The Iroquois Model: Using Temporal Dynamics to Separate Speakers

Steven Rennie, Peder Olsen, John Hershey, Trausti Kristjansson

IBM T. J. Watson Research Center, Yorktown Heights, NY, USA

Abstract

We describe a system that can separate and recognize the simultaneous speech of two speakers from a single channel recording and compare the performance of the system to that of human subjects.

The system, which we call *Iroquois*, uses models of dynamics to achieve performance near that of human listeners, when averaged across all conditions. However the system exhibits a pattern of performance across conditions that is different from that of human subjects. In conditions where the amplitude of the speakers is similar, our surpasses human performance by over 50%. We hypothesize that the system accomplishes this remarkable feat by employing a different strategy to that of the human auditory system.

1. Introduction

Listening to and understanding the speech of two people when they talk simultaneously is a difficult task and has been considered one of the most challenging problems for automatic speech recognition. The ICSLP 2006 Speech Separation Challenge [1] gives us an opportunity to demonstrate the importance of temporal dynamics at an acoustic and sentence level, and to contrast the system performance to that of human subjects.[1]

Single-channel speech separation has previously been attempted using Gaussian mixture models (GMMs) on individual frames of acoustic features. However such models tend to perform well only when speakers are of different gender or have rather different voices [3]. When speakers have similar voices, speaker-dependent mixture models cannot unambiguously identify the component speakers. In such cases it is helpful to model the temporal dynamics of the speech. Several models in the literature have attempted to do so either for recognition [4, 5] or enhancement [6, 7] of speech. Such models have typically been based on a discrete state hidden Markov model (HMM) operating on a frame-based acoustic feature vector.

One of the challenges of such modeling is that speech contains patterns at different levels of detail, that evolve at different time-scales. For instance, two major components of the voice are the excitation, which consists of pitch and voicing, and the filter, which consists of the formant structure due to the mouth position. The pitch appears in the short-time spectrum as a closely-spaced harmonic series of peaks, whereas the formant structure has a smooth frequency envelope. The formant structure and voicing are closely related to the phoneme being spoken, whereas the pitch evolves somewhat independently of the phonemes during voiced segments.

At small time-scales these processes evolve in a somewhat predictable fashion, with relatively smooth pitch and formant trajectories, interspersed with sharper transients. If we begin with a Gaussian mixture model of the log spectrum, we can hope to capture something about the dynamics of speech by just looking at pair-wise relationships between the acoustic states ascribed to individual frames of speech data.

In addition to these low-level acoustical constraints, there are linguistic constraints that describe the dynamics of syllables, words, and sentences. These constraints depend on context over a longer time-scale and hence cannot be modeled by pair-wise relationships between acoustic states. In speech recognition systems such long-term relationships are handled using concatenated left-to-right models of context-dependent phonemes, that are derived from a grammar or language model.

Typically models in the literature have focused on only one type of dynamics, although some models have factored the dynamics into excitation and filter components [8]. Here we explore the combination of low-level acoustic dynamics with high-level grammatical constraints. We compare three levels of dynamic constraints: simple GMM models of the log spectrum with no dynamics, acoustic-level HMM dynamics, and a layered combination of acoustic-level and grammar-level dynamics. The models are combined at the observation level using a nonlinear model known as Algonquin, which models the sum of log-normal spectrum models. Inference on the state level is carried out using an iterative two-dimensional Viterbi decoding scheme.

Using both acoustic and sentence level dynamics our signal separation system, which we call *Iroquois*, produces remarkable results: it is often able to extract two utterances from a mixture even when they are from the same speaker. [2].

The system is composed of the three components: a speaker identification and gain estimation component, a signal separation component, and a speech recognition system.

Section two and three describe the acoustic model and dynamics of the signal separation system. Section four describes the speaker identification and gain estimation system, section five describes the speaker-dependent labeling (SDL) recognizer, and section six describes the experiments and results.

2. Acoustic Models and Likelihood Estimation

The speech separation challenge involves recognizing speech in files that are mixtures of two component signals, x_t^a and x_t^b.

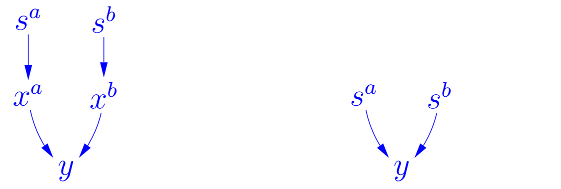
The model for mixed speech in the time domain is (omitting the channel) $y_t = x_t^a + x_t^b$ where y_t denotes the mixed signal at time t. We approximate this relationship in the log power spectral domain as

$$p(\mathbf{y}|\mathbf{x}^a, \mathbf{x}^b) = N(\mathbf{y}; \ln(\exp(\mathbf{x}^a) + \exp(\mathbf{x}^b)), \mathbf{\Psi}) \qquad (1)$$

where $\mathbf{\Psi}$ is introduced to model the error due to the omission of phase, and time has been omitted for simplicity.

[1] We expand upon the conference version presented at ICSLP 2006[2].

[2] Audio samples and further information can be found at



(a) Combining GMMs

(b) Simplified Model

Figure 1: Graphical models of the feature layer of our separation system. In (a) all dependencies are shown. In (b) the source features $\mathbf{x}^a$ and $\mathbf{x}^b$ have been integrated out.

We model the conditional probability of the log-power spectrum of the each source signal given their acoustic state as gaussian: $p(\mathbf{x}^a|s^a) = N(\mathbf{x}^a; \mu_{s^a}, \Sigma_{s^a})$.

The joint distribution of the observation and source features given the source states is:

$$p(\mathbf{y}, \mathbf{x}^a, \mathbf{x}^b | s^a, s^b) = p(\mathbf{y}|\mathbf{x}^a, \mathbf{x}^b) p(\mathbf{x}^a|s^a) p(\mathbf{x}^b|s^b). \qquad (2)$$

Figure 1 depicts graphical models describing the relationships between the random variables of the feature layer of our speech separation system.

2.1. Likelihood Estimation

Unlike a traditional recognizer, we take into account the joint evolution of the two signals simultaneously. We therefore need to evaluate the joint state likelihood $p(\mathbf{y}|s^a, s^b)$ at every time step.

The iterative Newton-Laplace method Algonquin [3] can be used to accurately approximate the conditional posterior $p(\mathbf{x}^a, \mathbf{x}^b|s^a, s^b)$ from (2) as Gaussian, and to compute an analytic approximation to the observation likelihood $p(\mathbf{y}|s^a, s^b)$. The approximate joint posterior $p(\mathbf{x}^a, \mathbf{x}^b|\mathbf{y})$ is therefore a GMM and the minimum mean squared error (MMSE) estimators $E[\mathbf{x}^i|\mathbf{y}]$ or the maximum *a posteriori* (MAP) state-based estimate $(\hat{s^a}, \hat{s^b}) = \arg\max_{s^a, s^b} p(s^a, s^b|\mathbf{y})$ may be analytically computed and used to form an estimate of $\mathbf{x}^a$ and $\mathbf{x}^b$, given a prior for the joint state $\{s^a, s^b\}$.

We used 256 Gaussians, one per acoustic state, to model the acoustic space of each speaker. Dynamic state priors on these acoustic states are described in section three, In this case, the computation of $p(\mathbf{y}|s^a, s^b)$ requires the evaluation of 256^2 or over 65k state combinations.

2.2. Fast Likelihood Estimation

In order to speed up the evaluation of the joint state likelihood, we employed both *band quantization* of the acoustic Gaussians and joint-state pruning.

One source of computational savings stems from the fact that some of the Gaussians in our model may differ only in a few features. Band quantization addresses this by approximating each of the D Gaussians of each model with a shared set of d Gaussians, where $d \ll D$, in each of the F frequency bands of the feature vector. A similar idea is described in [9]. It relies on the use of a diagonal covariance matrix, so that $p(x^a|s^a) = \prod_f N(x^a_f; \mu_{f,s^a}, \sigma^2_{f,s^a})$, where σ^2_{f,s^a} are the diagonal elements of covariance matrix Σ_{s^a}. The mapping $M_f(s^i)$ associates each of the D Gaussians with one of the d Gaussians in band f. Now $\hat{p}(x^a|s^a) = \prod_f N(x^a_f; \mu_{f,M_f(s^a)}, \sigma^2_{f,M_f(s^a)})$ is used as a surrogate for $p(x^a|s^a)$.

Under this model the d Gaussians are chosen to minimize the KL-distance $D(p(x^a|s^a)||\hat{p}(x^a|s^a))$, and likewise for s^b. Then in each frequency band, only $d \times d$, instead of $D \times D$ combinations of Gaussians have to be evaluated to compute $p(\mathbf{y}|s^a, s^b)$.

Despite the relatively small number of components d in each band, taken across bands, the model is in theory capable of expressing d^F distinct patterns. In practice only a subset of the possible patterns match the Gaussians in a given model well enough to achieve good results. In our case, we acheived good results with $d = 8$ and $D = 256$. This saved over three orders of magnitude of computation time over the exhaustive approach.

Another source of computational savings comes from the sparseness of the model. Only a handful of s^a, s^b combinations have likelihoods that are significantly larger than the rest for a given observation. Only these states are required to adequately explain the observation. By pruning the total number of combinations down to a smaller number we can speed up the likelihood calculation, estimation of the components signals, as well as the temporal inference.

However, we must evaluate the likelihoods in order to determine which states to retain. Therefore we use faster approximations to initially estimate the likelihoods, followed by slower but more accurate methods after pruning. The *max* approximation [4, 10] provides an efficient approximation to the joint observation likelihood. The max approximation assumes $p(\mathbf{y}|s^a, s^b) = p_{x^a}(\mathbf{y}|s^a)$ if the mean μ^a of x^a is larger than the mean μ^b of x^b and $p(\mathbf{y}|s^a, s^b) = p_{x^b}(\mathbf{y}|s^b)$ otherwise.

We relied on the max approximation for speaker identification and gain estimation. For signal separation we used band-quantization to perform state pruning, and then Algonquin method on the pruned states using the original un-quantized parameters. In the experiments reported here, we pruned down to 256 state combinations. The effect of these speedup methods on accuracy will be reported in a future publication.

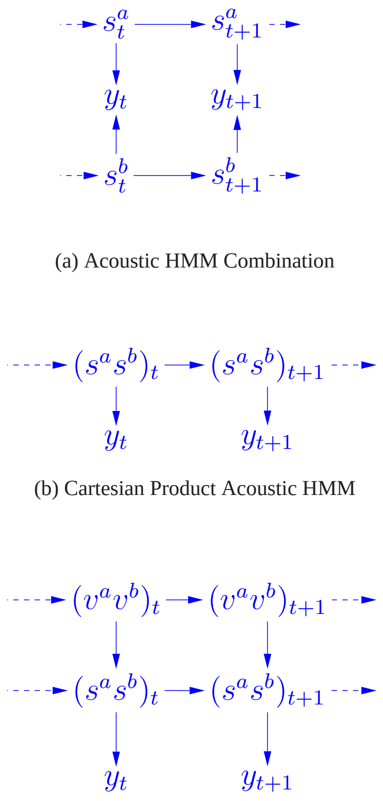
3. Temporal Dynamics

In a traditional speech recognition system, speech dynamics are captured by state transition probabilities. We took this approach and incorporated both *acoustic dynamics* and *grammar dynamics* via state transition probabilities.

3.1. Acoustic dynamics

To capture acoustic level dynamics, which directly models the dynamics of the log-spectrum, we estimated transition probabilities between the 256 acoustic states for each speaker. The acoustic dynamics of the two independent speakers are modeled by state transitions $p(s^a_{t+1}|s^a_t)$ and $p(s^b_{t+1}|s^b_t)$ for speaker a and b respectively, as shown in Figures 2(a) and 2(b). Hence, for each speaker c, we estimated a 256×256 component transition matrix A_c.

http : //www.research.ibm.com/speechseparation



(a) Acoustic HMM Combination

(b) Cartesian Product Acoustic HMM

(c) Grammar + Acoustic Model

Figure 2: Graph of acoustic HMM model for two sources. In (a), the two state chains are shown separately. In (b), the s^a and s^b are combined into a Cartesian product state $(s^a s^b)$. In (c) a Cartesian product of two grammars v^a and v^b has been added on top of the acoustic state sequence. Note that this makes the graphical model loopy.

3.2. Grammar dynamics

The grammar dynamics are modeled by grammar state transitions, $p(v_{t+1}^c|v_t^c)$, which consist of left-to-right phone models. The legal word sequences are given by the Speaker Separation Challenge grammar [1] and are modeled using pronunciations that map to three-state context-dependent phone models. The sequences of phone states for each pronunciation, along with self-transitions produce a Finite State Machine (FSM). The state transition probabilities derived from this machine are sparse in the sense that most state transition probabilities are zero.

For a given speaker, the FSM of our system has 506 grammar states v. We then model speaker dependent distributions $p(s^c|v^c)$ that associate the FSM states to the speaker-dependent acoustic states. These are learned from training data where the grammar state sequences and acoustic state sequences are known for each utterance. This combined model is depicted in Figure 2(c).

To combine the acoustic dynamics with the grammar dynamics, it was useful to avoid modeling the full combination of s and v states in the joint transitions $p(s_{t+1}^c|s_t^c, v_t)$. Instead we make a naive-Bayes assumption to approximate this as $\frac{1}{z}p(s_{t+1}^c|s_t^c)p(s_{t+1}^c|v_{t+1})$, where z is the normalizing constant.

3.3. 2D Viterbi search

The Viterbi algorithm estimates the maximum-likelihood state sequence $s_{1..T}$ given the observations $x_{1..T}$. The complexity of the Viterbi search is $O(TD^2)$ where D is the number of states and T is the number of frames. For producing MAP estimates of the 2 sources, we require a 2 dimensional Viterbi search which finds the most likely joint state sequences $s_{1..T}^a$ and $s_{1..T}^b$ given the mixed signal $y_{1..T}$ as was proposed in [4].

On the surface, the 2D Viterbi search appears to be of complexity $O(TD^4)$. Surprisingly, it can be computed in $O(TD^3)$ operations. This stems from the fact that the dynamics for each chain are independent. It is easy to show, for example, how the dynamics decouple in a forward inference algorithm:

$$\begin{aligned} & p(s_t^a, s_t^b|y_{1..t}) \\ = & \sum_{s_{t-1}^a s_{t-1}^b} p(s_t^a|s_{t-1}^a)p(s_t^b|s_{t-1}^b)p(s_{t-1}^a, s_{t-1}^b|y_{1..t-1}) \\ = & \sum_{s_{t-1}^a} p(s_t^a|s_{t-1}^a) \sum_{s_{t-1}^b} p(s_t^b|s_{t-1}^b)p(s_{t-1}^a, s_{t-1}^b|y_{1..t-1}). \end{aligned}$$

Computing the inner sum takes $O(D^3)$ operations and can be stored in $O(D^2)$ memory, and the outer sum is of the same complexity. The backward inference algorithm is of the same complexity. In general the forward-backward algorithm for a factorial HMM with N state variables requires only $O(TND^{N+1})$ rather than the $O(TD^{2N})$ required for a naive implementation [11].

In the Viterbi algorithm, we wish to find the most probable paths leading to each state by finding the two arguments s_{t-1}^a and s_{t-1}^b of the following maximization:

$$\begin{aligned} & \max_{s_{t-1}^a s_{t-1}^b} p(s_t^a|s_{t-1}^a)p(s_t^b|s_{t-1}^b)p(s_{t-1}^a, s_{t-1}^b|y_{1..t-1}) \\ = & \max_{s_{t-1}^a} p(s_t^a|s_{t-1}^a) \max_{s_{t-1}^b} p(s_t^b|s_{t-1}^b)p(s_{t-1}^a, s_{t-1}^b|y_{1..t-1}). \end{aligned}$$

For each state s_t^b, we first compute the inner maximum over s_{t-1}^b as a function of s_{t-1}^a, and store the max value and its argument. Then we compute, for each state s_t^a and s_t^b, the outer maximum over s_{t-1}^a, using the inner max evaluated at s_{t-1}^a. Finally, we look up the stored argument, s_{t-1}^b, of the inner maximization evaluated at the max s_{t-1}^a, for each state s_t^a and s_t^b. Again we require $O(D^3)$ operations with $O(D^2)$ storage for each step. In general, as with the forward-backward algorithm, the N-dimensional Viterbi search requires $O(TND^{N+1})$ operations.

We can also exploit the sparsity of the transition matrices and observation likelihoods, by pruning unlikely values. Using both of these methods our implementation of 2D Viterbi search is faster than the acoustic likelihood computation that serves as its input, for the model sizes and grammars chosen in the speech separation task.

3.4. Methods of Inference

In our experiments we performed inference in three different conditions: without dynamics, with acoustic dynamics, and with acoustic and grammar dynamics. Without dynamics the source models reduce to GMMs and we infer MMSE estimates of the sources based on $p(x^a, x^b|y)$ as computed analytically from (2) via Algonquin as discussed in section 2.1.

In the acoustic dynamics condition, the exact inference algorithm uses the 2D Viterbi search, with acoustic temporal constraints $p(s_t|s_{t-1})$ and likelihoods from Eqn. (2), to find the most likely joint state sequence $s_{1..T}$.

In the grammar dynamics condition we use the model of section 3.2. Exact inference is computationally complex because the full joint distribution of the grammar and acoustic states, $(v^a \times s^a) \times (v^b \times s^b)$ is required and is very large in number.

Instead we perform approximate inference by alternating the 2D Viterbi search between two factors: the Cartesian product $s^a \times s^b$ of the acoustic state sequences and the Cartesian product $v^a \times v^b$ of the grammar state sequences. When evaluating each state sequence we hold the other chain constant, which decouples its dynamics and allows for efficient inference.

This is a useful factorization because the states s^a and s^b interact strongly with each other and similarly for v^a and v^b. In fact, in the same-talker condition the corresponding states exhibit an exactly symmetrical distribution. The 2D Viterbi search breaks this symmetry on each factor. Details of various alternative approximate inference strategies for this model will be explored in future publications.

Once the maximum likelihood joint state sequence is found we can infer the source log-power spectrum of each signal and reconstruct them as shown in [3].

4. Speaker Identification and Gain Estimation

In the challenge task, the gains and identities of the two speakers were unknown at test time and were selected from a set of 34 speakers which were mixed at SNRs ranging from 6dB to -9dB. We used speaker-dependent acoustic models because of their advantages when separating different speakers. These models were trained on data with a narrow range of gains, so it is necessary to match the models to the gains of the signals at test time. This means that we have to estimate both the speaker identities and their gains in order to successfully infer the source signals.

However, the number of speakers and range of SNRs in the test set makes it too expensive to consider every possible combination of models and gains. Furthermore we found that the optimal gain, in the sense of maximum likelihood under our models, differed significantly from the nominal gains in the test set. Hence we developed an efficient model-based method for identifying the speakers and estimating the gains.

The algorithm is based upon a simple idea: identify and utilize frames that are dominated by a single source to determine what sources are present in the mixture. The output of this stage is a short list of candidate speaker IDs and associated gain estimates. We then estimate the posterior probability of combinations of these candidates and refine the estimates of their respective gains via an approximate EM procedure. In this EM procedure we use the max model of the source interaction likelihood mentioned in section 2.2.

To identify frames dominated by a single source, we model the signal for each processing frame t as generated from a single source class c, and assume that each source class is described by a mixture model:

$$p(\boldsymbol{y}_t|c) \quad = \quad \sum_g \sum_{s^c} \pi_{s^c} \pi_g \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{\mu}_{s^c} + g, \boldsymbol{\Sigma}_{s^c}) \qquad (3)$$

where the gain parameter g takes a range of discrete values $\{6, 3, 0, -3, -6, -9, -12\}$ with prior π_g, and π_{s^c} is the prior probability of state s in source class c. Although not all frames are in fact dominated by only one source, such a model will tend to ascribe greater likelihood to the frames that are dominated by one source. The mixture of gains allows the model to be gain-independent at this stage.

To form a useful estimate of $p(c|\boldsymbol{y})$ we apply the following simple algorithm:

1. Compute the normalized likelihood of c given $\boldsymbol{y}_t$ for each frame
$$b_{\boldsymbol{y}_t}(c) = p(\boldsymbol{y}_t|c) / \sum_{c'} p(\boldsymbol{y}_t|c'). \qquad (4)$$

2. Approximate the component class likelihood by
$$p(\boldsymbol{y}|c) = \sum_t \phi(b_{\boldsymbol{y}_t}(c)) \cdot b_{\boldsymbol{y}_t}(c), \qquad (5)$$
where $\phi(b_{\boldsymbol{y}_t}(c))$ is a confidence weight that is assigned based on the structure of $b_{\boldsymbol{y}_t}(c)$, defined here as
$$\phi(b_{\boldsymbol{y}_t}(c)) = \begin{cases} 1 & \max_c b_{\boldsymbol{y}_t}(c) > \gamma \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$
where γ is a chosen threshold.

3. Compute the source class posterior as usual via:
$$p(c|\boldsymbol{y}) \propto p(\boldsymbol{y}|c)p(c)$$

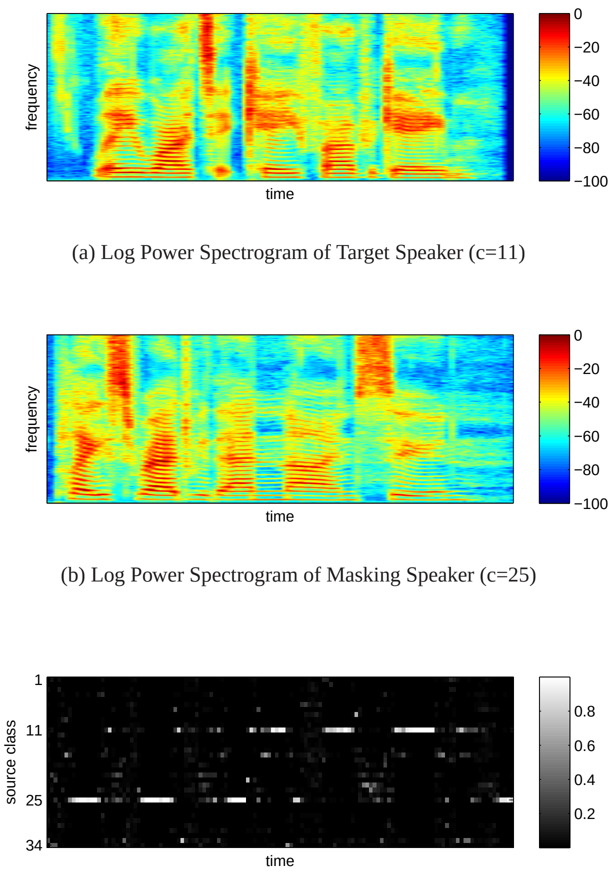
This method for estimating $p(c|\boldsymbol{y})$ is useful in situations where there may be many frames that are not dominated by a single source. In (5) the normalized likelihoods are summed rather than multiplied, because the observations may be unreliable. For instance, in many frames the model will assign a likelihood of nearly zero, even though the source class is present in the mixture. The confidence weight $\phi(b_{\boldsymbol{y}_t}(c))$ in (5) also favors frames that are well described by a single component, that is, where the likelihood $b_{\boldsymbol{y}_t}(c)$ is high for some component c. Frames that do not have this property might be misleading if they constitute an overwhelming majority.

Figure 3 depicts plots of the original spectrograms of the target and masker speakers along with the normalized likelihoods $b_{\boldsymbol{y}_t}(c)$ plotted as a function of t, for a typical test mixture in the SSC two-talker corpus. From the plots we can see that the likelihood functions $b_{\boldsymbol{y}_t}(c)$ are sharply peaked in regions of the mixture where one source dominates.

Given a short-list of finalists chosen according to $p(c|\boldsymbol{y})$ as computed above, we identify the present source components by applying the following max-based approximate EM algorithm to find the gains and identify the most probable speaker combination:

1. E-Step: Compute $p_i(s^j_t, s^k_t|\boldsymbol{y}_t)$ for all t using the max approximation (See section 2.2), in iteration i, for a hypothesis of speaker IDs j and k.

2. M-Step: Estimate $\triangle g_{j,i}$ via:
$$\triangle g_{j,i} = \alpha_i \frac{\sum_t \sum_{s^j_t, s^k_t} p_i(s^j_t, s^k_t|\boldsymbol{y}_t) \sum_{d \in D_{s^j_t|s^k_t}} \frac{\triangle g_{j,k,d,t}}{\sigma^2_{s^j_t, s^k_t, d}}}{\sum_t \sum_{s^j_t, s^k_t} p_i(s^j_t, s^k_t|\boldsymbol{y}_t) \sum_{d \in D_{s^j_t|s^k_t}} \frac{1}{\sigma^2_{s^j_t, s^k_t, d}}} \qquad (7)$$



(a) Log Power Spectrogram of Target Speaker (c=11)

(b) Log Power Spectrogram of Masking Speaker (c=25)

(c) Source class likelihoods $b_{\boldsymbol{y}_t}(c)$

Figure 3: Plots of the (unobserved) spectrograms of the target and masker speakers and the computed source class frame likelihoods $b_{\boldsymbol{y}_t}(c)$ (4), for a typical test utterance in the SSC two-talker corpus (mixed at 0 dB). From the plots we can see that the (normalized) source likelihoods are sharply peaked in regions of the mixture where one source dominates.

> where $\triangle g_{j,k,d,t} = (y_{d,t} - \mu_{s_t^j,s_t^k,d} - g_{j,i-1})$, $D_{s_t^j|s_t^k}$ is all dimensions where $\mu_{s_t^j,d} - g_{j,i-1} > \mu_{s_t^k,d} - g_{k,i-1}$, and α_i is a learning rate.

Note that the probability of the data is not guaranteed to increase at each iteration of this EM procedure even when $\alpha_i = 1$, because the joint state posterior $p_i(s^j, s^k|\boldsymbol{y}_t)$ is not continuous in $g_{j,i}$ and $g_{k,i}$: the dimension assignment $D_{s^j|s^k}$ changes depending on the current gain estimate. Empirically however, this approach has proved to be effective.

Table 1 reports the speaker identification accuracy obtained on the SSC two-talker test set via this approach, when all combinations of the most probable source and the six most probable sources are considered (six combinations total), and the speaker combination maximizing the probabilility of the data is selected. Over all mixture cases and conditions on the SSC two-talker test set we obtained greater than 98% speaker identification accuracy overall.

	6 dB	3 dB	0 dB	-3 dB	-6 dB	-9dB	All
ST	100	100	100	100	100	99	99
SG	97	98	98	97	97	96	97
DG	99	99	98	98	97	96	98
All	99	99	99	98	98	97	98

Table 1: Speaker identification accuracy (percent) as a function of test condition and case on the SSC two-talker test set, for the presented source identification and gain estimation algorithm. ST-Same Talker, SG-Same Gender, DG-Different Gender.

5. Recognition using Speaker Dependent Labeling (SDL)

Once the two signals have been separated, we decode each of the signals with a speech recognition system that incorporates SDL.

We employed MAP training [12] to train speaker dependent models for each of the 34 speakers. The Speech Separation Challenge also contains a stationary colored noise condition, which we used to test the noise-robustness of our recognition system. The performance obtained using MAP adapted speaker dependent models with the baseline gender dependent labeling system (GDL) and SDL are shown in Table 2. As we can see the SDL technique (described below) achieves better results than the MAP adapted system using oracle knowledge of the speaker id.

5.1. Theory of SDL

Instead of using the speaker identities provided by the speaker ID and gain module directly in the recognizer, we followed the approach for gender dependent labeling (GDL) described in [13].

Each speaker c is associated with a set, $\mathcal{S}_c$, of 39 dimensional cepstrum domain acoustic Gaussian mixture models. At a particular time frame then we have the following estimate of the *a posteriori* speaker probability given the speech feature $\mathbf{x}_t$:

$$p(c_t|\mathbf{x}_t) = \frac{\sum_{s\in\mathcal{S}_c} \pi_s \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)}{\sum_{c'} \sum_{s\in\mathcal{S}_{c'}} \pi_s \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)}.$$

SDL does not make the assumption that each file contains only one speaker, but instead assumes only that the speaker identity is constant for a short time, and that the observations are unreliable. The speaker probability is thus averaged over a time window using the following recursive formula:

$$p(c_t|\mathbf{x}_{1:t}) \overset{\text{def}}{=} \alpha p(c_{t-1}|\boldsymbol{x}_{1:t-1}) + (1-\alpha) p(c_t|\mathbf{x}_t) \qquad (8)$$

for speaker c at time t, and where α is a time constant. This is equivalent to smoothing the frame-based speaker posteriors using the following exponentially decaying time window.

$$p(c_t|\mathbf{x}_{1:t}) = \sum_{t'=1}^{t} (1-\alpha)\alpha^{t-t'} p(c_{t'}|\mathbf{x}_{t'}), \qquad (9)$$

The effective window size for the speaker probabilities is given by $\alpha/(1-\alpha)$, and can be set to match the typical duration of each speaker. We chose $\alpha/(1-\alpha) = 100$, corresponding to a speaker duration of 1.5s.

Equation (8) can also be interpreted as forward inference in a model that consists of a probabilistic mixture of two conditions at each time point. The first term corresponds to the assumption that the observation $\mathbf{x}_t$ is unreliable and the speaker id c_t is the same as the previous time step. The second term corresponds to the assumption that the observation is reliable and the speaker id c_t is independent of the previous time step. The value α represents the prior probability of each condition at each time step. Such a system can be more robust than a system that simply assumes the speaker is unlikely to change over time.

The online *a posteriori* speaker probabilities are close to uniform even when the correct speaker is the one with the highest probability. We can remedy this problem by sharpening the probabilities to look more like 0-1 probabilities. The boosted speaker detection probabilities are defined as

$$\pi_{c_t} = p(c_t|\mathbf{x}_{1:t})^{\beta} / \sum_{c'} p(c'_t|\mathbf{x}_{1:t})^{\beta}. \qquad (10)$$

We used $\beta = 6$ for our experiments. During decoding we can now use the boosted speaker detection probabilities to give a time-dependent Gaussian mixture distribution:

$$\mathrm{GMM}(\mathbf{x}_t) = \sum_c \pi_{c_t} \mathrm{GMM}_c(\mathbf{x}_t).$$

As can be seen in Table, 2 the SDL system outperforms the oracle system [3].

System	Noise Condition				
	clean	6dB	0dB	-6dB	-12dB
HTK	***1.0***	45.7	82.0	88.6	87.2
GDL-MAP I	2.0	33.2	68.6	85.4	87.3
GDL-MAP II	2.7	7.6	14.8	49.6	77.2
oracle	1.1	4.2	8.4	39.1	***76.4***
SDL	1.4	***3.4***	***7.7***	***38.4***	77.3

Table 2: Word error rates (percent) on the SSC stationary noise development set. The error rate for the "random-guess" system is 87%. The systems in the table are: 1) The default HTK recognizer, 2) IBM–GDL MAP–adapted to the speech separation training data, 3) MAP–adapted to the speech separation training data and artificially generated training data with added noise, 4) Oracle MAP adapted Speaker dependent system with known speaker IDs, 5) MAP adapted speaker dependent models with SDL.

6. Experiments and Results

The Speech Separation Challenge [1] involves separating the mixed speech of two speakers drawn from of a set of 34 speakers. An example utterance is *place white by R 4 now*. In each recording, one of the speakers says *white* while the other says *blue*, *red* or *green*. The task is to recognize the letter and the digit of the speaker that said *white*.

Using the SDL recognizer, we decoded the two component signals under the assumption that one signal contains white and the other does not, and vice versa. We then used the association that yielded the highest combined likelihood.

Log-power spectrum features were computed at a 15 ms rate. Each frame was of length 40 ms and a 640 point FFT was used, and the DC component was discarded, producing a 319-dimensional log-power-spectrum feature vector.

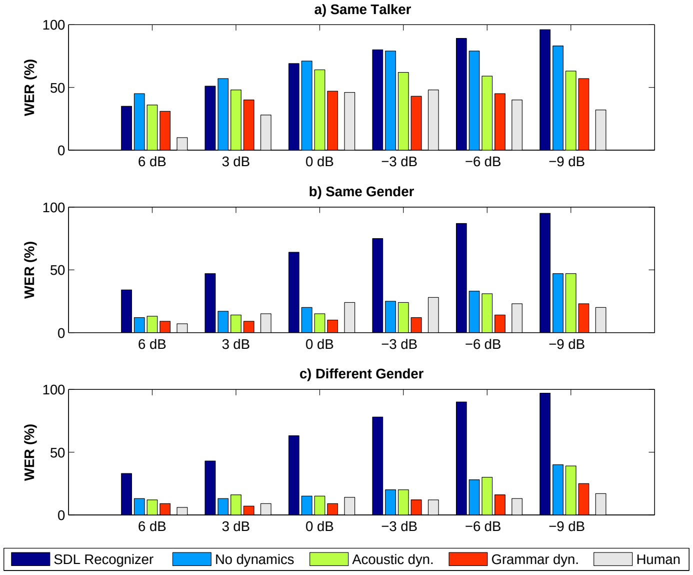


Figure 4: Word error rates for the a) Same Talker, b) Same Gender and c) Different Gender cases. All results were decoded using the SDL recognizer.

	6 dB	3 dB	0 dB	-3 dB	-6 dB	-9dB	total
ST	31	40	47	***43***	45	57	43.8
SG	9	***9***	***10***	***12***	***14***	23	***12.9***
DG	9	***7***	***9***	***12***	16	25	12.9
All	17.3	19.8	***23.3***	***23.2***	***25.9***	36.1	24.3

Table 3: Word error rates (percent) for grammar and acoustic constraints. ST-Same Talker, SG-Same Gender, DG-Different Gender. Conditions where our system performed as well or better than human listeners are emphasized.

Figure 4 shows results for the: a) Same Talker, b) Same Gender, and c) Different Gender conditions. Human listener performance [1] is shown along with the performance of the SDL recognizer applied to: 1) the unprocessed mixed features, and the signals obtained from the separation system 2) without dynamics 3) using acoustic level dynamics, and 4) using both grammar and acoustic level dynamics.

The top plot in Figure 4 shows word error rates (WER) for the *Same Talker* condition. In this condition, two recordings from the same speaker are mixed together. This conditions best illustrates the importance of temporal constraints. By adding the acoustic dynamics, performance is improved considerably. By combining grammar and acoustic dynamics, performance improves again, surpassing human performance in the -3 dB condition.

The second plot in Figure 4 shows WER for the Same Gender condition. In this condition, recordings from two different speakers of the same gender are mixed together. In this condition our

[3] No prior knowledge of the speaker ID or noise condition was used in generating the results (save the oracle system).

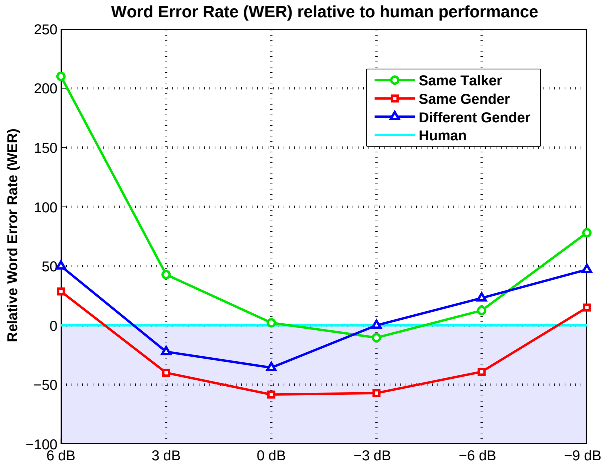


Figure 5: Word error rate of system relative to human performance. Shaded area is where the system outperforms human listeners.

system surpasses human performance in all conditions except 6 dB and -9 dB.

The third plot in Figure 4 shows WER for the Different Gender condition. In this condition, our system surpasses human performance in the 0 dB and 3 dB conditions. Interestingly, temporal constraints do not improve performance relative to GMM without dynamics as dramatically as in the same talker case, which indicates that the characteristics of the two speakers in a short segment are effective for separation.

The performance of the Iroquois model, which uses both grammar and acoustic-level dynamics, is summarized in Table 3. This system surpassed human lister performance at SNRs of 0 dB to -6 dB on average across all speaker conditions. Averaging across all SNRs, the Iroquois model surpassed human performance in the Same Gender condition. Based on these initial results, we envision that super-human performance over all conditions is within reach.

7. Discussion

The absolute performance of human listeners is shown in Figure 4. As expected, human listeners perform well when the amplitude of target speaker is considerably higher than the masker. Surprisingly, human listeners also perform well when the target speaker is speaking at a lower amplitude than the masker. Human subjects perform worst when the speakers are at a similar amplitude. Figure 5 shows the relative Word Error Rate (WER) of our system compared to human subjects. The same general trend can be seen all three cases (Same Talker, Same Gender and Different Talker). The system performs poorly compared to human subjects when the target speaker is relatively strong. This is to be expected since state of the art ASR systems cannot match human performance for letter recognition.

However, the Iroquois model performs relatively well when the amplitude of the signals is similar. Remarkably, in the *Same Gender* condition, the system is up to 50% better than human subjects. It seems that the human auditory system employs different cues and strategies for accomplishing recognition in these conditions. Perhaps human listeners are better able to make use of differences in amplitude as a cue for separation.

It is our hope that further experiments with both human and machine listeners will provide us with a better understanding of the differences in their performance characteristics. This may provide insights into how the human auditory system functions, as well as how automatic speech recognition can be brought to human levels of performance.

8. References

[1] Martin Cooke and Tee-Won Lee, “Interspeech speech separation challenge,” `http : //www.dcs.shef.ac.uk/ ∼ martin/SpeechSeparationChallenge.htm`, 2006.

[2] T. Kristjansson, J. Hershey, P. Olsen, S. Rennie, and R. Gopinath, “Super-human multi-talker speech recognition: The ibm 2006 speech separation challenge system,” in *ICSLP*, Pittsburgh, PA, U.S.A., 2006, in press.

[3] T. Kristjansson, J. Hershey, and H. Attias, “Single microphone source separation using high resolution signal reconstruction,” *ICASSP*, 2004.

[4] P. Varga and R.K. Moore, “Hidden Markov model decomposition of speech and noise,” *ICASSP*, pp. 845–848, 1990.

[5] M. Gales and S. Young, “Robust continuous speech recognition using parallel model combination,” *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 352–359, September 1996.

[6] Y. Ephraim, “A Bayesian estimation approach for speech enhancement using hidden Markov models.,” vol. 40, no. 4, pp. 725–735, 1992.

[7] Sam T. Roweis, “One microphone source separation.,” in *NIPS*, 2000, pp. 793–799.

[8] John Hershey and Michael Casey, “Audio-visual sound separation via hidden Markov models.,” in *NIPS*, 2001, pp. 1173–1180.

[9] E. Bocchieri, “Vector quantization for the efficient computation of continuous density likelihoods. proceedings of the international conference on acoustics,” in *ICASSP*, Minneapolis, MN, U.S.A., April 1993, IEEE, vol. II, pp. 692–695.

[10] S. Roweis, “Factorial models and refiltering for speech separation and denoising,” *Eurospeech*, pp. 1009–1012, 2003.

[11] Zoubin Ghahramani and Michael I. Jordan, “Factorial hidden Markov models,” in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, Eds. 1995, vol. 8, pp. 472–478, MIT Press.

[12] J.-L. Gauvain and C.-H. Lee, “Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains,” *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[13] Peder Olsen and Satya Dharanipragada, “An efficient integrated gender detection scheme and time mediated averaging of gender dependent acoustic models,” in *Eurospeech 2003*, Geneva, Switzerland, September 1-4 2003, vol. 4, pp. 2509–2512.

9. Acknowledgements

We would like to thank IBM T.J. Watson Research Center and in particular Ramesh Gopinath for hosting this research and providing thoughtful discussion. In addition, we are grateful to the organizers if the Speech Separation Challenge, Martin Cooke and Te-Won Lee, as well as the other participants for stimulating a fascinating area of research.