  - Aurora4: Artificially noisy Wall Street Journal.
- SpeechDat Car
  - Noisy digits in European languages within cars.
- SPINE
  - Noises that can be mixed at various levels to clean speech signals to simulate noisy environments.

Feature-Based Techniques

- Only useful if you have the ability to retrain the acoustic model.
- Normalization: simple, yet powerful.
  - Moment matching (CMN, CVN, CHN)
  - Cepstral modulation filtering (RASTA, ARMA)
- Enhancement: more complex, but worth it.
  - Data-driven (POF, SVM, SPLICE)
  - Model-based (Spectral Subtraction, VTS)

Model-Based Techniques

- Retraining
  - When data is available, this is the best option.
- Adaptation
  - Approximate retraining with a low cost.
  - Re-purpose model-free techniques (MLLR, MAP)
  - Specialized techniques use a corruption model (VTS, PMC)

Joint Techniques

- Noise Adaptive Training
  - A hybrid of multi-style training and feature enhancement.
- Uncertainty Decoding
  - Allows the enhancement algorithm to make a “soft decision” when modifying the features.
- Missing Feature Theory
  - The feature extraction can define some spectral bins as unreliable, and exclude them from decoding.

General Design Guidelines

- Spend the effort for good audio capture
  - Close-talking microphones, microphone arrays, and intelligent microphone placement
  - Information lost during audio capture is not recoverable.
- Use training data similar to what is expected from the end-user
  - Matched condition training is best, followed by multistyle and mismatched training.

General Design Guidelines

- Use feature normalization
  - Low cost, high benefit
  - It eliminates signal variability not relevant to the transcription.
  - Not viable unless you control the acoustic model training.

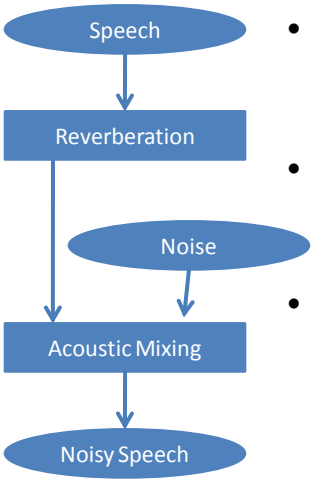
Overview

- Introduction
  - Standard noise robustness tasks
  - Overview of techniques to be covered
  - General design guidelines
- **Analysis of noisy speech features**
- Feature-based Techniques
  - Normalization
  - Enhancement
- Model-based Techniques
  - Retraining
  - Adaptation
- Joint Techniques
  - Noise Adaptive Training
  - Joint front-end and back-end training
  - Uncertainty Decoding
  - Missing Feature Theory

How Does Noise Affect Speech Feature Distributions?

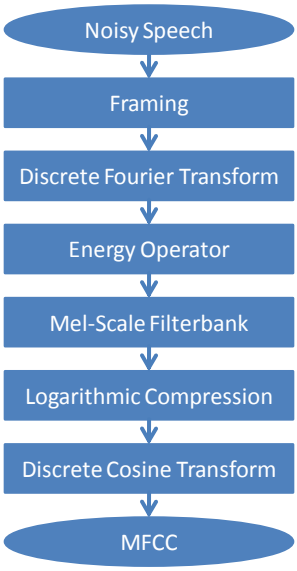
- Feature distributions trained in one noise condition will not match observations in another noise condition.

A Noisy Environment



- A clean, close-talk microphone signal exists (in theory) at the user's mouth.
- Reverberation
  - A difficult problem, not addressed in this tutorial.
- Additive noise
  - At reasonable sound pressure levels, acoustic mixing is linear.
  - So, this should be easy, right?

Feature Extraction



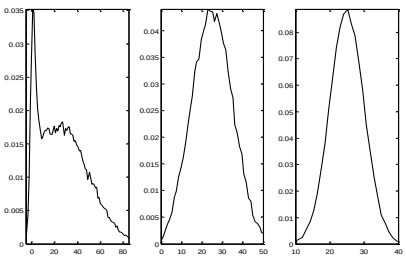
- It is harder than you think.
- At the input to the speech recognition system, the corruption is linear.
- The "energy operator" and "logarithmic compression" are non-linear.
- The "mel-scale filterbank" and "discrete cosine transform" are one-way operations.
- The speech recognition features (MFCC) have non-linear corruption.

Analysis of Noisy Speech Features

- Additive noise systematically corrupts speech features.
  - Creates a *mismatch* between the training and testing data.
  - When the noise is highly variable, it can broaden the acoustic model.
  - When the noise masks dissimilar acoustics so their observations are similar, it can narrow the acoustic model.
- Additive noise affects static features (especially energy) more than dynamic features.
- Speech and noise mix non-linearly in the cepstral coefficients.

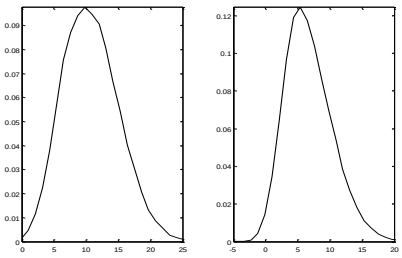
Distribution of noisy speech

- "Clean speech" distribution is Gaussian
  - Mean 25dB and sigma of 25, 10, and 5 dB.
- "Noise" distribution is also Gaussian
  - Mean 0db, sigma 2dB
- "Noisy speech" distribution is not Gaussian
  - Can be bi-modal, skewed, or unchanged.
- Smaller standard deviation yields better Gaussian.
- Additive noise decreases the variances in your acoustic model.



Distribution of noisy speech

- "Clean speech" distribution is Gaussian
  - Smaller covariance than previous example.
  - Sigma of 5dB and means of 10 and 5 dB.
- "Noise" distribution is also Gaussian
  - Same as previous example.
  - Sigma 2dB and mean 0dB
- "Noisy speech" distribution is skewed.



Overview

- Introduction
  - Standard noise robustness tasks
  - Overview of techniques to be covered
  - General design guidelines
- Analysis of noisy speech features
- **Feature-based Techniques**
  - **Normalization**
  - **Enhancement**
- Model-based Techniques
  - Retraining
  - Adaptation
- Joint Techniques
  - Noise Adaptive Training
  - Joint front-end and back-end training
  - Uncertainty Decoding
  - Missing Feature Theory

Feature-Based Techniques

- Voice Activity Detection
  - Eliminate signal segments that are clearly not speech.
  - Reduces “insertion errors”
  - May take advantage of features that the recognizer might not normally use.
    - Zero crossing rate.
    - Pitch energy.
    - Hysteresis.

Feature-Based Techniques

- Significant VAD Improvements on Aurora 3.

	Without VAD			"Oracle" VAD		
	WM	MM	HM	WM	MM	HM
Danish	82.69	58.20	32.00	88.76	73.05	45.08
German	92.07	80.82	74.51	92.09	81.48	77.38
Spanish	88.06	78.12	28.87	93.66	86.81	47.13
Finnish	94.31	60.33	45.51	91.88	79.82	61.48
Ave	89.28	69.37	45.22	91.60	80.29	57.77

Feature-Based Techniques

- Cepstral Normalization
  - Mean Normalization (CMN)
  - Variance Normalization (CVN)
  - Histogram Normalization (CHN)
  - Cepstral Time Smoothing

Cepstral Mean Normalization

- Modify every signal so that its expected value is zero.
- Often coupled with automatic gain normalization (AGN).
- Eliminates variability due to linear channels with short impulse responses.

Further reading:
B.S. Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America*, 55(6):1304-1312, 1974.

Cepstral Mean Normalization

$$
\begin{aligned}
\bar{\mathbf{x}} &= \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{x}_t \\
\hat{\mathbf{x}}_t &= \mathbf{x}_t - \bar{\mathbf{x}}_t \\
\mathbf{y}_t &= \mathbf{x}_t + \mathbf{h} \\
\bar{\mathbf{y}} &= \frac{1}{T}\sum_{t=1}^{T-1}\mathbf{y}_t \\
&= \frac{1}{T}\sum_{t=0}^{T-1}(\mathbf{x}_t + \mathbf{h}) \\
&= \bar{\mathbf{x}} + \mathbf{h} \\
\hat{\mathbf{y}}_t &= \mathbf{y}_t - \bar{\mathbf{y}} = \hat{\mathbf{x}}_t
\end{aligned}
$$

- Compute the feature mean across the utterance
- Subtract this mean from each feature
- New features are invariant to linear filtering

Cepstral Variance Normalization

- Modify every signal so that the second central moment is one.
- No physical interpretation.
- Effectively normalizes the range of the input features.

Cepstral Variance Normalization

$$\mu_{\mathbf{x}} = \frac{1}{T}\sum_{t=0}^{T-1} \mathbf{x}_t$$
$$\sigma_{\mathbf{x}}^2 = \frac{1}{T}\sum_{t=0}^{T-1} (\mathbf{x}_t - \mu_{\mathbf{x}})^2$$
$$\hat{\mathbf{x}}_t = \frac{1}{\sigma_{\mathbf{x}}}(\mathbf{x}_t - \mu_{\mathbf{x}})$$

- Compute the feature mean and covariance across the utterance
- Subtract the mean and divide by the standard deviation
- New features are invariant to linear filtering and scaling

Cepstral Histogram Normalizaion

- Logical extension of CMVN.
  - Equivalent to normalizing each moment of the data to match a target distribution.
- Includes “Gaussianization” as a special case.
- Potentially destroys transcript-relevant information.

Further reading:
A. de la Torre, A.M. Peinado, J.C. Segura, J.L. Perez-Cordoba, M.C. Benitez and A.J. Rubio. Histogram equalization of speech representation for robust speech recognition. *IEEE Transactions on Speech and Audio Processing*, 13(3):355-366, May 2005.

Effects of increasing the level of moment matching.

	Clean AM			Multi-Style AM		
Front End	Base	CMN	CHN	Base	CMN	CHN
Set A	63.66	67.02	79.89	87.74	89.44	91.08
Set B	57.83	69.22	81.26	87.63	89.34	90.75
Clean Test	99.01	99.07	98.94	98.67	98.97	98.13

- Notice first that Multi-Style is better on Set A and Set B, but worse on Clean Test.
- Increased moment matching always improves the accuracy, except on the clean data.
  - Short utterances (1.7s) don't have enough data to do a stable CHN transformation.

Practical CHN

- CHN works well on long utterances, but can fail when there is not enough data.
- Polynomial Histogram Equalization (PHEQ)
  - Approximates inverse CDF function as a polynomial.
- Quantile-Based Histogram Normalization
  - Fast, on-line approximation to full HEQ.
- Cepstral Shape Normalization (CSN)
  - Starts with CMVN, then applies exponential factor.

Further reading:

S.-H. Lin, Y.-M. Yeh and B. Chen. Cluster-based polynomial-fit histogram normalization (CPHEQ) for robust speech recognition. *Proceedings Interspeech 2007*, 1054-1057, August 2007.
F. Hilger and H. Ney. Quantile based histogram equalization for noise robust large vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing,* 14(3):845-854, May 2006.
J. Du and R.-H. Wang. Cepstral shape normalization (CSN) for robust speech recognition. *Proceedings ICASSP 2008,* 4389-4392, April 2008.

Cepstral Time-Smoothing

- The time-evolution of cepstra carries information about both speech and noise.
- High-frequency modulations have more noise than speech.
  - So, low-pass filter the time series (~16Hz)
- Stationary components of the cepstral time-series do not carry relevant information.
  - So, high-pass filter the time series (~1Hz)
  - Similar to CMN?

Further reading:

N. Kanedera, T. Arai, H. Hermansky, and M. Pavel. On the relative importance of various components of the modulation spectrum for automatic speech recognition. *Speech Commnunication,* 28(1):43-55, 1999.

Cepstral Time-Smoothing

- RASTA

$$H(z) = 0.1(z^4)\frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{1 - 0.98z^{-1}}$$

  - A fourth-order ARMA filter
  - Passband from 0.26Hz to 14.3Hz.
  - Empirically designed, shown to improve noise robustness considerably.
- MVA
  - Cascades CMVN and ARMA filtering

Further reading:

H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Trans. On Speech and Audio Processing*, 2(4):578-589, 1994.
C.-P. Chen, K. Filali and J.A. Bilmes. Frontend post-processing and backend model enhancement on the Aurora 2.0/3.0 databases. In *Int. Conf. on Spoken Language Processing*, 2002.

Cepstral Time-Smoothing

- Temporal Shape Normalization
  - Logical extension of fixed modulation filtering.
  - Compute the "average" modulation spectrum from clean speech.
  - Transform each component of the noisy input using a linear filter to match the average modulation spectrum.
  - Better than CMVN alone, RASTA, or MVA.

Further reading:

X. Xiao, E. S. Chng and H. Li. Normalizing the speech modulation spectrum for robust speech recognition. *Proceedings 2007 ICASSP,* (IV):1021-1024, April 2007.
X. Xiao, E. S. Chng and H. Zi. Evaluating the Temporal Structure Normalzation Technique on the Aurora-4 task. *Proceedings Interspeech 2007,* 1070-1073, August 2007.
C.-A. Pan, C.-C. Wang and J.-W. Hung. Improved modulation spectrum normalization techniques for robust speech recognition. *Proceedings 2008 ICASSP,* 4089-4092, April 2008.

Overview

- Introduction
  - Standard noise robustness tasks
  - Overview of techniques to be covered
  - General design guidelines
- Analysis of noisy speech features
- **Feature-based Techniques**
  - **Normalization**
  - **Enhancement**
- Model-based Techniques
  - Retraining
  - Adaptation
- Joint Techniques
  - Noise Adaptive Training
  - Joint front-end and back-end training
  - Uncertainty Decoding
  - Missing Feature Theory

Feature Enhancement

- Most useful technique when one doesn't have access to retraining the recognizer's acoustic model.
- Also provides some gain for the retraining case.
- Attempts to transform the existing observations into what the observations would have been in the absence of corruption.

Is Enhancement for ASR different?

- Design Constraints are Different than the general speech enhancement problem.
- Can tolerate extra delay.
  - Delayed decisions are generally better.
- ASR more sensitive to artifacts.
  - Less-aggressive parameter settings are needed.
- Can operate in log-Mel frequency domain
  - Fewer parameters, more well-behaved estimators.

Feature Enhancement

- Feature enhancement recipes contain three key ingredients:
  - A noise suppression rule
  - Noise parameter estimation
  - Speech parameter estimation

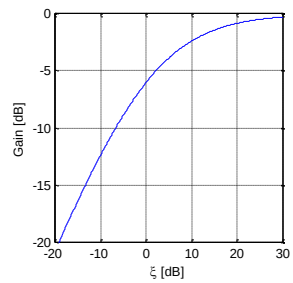
Noise Suppression Rules

- Estimate of the clean speech observation
  - That would have been measured in the absence of noise.
  - Given noise model and speech model parameters.
- Many Flavors
  - Spectral Subtraction
  - Weiner
  - logMMSE-STSA
  - CMMSE

Wiener Filtering

- Assume both speech and noise are independent, WSS Gaussian processes.
- The spectral time-frequency bins are complex Gaussian.
- MMSE estimate of the complex spectral values.

$$\xi = \frac{\sigma_x^2}{\sigma_n^2}$$
$$\hat{x} = \left(\frac{\xi}{\xi+1}\right) y$$



Log-MMSE STSA

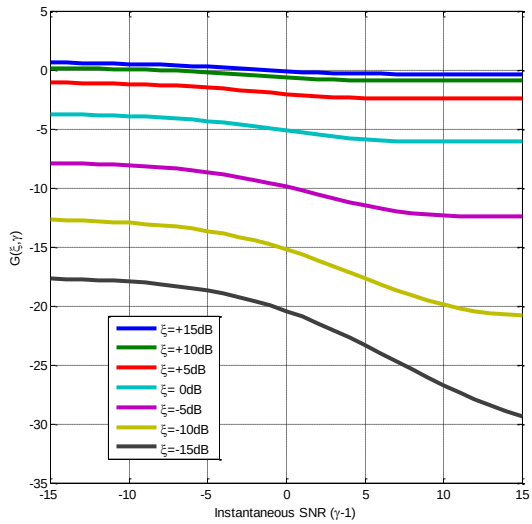
- MMSE of the log-magnitude spectrum.
- Gain is a function of speech parameters, noise parameters, and current observation
- Similar domain to MFCC

$$\begin{aligned}
\xi &= \frac{\sigma_x^2}{\sigma_n^2}, \gamma = \frac{|y|^2}{\sigma_n^2}, \nu = \frac{\xi\gamma}{\xi+1} \\
\hat{x} &= G(\xi,\gamma)y, \text{ where} \\
G(\xi,\gamma) &= \frac{\xi}{\xi+1}\exp\left(\frac{1}{2}\int_\nu^\infty \frac{e^{-z}}{z}dz\right)
\end{aligned}$$

Further Reading:

Y. Ephraim and D. Malah. Speech enhancement using a minimum mean square error Log spectral amplitude estimator, in IEEE Trans. ASSP, vol. 33, pp. 443–445, April 1985.

Log-MMSE STSA



CMMSE

- Ephraim and Malah's log-MMSE is formulated in the FFT-bin domain.
- Cepstral coefficients are different!
- New derivation by Yu is optimal in the cepstral domain.

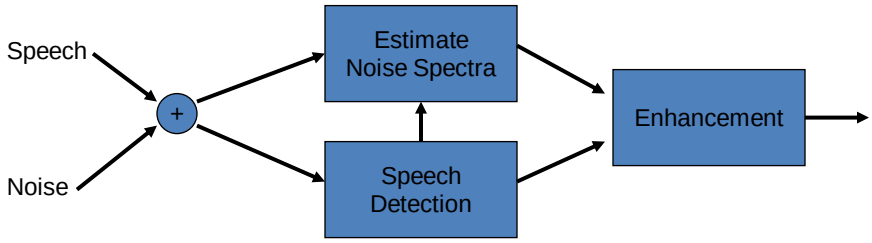
Further Reading:

D. Yu, L. Deng, J. Droppo, J. Wu, Y. Gong and A. Acero. Robust speech recognition using a cepstral minimum-mean-square-error-motivated noise suppressor. *IEEE Transactions on Audio, Speech, and Language Processing,* Vol. 16, No. 5, pp. 1061-1070, July 2008.

Noise Estimation

- To estimate clean speech, feature enhancement techniques need an estimate of the noise spectrum.
- Useful methods
  - Noise tracker
  - Speech detection
    - Track noise statistics in non-speech regions
    - Interpolate statistics into speech regions
  - Integrated with model-based techniques
  - Harmonic tunneling

The Importance of Noise Estimation



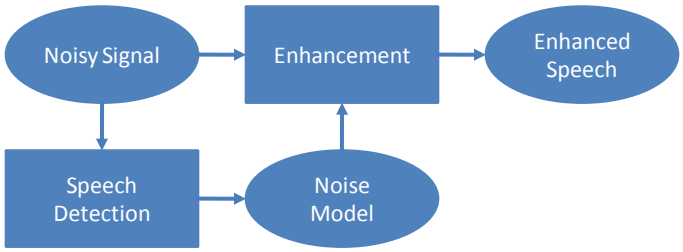
- Simple enhancement algorithms are sensitive to the noise estimation.
- Cheap improvements to noise estimation benefit any enhancement algorithm.

MCRA Noise Estimate

- Least-energetic samples are likely to be (biased) samples of the background noise.
- Components
  - Bias model
  - Per-bin speech activity detector
- Behavior
  - Quickly tracks noise when speech is absent
  - Smoothes the estimate when speech is present

Further reading:
I. Cohen. Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging, in *IEEE Trans. SAP*, vol. 11, no. 5, pp. 466–475, September 2003.

Noise Estimation: Speech Detection



- As frames are classified as non-speech, the noise model is updated.

$$\hat{n}[t] = \begin{cases} \hat{n}[t-1], & y[t] \in \text{speech} \\ \alpha\hat{n}[t-1] + (1-\alpha)y[t], & \text{otherwise} \end{cases}$$

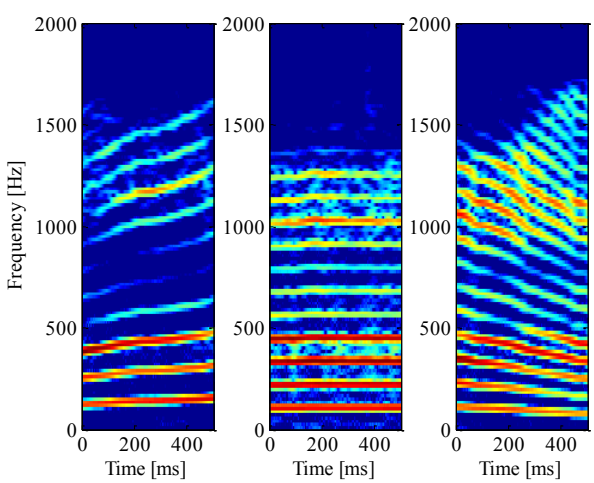
Noise Estimation: Model Based

- Integrated with model-based feature enhancement
  - Uses "VTS Enhancement" theory later in this lecture.
  - Treat noise parameters as values that can be learned from the data.
- Dedicated noise tracking.
  - Uses a model for speech to closely track non-stationary additive noise.

Further reading:

L. Deng, J. Droppo, and A. Acero. Enhancement of log Mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise, in IEEE Transactions on Speech and Audio Processing. Volume: 12 Issue: 2 , Mar 2004. pp. 133-143.
L. Deng, J. Droppo, and A. Acero. Recursive estimation of nonstationary noise using iterative stochastic approximation for robust speech recognition, in IEEE Transactions on Speech and Audio Processing. Volume: 11 Issue: 6 , Nov 2003. pp. 568-580.
G.-H. Ding, X. Wang, Y. Cao, F. Ding and Y. Tang. Sequential Noise Estimation for Noise-Robust Speech Recognition Based on 1st-Order VTS Approximation. 2005 IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 337-342, Nov. 27, 2005.

Noise Estimation: Harmonic Tunneling



Noise Estimation: Harmonic Tunneling

- Attempts to solve the problems of
  - Tracking noises during voiced speech
  - Separating speech from noise
- Most speech energy is during voiced segments, in harmonics of the pitch period.
  - If the noise floor is above the valleys between the peaks, the noise spectrum can be estimated.

Further reading:

J. Droppo, L. Buera, and A. Acero. Speech Enhancement using a Pitch Predictive Model, in *Proc. ICASSP*, Las Vegas, USA, 2008.
M. Seltzer, J. Droppo, and A. Acero. A Harmonic-Model-Based Front End for Robust Speech Recognition, in *Proc. of the Eurospeech Conference*. Geneva, Switzerland, Sep, 2003.
D. Ealey, H. Kelleher and D. Pearce. Harmonic tunneling: tracking non-stationary noises during speech. *Proc. Of the Eurospeech Conference*. Sep, 2001.

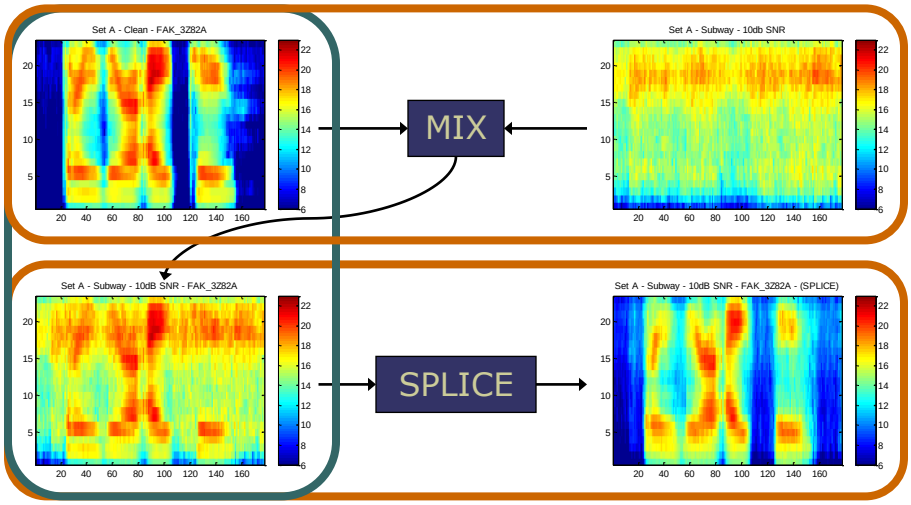
SPLICE

- The SPLICE transform defines a piecewise linear relationship between two vector spaces.
- The parameters of the transform are trained to learn the relationship between clean and noisy speech.
- The relationship is used to infer clean speech from noisy observations.

Further reading:

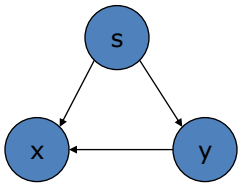
J. Droppo, A. Acero and L. Deng. Evaluation of the SPLICE algorithm on the Aurora 2 database. In *Proc. Of the Eurospeech Conference*, September 2001.

SPLICE Framework



SPLICE

- Learns a joint probability distribution for clean and noisy speech.
- Introduces a hidden discrete random variable to partition the acoustic space.
- Assumes the relationship between clean and noisy speech is linear within each partition.
- Standard inference techniques produce
  - MMSE or MAP estimates of the clean speech.
  - Posterior distributions on clean speech given the observation.



$$
\begin{aligned}
p(s) &= \text{constant} \\
p(\mathbf{y}|s) &= N(\mathbf{y}; \mu_s, \sigma_s^2) \\
p(\mathbf{x}|\mathbf{y}, s) &= N(\mathbf{x}; \mathbf{y} + \mathbf{r}_s, \Gamma_s) \\
p(\mathbf{x}, \mathbf{y}, s) &= p(\mathbf{x}|\mathbf{y}, s)p(\mathbf{y}|s)p(s)
\end{aligned}
$$

$$\hat{x} = E[\mathbf{x}|\mathbf{y}] = \mathbf{y} + \sum_s p(s|\mathbf{y})\mathbf{r}_s$$

SPLICE as Universal Transform

The complete SPLICE transform is a piecewise linear approximation to an arbitrary feature space mapping.

$$x = \sum_m p(m|y)(A_m y + b_m)$$

Without the A_m, it reduces to the usual SPLICE formula,

$$x = y + \sum_m p(m|y) b_m$$

With one mixture component, it reduces to a linear (affine) transform.

$$x = Ay + b$$

Other SPLICE-like Transforms

- Probabilistic Optimal Filtering (POF)
  - Earliest work on this type of transform for ASR.
  - Transform uses current and previous noisy estimates.
- Region-Dependent Transform

Further reading:
L. Neumeyer and M. Weintraub. Probabilistic optimum filtering for robust speech recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 417-420, April 1994.
B. Zhang, S. Matsoukas and R. Schwartz. Recent progress on the discriminative region-dependent transform for speech feature extraction. *Proceedings Interspeech 2006, September 2006.*

Other SPLICE-like Transforms

- Stochastic Vector Mapping (SVM)
- Multi-environment models based linear normalization (MEMLIN)
  - Generalization for multiple noise types
  - Models joint probability between clean and noisy Gaussians.

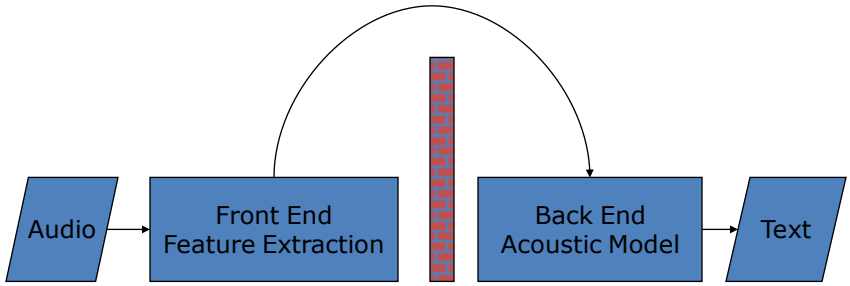
Further reading:

J. Wu and Q. Huo. An environment-compensated minimum classification error training approach based on stochastic vector mapping. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14, No. 6, November 2006, pp. 2147-2155.
M. Afify, X. Cui and Y. Gao. Stereo-based stochastic mapping for robust speech recognition. In *Proceedings ICASSP 2007*, (IV):377-380.
C.-H. Hsieh and C.-H. Wu. Stochastic vector mapping-based feature enhancement using prior models and model adaptation for noisy speech recognition. In *Speech Communication*, 50 (2008) 467-475.
L. Buera, E. Lleida, A. Miguel and A. Ortega. Multienvironment models based linear normalization for robust speech recognition in car conditions, *Proceedings ICASSP 2004.*

Training SPLICE-like Transformations

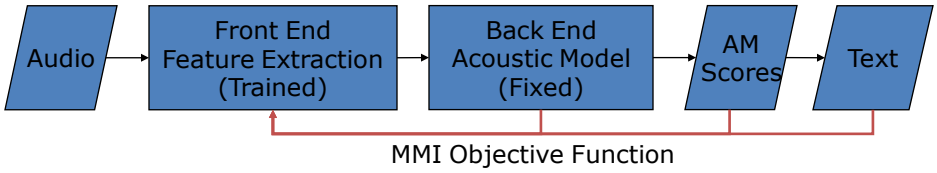
- Minimum mean-squared error (MMSE)
  - POF, SPLICE
  - Generally need stereo data.
- Maximum mutual information (MMI)
  - SVM, SPLICE
  - Objective function computed from cleanacoustic model.
- Minimum phone error (MPE)
  - RDT, fMPE
  - Objective function computed from clean acoustic model.

The Old Way: A Fixed Front End



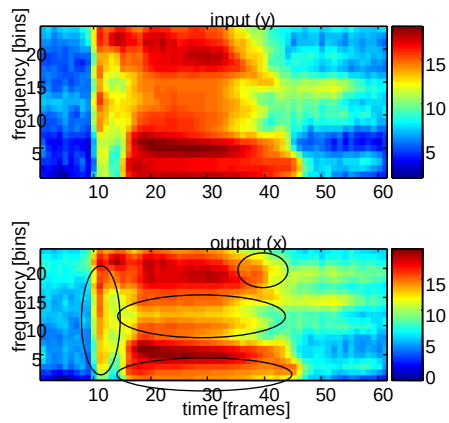
A Better Way: Trainable Front End

- Discriminatively optimize the feature extraction.
- Front end learns to feed better features to the back end.



Example

- The digit "2" extracted from the utterance clean1/FAK_3Z82A
- MMI-SPLICE modifies the features to match the canonical "two" model stored in the decoder.
- Four regions are modified:
  - The "t" sound at the beginning is broad-ened in frequency and smoothed
  - The low-frequency energy is suppressed
  - The mid-range energy is supressed
  - The tapering at the end of the top formant is smoothed



Since the Objective Function is Clearly Defined …

- Accuracy, or a discriminative measure like MMI or MPE.

$$\mathcal{F}_{MMI} = \frac{(p(X|w_c))^{\gamma}\, p(w_c)}{\sum_w (p(X|w))^{\gamma}\, p(w)} \approx p(w_c|X)$$

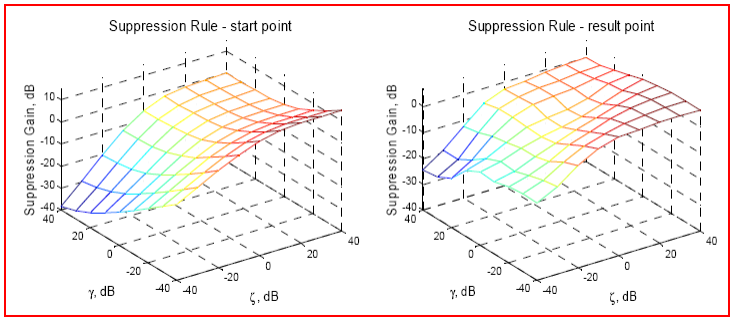
- Find derivative of objective function with respect to all parameters in the front end.
  - And I mean *all* the parameters.

$$\theta_k \leftarrow \theta_k + \alpha_k \frac{\partial \mathcal{F}_{MMI}}{\partial \theta_k}$$

- Typical 10%-20% relative error rate reduction
  - Depends on the parameter chosen.

Training the Suppression Parameters

- Suppression Rule Modification
  - Parameters are 9x9 sample grid
  - 12% fewer errors on average (Aurora 2)
  - 60% fewer errors on clean test (Aurora 2)



Training the Suppression Parameters

- There are many free parameters in a modern noise suppression system.
  - Decision directed learning rate, probability of speech transitions, spectral/temporal smoothing constants, thresholds, etc.
- Up to 20% improvement without changing the underlying algorithm.
- Automatic learning can find combinations of parameters that were unexpected by the system designer.

Further Reading:

J. Droppo and I. Tashev. Speech Recognition Friendly Noise Suppressor, in Proc. DSPA, Moscow, Russia, 2006.
J. Erkelens, J. Jensen and R. Heusdens. A data-driven approach to optimizing spectral speech enhancement methods for various error criteria. In *Speech Communication*, Vol. 49, No. 7-8, July-August 2007, pp. 530-541.

Overview

- Introduction
  - Standard noise robustness tasks
  - Overview of techniques to be covered
  - General design guidelines
- Analysis of noisy speech features
- Feature-based Techniques
  - Normalization
  - Enhancement
- **Model-based Techniques**
  - **Retraining**
  - **Adaptation**
- Joint Techniques
  - Noise Adaptive Training
  - Joint front-end and back-end training
  - Uncertainty Decoding
  - Missing Feature Theory

Model-based Techniques

- Goal is to approximate matched-condition training.
- Ideal scenario:
  - Sample the acoustic environment
  - Artificially corrupt a large database of clean speech
  - Retrain the acoustic model from scratch
  - Apply new acoustic model to the current utterance.
- Ideal scenario is infeasible, so we can choose to
  - Blindly adapt the model to the current utterance, or
  - Use a corruption model to approximate how the parameters would change, if retraining were pursued.

Retraining on Corrupted Speech

- Matched condition
  - All training data represents the current target condition.
- Multi-condition
  - Training data composed of a set of different conditions to approximate the types of expected target conditions.
- These are simple techniques that should be tried first.
- Requires data from the target environment.
  - Can be simulated

Model Adaptation

- Many standard adaptation algorithms can be applied to the noise robustness problem.
  - CMLLR, MLLR, MAP, etc.
- Consider a simple MLLR transform that is just a bias ***h***.

$$\hat{\mathrm{h}} = \mathrm{argmax}_{\mathbf{h}} p(\mathbf{X}|\mathbf{\Lambda}, \mathrm{h})$$

  - Solution is an E.-M. algorithm where all the means of the acoustic model are tied.
  - Compare to CMN, which blindly computes a bias.

Further reading:

M. Matassoni,M. Omologo and D. Giuliani. Hands-free speech recognition using a filtered clean corpus and incremental HMM adaptation. In *Proc. Int. Conf on Acoustics, Speech and Signal Processing*, pp. 1407-1410, 2000.

M.G. Rahim and B.H. Juang. Signal bias removal by maximum likelihood estimation for robust telephone speech recognition. *IEEE Trans. On Speech and Audio Processing*, 4(1):19-30, 1996.

Parallel Model Combination

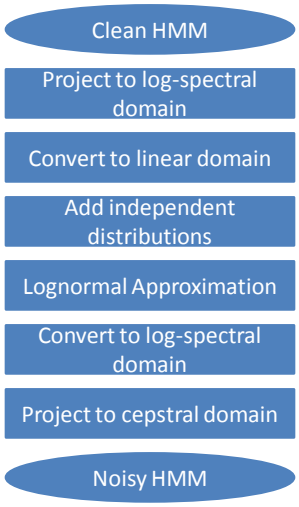
- Approximates retraining your acoustic model for the target environment.
- Compose a clean speech model with a noise model, to create a noisy speech model.

Further reading:
M.J. Gales. *Model Based Techniques for Noise Robust Speech Recognition*. Ph.D. thesis in engineering department, Cambridge University, 1995.

Parallel Model Combination "Data Driven"

- Procedure
  - Estimate a model for the additive noise.
  - Run Monte Carlo simulation to corrupt the parameters of your clean speech model.
  - Recognize using the corrupt model parameters.
- Analysis
  - Performance is limited only by the quality of the noise model.
  - More CPU-intensive than model-based adaptation.

Parallel Model Combination "Lognormal Approximation"



- Combine clean speech HMM with a model for the noise.
- Bring both models into the linear spectral domain
- Approximate adding of random variables
  - Assume the sum of two lognormal distributions is lognormal
- Convert back into cepstral model space.

Parallel Model Combination "Vector Taylor Series Approximation"

- Lognormal approximation is very rough.
- How can we do better?
  - Derive a more precise formula Gaussian adaptation.
  - Use a VTS approximation of this formula to adapt each Gaussian.

Vector Taylor Series Model Adaptation

- Similar in spirit to Lognormal PMC
  - Modify acoustic model parameters as if they had been retrained
- First, build a VTS approximation of ***y***.

$$
\begin{aligned}
\mathbf{y} &= \mathbf{x} + \mathbf{g}(\mathbf{n} - \mathbf{x}) \\
\mathbf{g}(\mathbf{z}) &= \mathbf{C}\ln\left(1 + \exp\left(\mathbf{D}\mathbf{z}\right)\right) \\
A &= \left.\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\right|_{(\mu_x, \mu_n)} \\
I - A &= \left.\frac{\partial \mathbf{y}}{\partial \mathbf{n}}\right|_{(\mu_x, \mu_n)} \\
\mathbf{y} &\approx \mu_x + \mathbf{g}(\mu_n - \mu_x) + A(\mathbf{x} - \mu_x) + (I - A)(\mathbf{n} - \mu_n)
\end{aligned}
$$

Vector Taylor Series Model Adaptation

- Then, transform the acoustic model means and covariances

$$
\begin{aligned}
\mu_y &\approx \mu_x + \mathbf{g}(\mu_n - \mu_x) \\
\Sigma_y &\approx A(\Sigma_x)A^T + (I - A)\Sigma_n(I - A)^T
\end{aligned}
$$

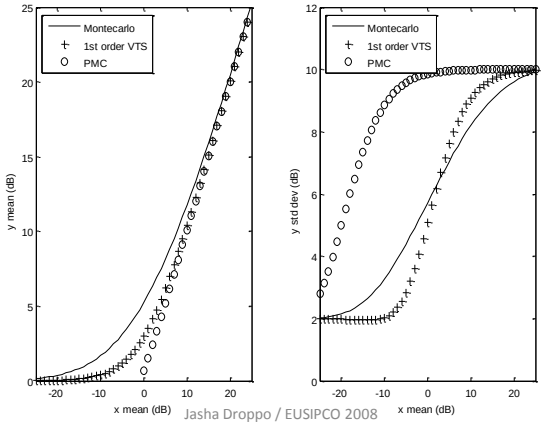
- In general, the new covariance will not be diagonal
  - It's usually okay to make a diagonal assumption

Further reading:

A. Acero, L. Deng, T. Kristjansson and J. Zhang. HMM adaptation using vector Taylor series for noisy speech recognition. In *Int. Conf on Spoken Language Processing*, Beijing, China, 2000.
J. Li, L. Deng, D. Yu, Y. Gong, A. Acero. HMM Adaptation Using a Phase-Sensitive Acoustic Distortion Model For Environment-Robust Speech Recognition, In *ICASSP 2008*, Las Vegas, USA., 2008.
P.J. Moreno, B. Raj and R.M. Stern. A vector Taylor series approach for environment independent speech recognition. In *Int. Conf on Acoustics, Speech and Signal Processing*, pp. 733-736, 1996.

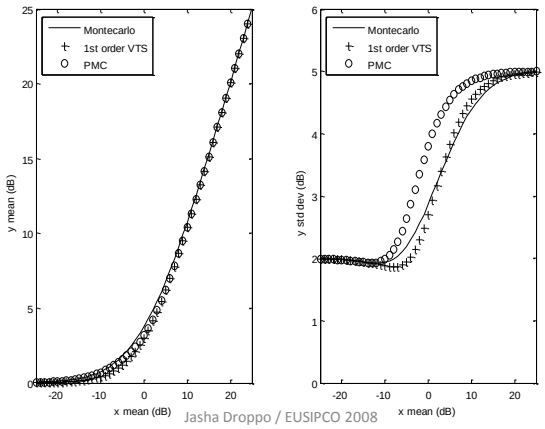
Data Driven, Lognormal PMC, and VTS

- Noise is Gaussian with mean 0dB, sigma 2dB
- "Speech" is Gaussian with sigma 10dB



Data Driven, Lognormal PMC, and VTS

- Noise is Gaussian with mean 0dB, sigma 2dB
- "Speech" is Gaussian with sigma 5dB

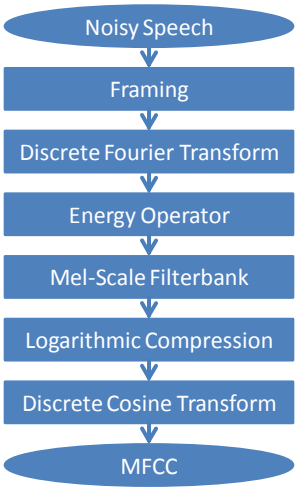


Vector Taylor Series Model Adaptation

- So, where does the function *g(z)* come from?

Vector Taylor Series Model Adaptation

- So, where does the function *g(z)* come from?
- To answer that, we need to trace the signal through the front-end.



A Model of the Environment

- Recall that acoustic noise is additive.

$$y[t] = x[t] + n[t]$$

- For the (framed) spectrum, noise is still additive.

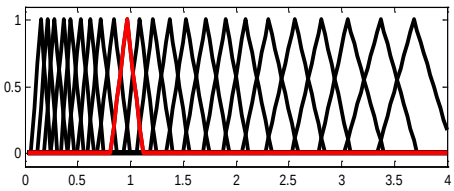
$$Y[k] = X[k] + N[k]$$

- When the energy operator is applied, noise is no longer additive.

$$|Y[k]|^2 = |X[k]|^2 + |N[k]|^2 + 2|X[k]N[k]|cos\theta$$

A Model of the Environment

- The mel-frequency filterbank combines
  - Dimensionality reduction
  - Frequency warping



- After logarithmic compression, the noisy *y*[*i*] is what we analyze.

$$\exp\left(y[i]\right) = \sum_k w_k^i |Y[k]|^2$$

Combining Speech and Noise

- Imagine two hypothetical observations
  - $x[i]$ = The observation that the clean speech would have produced in the absence of noise.
  - $n[i]$ = The observation that the noise would have produced in the absence of clean speech.
- We have the noisy observation $y[i]$.
- How do these three variables relate?

A model of the Environment: Grand Unified Equation

$$\exp y_i = \exp x_i + \exp n_i + 2\alpha_i \exp \frac{x_i + n_i}{2}$$

The conventional spectral subtraction model

Stochastic error caused by unknown phase of hidden signals.

Error term is also a function of hidden speech and noise features.

Further reading:

J. Droppo, A. Acero and L. Deng. A nonlinear observation model for removing noise from corrupted speech log mel-spectral energies. In *Proc. Int. Conf on Spoken Language Processing*, September 2002.
J. Droppo, L. Deng and A. Acero. A comparison of three non-linear observation models for noisy speech features. In *Proc. of the Eurospeech Conference*, September 2003.

The "phase term" is usually ignored

$$\begin{aligned}\exp(y_i) &= \exp(x_i) + \exp(n_i) + 2\alpha_i \exp\left(\frac{n_i + x_i}{2}\right) \\ y_i &\approx x_i + \ln\left(1 + \exp(n_i - x_i)\right)\end{aligned}$$

- Most cited reason: because expected value is zero.
  - But, every frame can be significantly different from zero.
  - We'll revisit this in a few slides.
- Appropriate when either *x* or *n* dominate the current observation.
- Otherwise, it represents (at best) a gross approximation to the truth.

Mapping to the cepstral space

- But, we've ignored the rest of the front end!
- The cepstral rotation
  - A linear (matrix) operation
  - From log mel-frequency filterbank coefficients (LMFB)
  - To mel-frequency cepstral coefficients (MFCC).

$$y_j^{\text{MFCC}} = \sum_i c_{ji} y_i^{\text{LMFB}}$$

- If the right-inverse matrix ***D*** is defined such that ***CD***=***I***, then the cepstral equation is

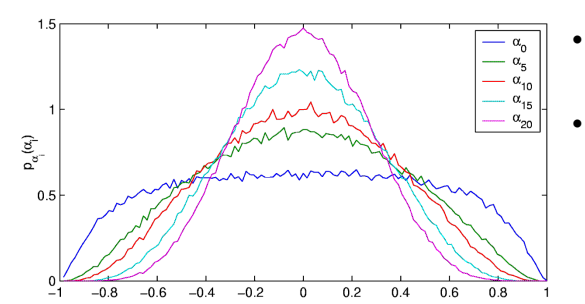
$$\begin{aligned}\mathbf{y} &= \mathbf{x} + \mathbf{C}\ln\left(1 + \exp\left(\mathbf{D}\left(\mathbf{n} - \mathbf{x}\right)\right)\right) \\ &= \mathbf{x} + \mathbf{g}\left(\mathbf{x} - \mathbf{n}\right)\end{aligned}$$

Vector Taylor Series: Correctly Incorporating Phase

- Unconditionally setting the phase to zero is a gross approximation.
- How much does it hurt the result?
- Where does it hurt most?
- Can we do better?

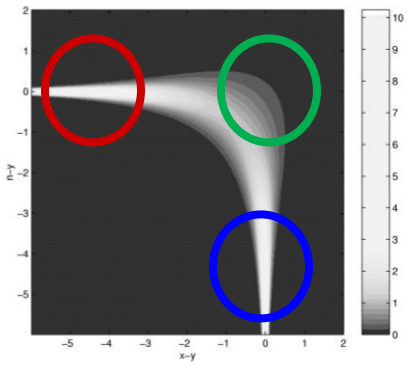
What is the Phase Term's Distribution?

$$\alpha_i = \frac{\sum_k w_k^i \, |X[k]| \, |N[k]| \cos\theta_k}{\sqrt{\sum_k w_k^i \, |X[k]|^2} \sqrt{\sum_k w_k^i \, |N[k]|^2}} \approx \sum_k \frac{w_k^i}{\sum_j w_j^i} \cos\theta_k.$$



- Distribution depends on filterbank.
- Approximately Gaussian for high frequency filterbanks.

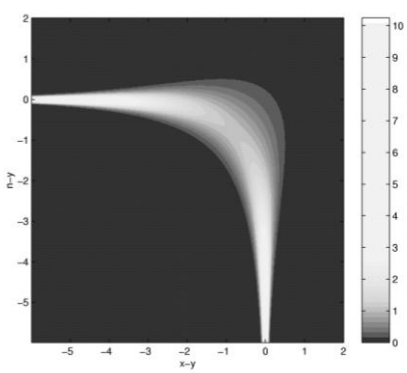
Theoretical Observation Likelihood



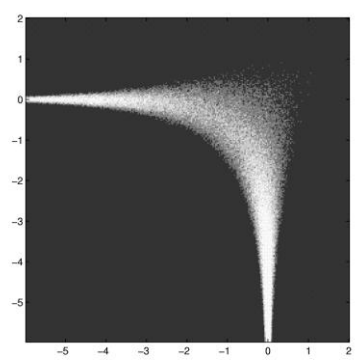
Model

- Observation Likelihood p(y|x,n) as a function of (x-y) and (n-y).
- Phase term broadens the distribution near 0dB SNR.
- n<y
- x<y
- x>y and n>y

Check: The Model Matches Real Data



Model



Data

Observation Likelihood

- The model places a hard constraint on four random variables, leaving three degrees of freedom:

$$p(y|x,n,\alpha) = \delta\left(y - \ln\left(e^x + e^n + 2\alpha e^{\frac{x+n}{2}}\right)\right).$$

- Third term is dependent on *x* and *n*.
  – For *x* >> *n* and *x* << *n*, error term is relatively small.

SNR Dependent Variance Model

- Including a Gaussian prior for alpha, and marginalizing, yields:

$$\ln p(y|x,n) = y - \frac{x+n}{2} - \frac{1}{2}\ln 8\pi\sigma_\alpha^2 - \frac{(e^y - e^x - e^n)^2}{8\sigma_\alpha^2 e^{(x+n)}}.$$

- But, this non-Gaussian posterior is difficult to evaluate properly.

Further reading:
J. Droppo, L. Deng and A. Acero. A comparison of three non-linear observation models for noisy speech features. In Proc. of the Eurospeech Conference, September 2003.

SNR Independent Variance Model

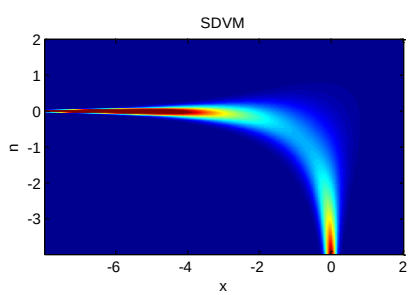
- The SIVM assumes the error term is small, constant, and independent of *x* and *n*. [Algonquin]

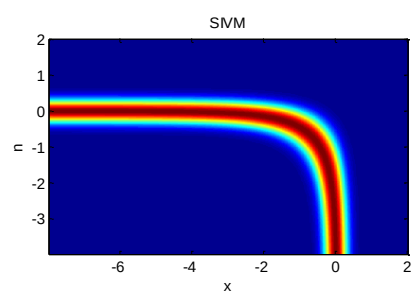
$$\begin{aligned} y &= \ln(\exp x + \exp n) + \epsilon \\ \epsilon &\sim N(\epsilon; 0, \psi) \end{aligned}$$

- Gaussian posterior is easy to derive and to evaluate.

$$p(y|x,n) = N(y; \ln(\exp x + \exp n), \psi)$$

Modeling and Complexity Tradeoff





- Models all regions equally well.
- Costly to implement properly.

- Models all regions poorly.
- More economical implementation.

Zero Variance Model

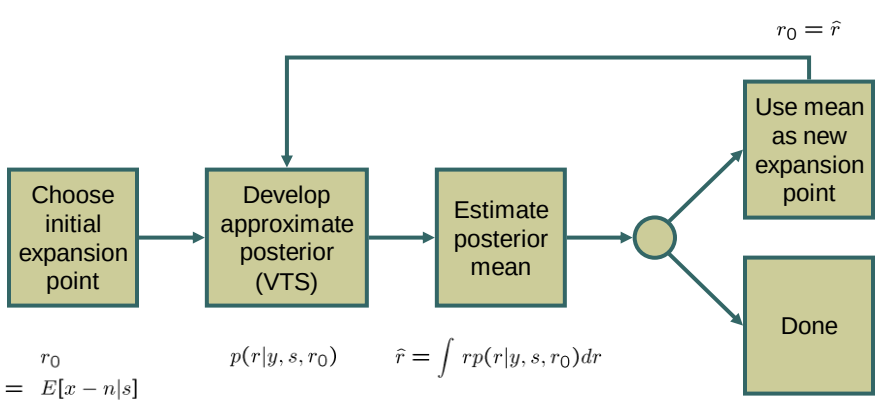
- A special, simpler case of both the SDVM and SIVM.
  - Correct model for high and low instantaneous SNR.
  - Approximate for 0dB SNR.
- Assume the phase term is always exactly zero.
- Introduce a new instantaneous SNR variable *r*.

$$\begin{aligned} r &= x - n \\ y &= \ln(\exp x + \exp n) \end{aligned}$$

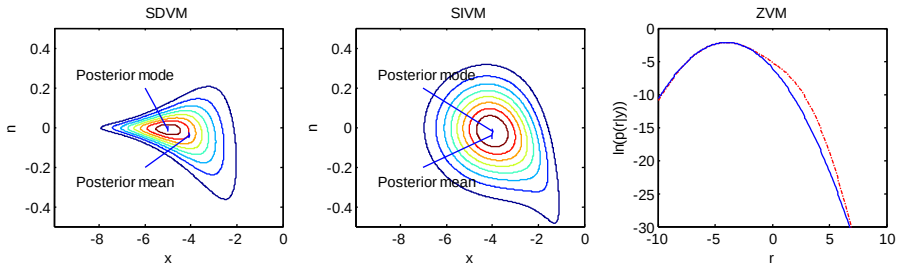
- Replace inference on x and n with inference on *r*.

$$\begin{aligned} x &= y - \ln(e^r + 1) + r \\ n &= y - \ln(e^r + 1). \end{aligned}$$

Iterative VTS



Iterative VTS: Behavior



- Iterative VTS is a Gaussian approximation that converges to a local maximum of the posterior.
- The SDVM is a non-Gaussian distribution whose mean and maximum are not co-incident.
- As a result, Iterative VTS fails spectacularly when used with the SDVM.
  - Good inference schemes under the CDVM have been developed [ICSLP 2002] but come at a high computational cost.

- Now that we know where g(z) comes from…
- What else is it good for?

Vector Taylor Series Enhancement

- Full VTS model adaptation
  - Can be quite expensive.
- VTS Enhancement
  - Uses the power of VTS to enhance the speech features.
  - Computes MMSE estimate of clean speech given noisy speech, and a noise model.
  - Recognizes with a standard acoustic model.

Vector Taylor Series Enhancement

- … Is very popular.

Further reading:

P.J. Moreno, B. Raj and R.M. Stern. A vector taylor series approach for environment independent speech recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 417-420, April 1994.
J. Droppo, A. Acero and L. Deng. A nonlinear observation model for removing noise from corrupted speech log mel-spectral energies. In Proc. Int. Conf on Spoken Language Processing, September 2002.
J. Droppo, L. Deng and A. Acero. A comparison of three non-linear observation models for noisy speech features. In Proc. of the Eurospeech Conference, September 2003.
B.J. Frey, L. Deng, A. Acero and T. Kristjansson. ALGONQUIN: Iterating Laplace's method to remove multiple types of acoustic distortion for robust speech recognition. In *Proc. Eurospeech*, 2001.
C. Couvreur and H. Van Hamme. Model-based feature enhancement for noisy speech recognition. In *Proc. ICASSP*, Vol. 3, pp. 1719-1722, June 2000.
W. Lim, J. Kim and N. Kim. Feature compensation using more accurate statistics of modeling error. In *Proc. ICASSP*, Vol. 4, pp. 361-364, April 2008.
W. Lim, C. Han, J. Shin and N. Kim. Cepstral domain feature compensation based on diagonal approximation. In *Proc. ICASSP*, pp. 4401-4404, 2007.
V. Stouten. Robust automatic speech recognition in time-varying environments. Ph.D. Dissertation, Katholieke Universitet Leuven, September 2006.
S. Windmann and R. Haeb-Umbach. An Approach to Iterative Speech Feature Enhancement and Recognition. In *Proc. Interspeech*, pp. 1086-1089, 2007.

VTS Enhancement

- The true minimum mean squared estimate for the clean speech should be,

$$\hat{x} = \sum_s E[x|y,s]p(s|y)$$

- A good approximation for this expectation, given the parameters available from the ZVM, is

$$E[x|y,s] \approx y - \ln(e^{\hat{\mu}_s^r} + 1) + \hat{\mu}_s^r$$

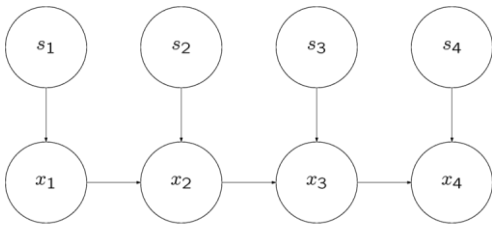
- Since the expectation is only approximate, the estimate is sub-optimal.

Using More Advanced Models with VTS Enhancement

- Hidden Markov models.
  - Replace the (time-indepenent) GMM mixture component with a Markov chain.
  - Observation probability still dominates.
  - More complex models (e.g., phone-loop or full decoding) propagate their errors forward.

Using More Advanced Models with VTS Enhancement

- Switching Linear Dynamic Models.
  - Inference is difficult (exponentially hard)



$$p(x_t, s_t | x_{t-1}) = N(x_t; A_{s_t} x_{t-1} + b_{s_t}, C_{s_t}) p(s_t)$$

$$p(x_1^T, s_1^T) = p(x_1, s_1) \prod_{t=2}^{T} p(x_t, s_t | x_{t-1})$$

Using More Advanced Models with VTS Enhancement

- Solutions to the Inference Problem
  - Generalized pseudo-Bayes (GPB)
  - Particle filters

Further reading:

J. Droppo and A. Acero. Noise robust speech recognition with a switching linear dynamic model. In *Proc. ICASSP,* May, 2004.
S. Windmann and R. Haeb-Umbach. An Approach to Iterative Speech Feature Enhancement and Recognition. In *Proc. Interspeech*, pp. 1086-1089, 2007.
B. Mesot and D. Barber. Switching linear dynamical systems for noise robust speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 15, No. 6, pp. 1850-1858, August 2007.
S. Windman and R. Haeb-Umbach. Modeling the dynamics of speech and noise for speech feature enhancement in ASR. IN *Proc. ICASSP*, pp. 4409-4412, April 2008.
N. Kim, W. Lim and R.M. Stern. Feature compensation based on switching linear dynamic model. *IEEE Signal Processing Letters*, Vol. 12, No. 6, pp. 473-476, June 2005.

Overview

- Introduction
  - Standard noise robustness tasks
  - Overview of techniques to be covered
  - General design guidelines
- Analysis of noisy speech features
- Feature-based Techniques
  - Normalization
  - Enhancement
- Model-based Techniques
  - Retraining
  - Adaptation
- **Joint Techniques**
  - **Noise Adaptive Training**
  - **Joint front-end and back-end training**
  - **Uncertainty Decoding**
  - **Missing Feature Theory**

Joint Techniques

- Joint methods address the inefficiency of partitioning the system into feature extraction and pattern recognition.
  - Feature extraction must make a hard decision
  - Hand-tuned feature extraction may not be optimal
- Methods to be discussed
  - Noise Adaptive Training
  - Joint Training of Front and Back Ends
  - Uncertainty Decoding
  - Missing Feature Theory

Noise Adaptive Training

- A combination of multistyle training and enhancement.
- Apply speech enhancement to multistyle training data.
  - Models are tighter, they don't need to describe all the variability introduced by different noises.
  - Models learn the distortions introduced by the enhancement process.
- Helps generalization
  - Under unseen conditions, the residual distortion can be similar, even if the noise conditions are not.

Further reading:

L. Deng, A. Acero, M. Plumpe and X.D. Huang. Large-vocabulary speech recognition under adverse acoustic environments. In *Int. Conf on Spoken Language Processing*, Beijing, China, 2000.

Joint Training of Front and Back Ends

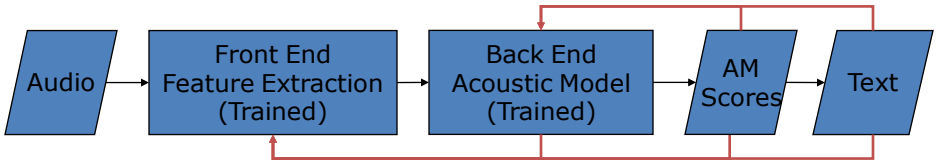
- A general discriminative training method for both the front end feature extractor and back end acoustic model of an automatic speech recognition system.
- The front end and back end parameters are jointly trained using the Rprop algorithm against a maximum mutual information (MMI) objective function.

Further reading:

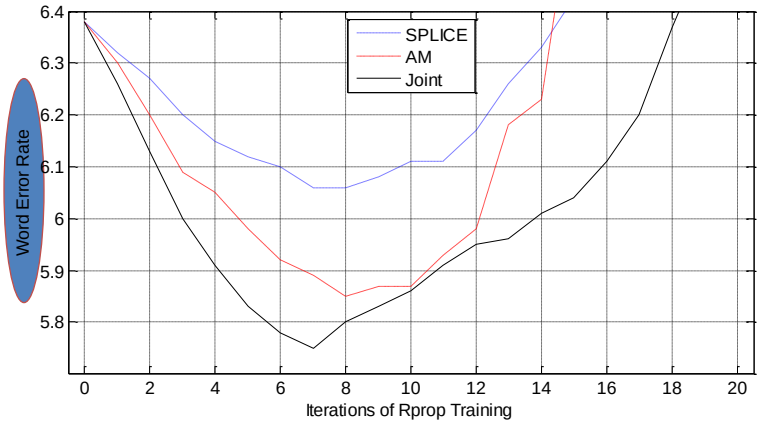
J. Droppo and A. Acero. Maximum mutual information SPLICE transform for seen and unseen conditions. In *Proc. Of the Interspeech Conference*, Lisbon, Portugal, 2005.
D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau and G. Zweig. FMPE: discriminatively trained features for speech recognition. In *Proc. ICASSP*, 2005.

The New Way: Joint Training

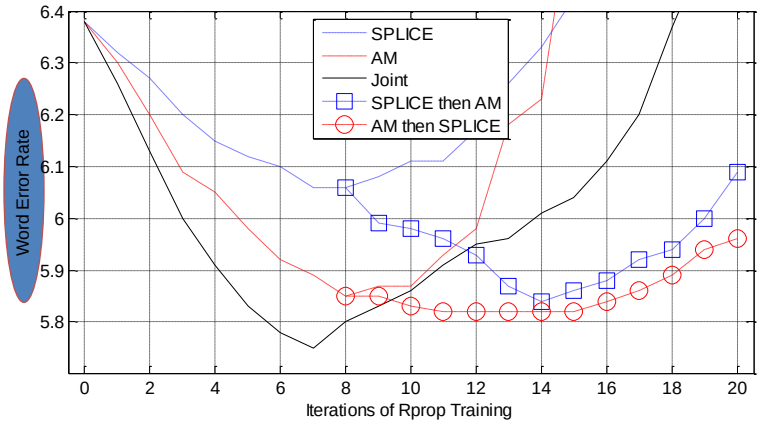
- Front end and back end updated simultaneously.
- Can cooperate to find a good feature space.



Joint training is better than either SPLICE or AM alone.



Joint training is better than serial training.



Uncertainty Decoding and Missing Feature Techniques

- Not all observations generated by the front end should be treated equally.
- Uncertainty Decoding
  - Grounded in probability and estimation theory
  - Front-end gives cues to the back-end indicating the reliability of feature estimation
- Missing Feature Theory
  - Grounded in auditory scene analysis
  - Estimates which observations are buried in noise (missing)
  - Mask is created to partition the features into reliable and missing (hidden) data.
  - The missing data is either *marginalized* in the decoder (similar to uncertainty decoding), or *imputed* in the front end.

Uncertainty Decoding

- When the front-end enhances the speech features, it may not always be confident.
- Confidence is affected by
  - How much noise is removed
  - Quality of the remaining cues
- Decoder uses this confidence to modify its likelihood calculations.

Uncertainty Decoding

- The decoder wants to calculate p(y|m), but its parameters model p(x|m).
- Front-end provided p(y|x), the probability the existing observation would be produced as a function of x.

$$
\begin{aligned}
p(\mathbf{y}|m) &= \int_{-\infty}^{\infty} p(\mathbf{y}|\mathbf{x}, m) p(\mathbf{x}|m)\, d\mathbf{x} \\
&= \alpha \int_{-\infty}^{\infty} N(\hat{\mathbf{x}}; \mathbf{x}, \sigma_{\hat{\mathbf{x}}}^2) N(\mathbf{x}; \mu_m, \sigma_m^2)\, d\mathbf{x} \\
&= \alpha N(\hat{\mathbf{x}}; \mu_m, \sigma_m^2 + \sigma_{\hat{\mathbf{x}}}^2)
\end{aligned}
$$

Uncertainty Decoding

- The trick is calculating reasonable parameters for $p(y|x)$.
- SPLICE
  - Model the residual variance in addition to bias.

$$\begin{aligned} r_m &= E[x-y|m] \\ \Gamma_m &= E[(x-y)^2|m] \end{aligned}$$

  - Compute $p(y|x)$ using Bayes' rule and other approximations.

Uncertainty Decoding

- For VTS, use the VTS approximation to get parameters of p(y|x).

Further reading:

J. Droppo, A. Acero and L. Deng. Uncertainty decoding with SPLICE for noise robust speech recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, 2002.
H. Liao and M.J.F. Gales. Joint uncertainty decoding for noise robust speech recognition. In *Proc. Interspeech*, 2005.
H. Liao and M.J.F. Gales. Issues with uncertainty decoding for noise robust automatic speech recognition. *Speech Communication,* Vol. 50, No. 4, pp. 265-277, April 2008.
V. Stouten, H. Van hamme and P. Wambacq. Model-based feature enhancement with uncertainty decoding for noise robust ASR. *Speech Communication*, Vol. 48, No. 11, November 2006, pp. 1502-1514.

Missing Feature: Spectrographic Masks

- A binary mask that partitions the spectrogram into reliable and unreliable regions.
- The reliable measurements are a good estimate of the clean speech.
- The unreliable measurements are an estimate of an upper bound for clean speech.

$$
\begin{aligned}
Y(n,k) &= \log\left(\exp\left(X(n,k)\right) + \exp\left(N(n,k)\right)\right) \\
X_r(n,k) &= Y_r(n,k) \\
X_u(n,k) &\leq Y_u(n,k)
\end{aligned}
$$

Missing Feature: Mask Estimation

- SNR Threshold and Negative Energy Criterion
  - Easy to compute.
  - Requires estimate of the noise spectrum.
  - Unreliable with non-stationary additive noises.
- Bayesian Estimation
  - Measure SNR and other features for each spectral component.
  - Build a binary classifier for each frequency bin based on these measurements.
  - More complex system, but more reliable with non-stationary additive noises.

Further reading:

A.Vizinho, P. Green, M. Cooke and L. Josifovski. Missing data theory, spectral subtraction and signal-to-noise estimation for robust ASR: An integrated study. *Proc. Eurospeech*, pp. 2407-2410, Budapest, Hungary, 1999.
M.K. Seltzer, B. Raj and R.M. Stern. A Bayesian framework for spectrographic mask estimation for missing feature speech recognition. *Speech Communication*, 43(4):379-393, 2004.

Missing Feature: Imputation

- Replace missing values from x with estimates based on the observed components.
- Decode on reliable and imputed observations.

$$\hat{X}_m(m,k) \;=\; E\left[X_m(n,k)|X_r(n,k)=Y_r(n,k)\right]$$

Further reading:

Bhiksha Raj and R.M. Stern. Missing-feature approaches in speech recognition. *IEEE Signal Processing Magazine*, 22(5):101-116, September, 2005.
M. Cooke, P. Green, L. Josifovski and A. Vizinho. Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Communication*, 34(3):267-285, June 2001.
P. Green, J.P. Barker, and M. Cooke. Robust ASR based on clean speech models: An evaluation of missing data techniques for connected digit recognition in noise. In *Proc. Eurospeech 2001*, pp. 213-216, September, 2001.
B. Raj, M. Seltzer and R. Stern. Reconstruction of missing features for robust speech recognition. *Speech Communication,* Vol. 43, No. 4, September 2004, pp. 275-296.

Missing Feature: Imputation

- Cluster-based reconstruction
  - Assume time slices of the spectrogram are IID.
  - Build a GMM to describe the PDF of these time slices.
  - Use the spectrographic mask, GMM, and bounds on the missing data to estimate the missing data.
- "Bounded Maximum a priori Approximation"

$$\begin{aligned}\hat{X}_m &= \arg\max_{X_m}\left\{P(X_m|X_r, X_m \le Y_m)\right\}\\ &\approx \sum_{\nu} P(\nu|X_r, X_m \le Y_m)\mathrm{BMAP}(\nu|X_r, X_m \le Y_m; \mu_\nu, \Theta_\nu)\end{aligned}$$

Missing Feature: Imputation

- Covariance-based reconstruction
  - Assume spectrogram is a sequence of correlated vector-valued Gaussian random process.

$$\begin{aligned} \mu(k) &= E\{X(n,k)\} \\ c(\zeta, k_1, k_2) &= E\left\{(X(n,k_1) - \mu_{k_1})(X(n+\zeta, k_2) - \mu_{k_2})\right\} \end{aligned}$$

- Compute bounded MAP estimate of missing data.
  - Ideally, simultaneous joint estimation of all missing data.
  - Practically, estimate one frame at a time from neighboring reliable components

Missing Feature: Classifier Modification

- Marginalization
  - When computing p(x|m) in the decoder, integrate over possible values of the missing x.
  - Similar to Uncertainty Decoding, when the uncertainty becomes very large.

$$\begin{aligned} p(\mathbf{x}|m) &= p(\mathbf{x}_r|m)p(\mathbf{x}_u|m) \\ &= p(\mathbf{x}^r|m)\int_{-\infty}^{\mathbf{y}_u} d\mathbf{x}_u p(\mathbf{x}_u|m) \end{aligned}$$

Missing Feature: Classifier Modification

- Fragment Decoding
  - Segment the data into two parts
    - "dominated by target speaker" (the reliable fragments)
    - "everything else" (the masker)
  - Decode over the fragments.
  - Not naturally computationally efficient
  - Quite promising for very noisy conditions, especially "competing speaker".

Further reading:

J.P. Barker, M.P. Cooke and D.P.W. Ellis. Decoding speech in the presence of other sources. *Speech Communication*, Vol. 45, No. 1, January 2008, pp. 5-25.

J. Barker, N. Ma, A. Coy and M. Cooke. Speech fragment decoding techniques for simultaneous speaker identification and speech recognition. *Computer Speech & Language*, *in press*, corrected proof available on-line May 2008.

Summary

- Evaluate on standard tasks
  - Good sanity check for your code
  - Allows others to evaluate your algorithm
- Spend the effort for good audio capture
- Use training data similar to what is expected at runtime
- Implement simple algorithms first, then move to more complex solutions
  - Always include feature normalization
  - When possible, add model adaptation
  - To achieve maximum performance, or if you can't retrain the acoustic model, implement feature enhancement.