Eigenvoice Speaker Adaptation via Composite Kernel PCA

James Kwok Brian Mak Simon Ho

Department of Computer Science
Hong Kong University of Science and Technology
Hong Kong

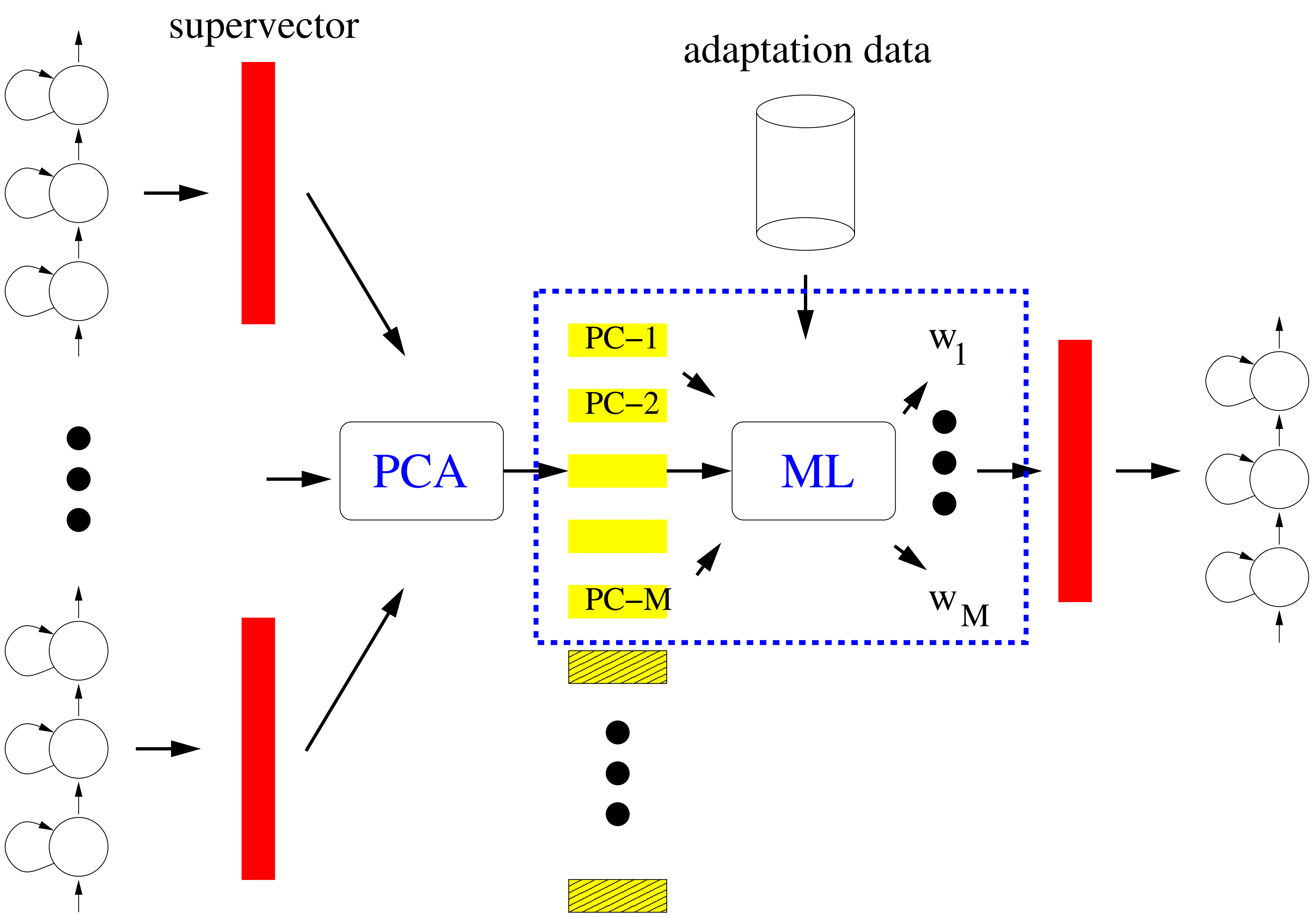
Speaker Adaptation

- A well-trained speaker-dependent (SD) model generally achieves a significantly lower word error rate than a speaker-independent (SI) model on recognizing speech from the specific speaker

- Hard to acquire a large amount of data from a user to train the SD model
  - adapt the SI model with a relatively small amount of SD speech
    - *maximum a posteriori* (MAP) adaptation
    - maximum likelihood linear regression (MLLR) adaptation
  - when the amount of available adaptation speech is really small (e.g., only a few seconds): eigenvoice-based adaptation

Eigenvoice vs Kernel Eigenvoice

- Eigenvoice (EV)
  - use principal component analysis (PCA) to find the eigenvoices
  - represent the new speaker as a linear combination of the leading eigenvoices
  - estimate the (small) set of weights by using maximum likelihood
  - linear PCA $\rightarrow$ captures only linear relationships
- Kernel eigenvoice (KEV)
  - kernel PCA
  - issues:
    - do all computations rely only on kernel evaluations?
    - how to compute the observation likelihood?


supervector
adaptation data
PCA
PC−1
PC−2
PC−M
ML
w_1
w_M

Eigenvoice: Training

- A set of speaker-dependent (SD) acoustic hidden Markov models (HMMs) are trained from each speaker
  - in general, the HMM states are GMMs
- A speaker's voice is represented by a speaker supervector that is composed by concatenating the mean vectors of all his HMM Gaussian distributions
  - R states in each HMM
  - $\mathbf{x}_i = [\mathbf{x}'_{i1}, \ldots, \mathbf{x}'_{iR}]'$
- PCA is then performed on a set of training speaker supervectors and the resulting eigenvectors are called eigenvoices

Eigenvoice: Adaptation

- The new speaker's supervector $\mathbf{s}$ is assumed to be a linear combination of the M leading eigenvoices $\{\mathbf{v}_1, \ldots, \mathbf{v}_M\}$

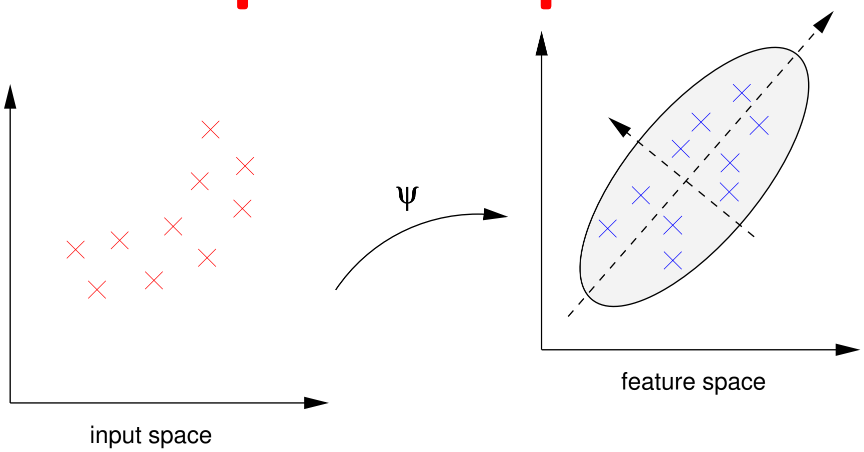
$$\mathbf{s} = \mathbf{s}^{(ev)} = \sum_{m=1}^{M} w_m \mathbf{v}_m$$

- Given the adaptation data $\boldsymbol{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T\}$, estimate the eigenvoice weights ($\mathbf{w} = [w_1, \ldots, w_m]'$) by maximum likelihood

$$\max_{\mathbf{w}} \; Q(\mathbf{w}) \equiv -\frac{1}{2} \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r) \|\mathbf{o}_t - \mathbf{s}_r(\mathbf{w})\|^2_{\mathbf{C}_r}$$

  - $\gamma_t(r)$: posterior probability of observation sequence being at state r at time t
  - $\mathbf{C}_r$: covariance matrix of the Gaussian at state r
  - $\mathbf{s}_r$: rth constituent of $\mathbf{s}$

Kernel Principal Component Analysis



- Kernel PCA: linear PCA in the feature space
- Given $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, construct $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)] = [\varphi(\mathbf{x}_i)'\varphi(\mathbf{x}_j)]$
- $\mathbf{K} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}'$ (assume that $\{\varphi(\mathbf{x}_1), \ldots, \varphi(\mathbf{x}_N)\}$ has been centered)
  - $\mathbf{U} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N]$ with $\boldsymbol{\alpha}_i = [\alpha_{i1}, \ldots, \alpha_{iN}]'$
  - $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_N)$
- kth orthonormal eigenvector: $\mathbf{v}_k = \sum_{i=1}^{N} \frac{\alpha_{ki}}{\sqrt{\lambda_k}} \varphi(\mathbf{x}_i)$

Problem

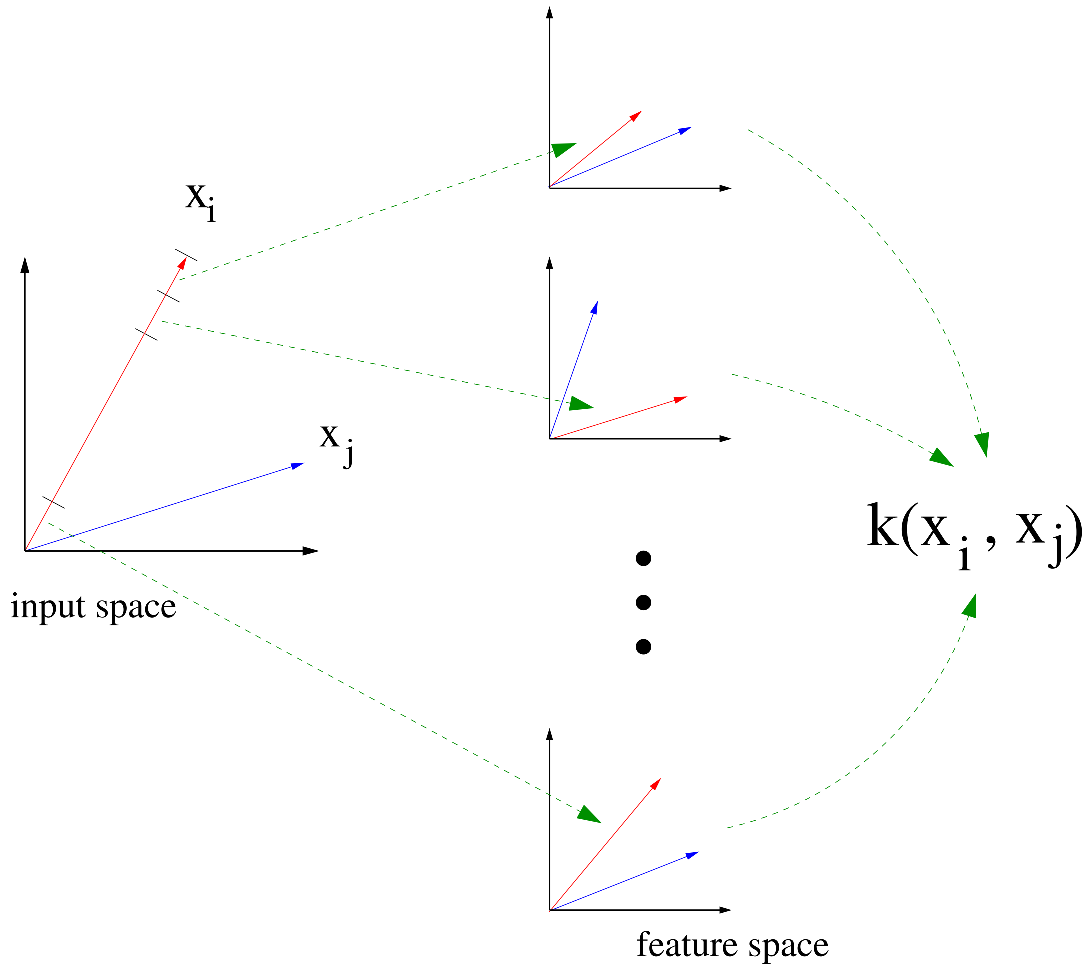
- Estimation of the eigenvoice weights requires the evaluation of the distances between adaptation data $\mathbf{o}_t$ and Gaussian means of the new speaker in the observation space

- EV: breaks up the speaker-adapted (SA) model found by EV adaptation into its constituent HMM Gaussians
  - $\mathbf{s}^{(ev)} \rightarrow \mathbf{s}_1^{(ev)}, \ldots, \mathbf{s}_R^{(ev)} \rightarrow$ Gaussian means

- KEV: the SA model found by KEV adaptation resides in the feature space, not in the input speaker supervector space
  - cannot access each constituent Gaussian directly

Composite Kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = f(k_1(\mathbf{x}_{i1}, \mathbf{x}_{j1}), \ldots, k_R(\mathbf{x}_{iR}, \mathbf{x}_{jR}))$$



Examples

- Direct sum kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sum_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr})$$

  – corresponding feature: $\varphi(\mathbf{x}_i) = [\varphi_1(\mathbf{x}_{i1})', \ldots, \varphi_R(\mathbf{x}_{iR})']'$

- Tensor product kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \prod_{r=1}^{R} k_r(\mathbf{x}_{ir}, \mathbf{x}_{jr})$$

- If $k_r(\cdot, \cdot)$'s are valid Mercer kernels, so is $k(\cdot, \cdot)$

New Speaker in the Feature Space

$$\varphi(\mathbf{s}) = \sum_{m=1}^{M} w_m \mathbf{v}_m = \sum_{m=1}^{M} \sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \varphi(\mathbf{x}_i)$$

- rth constituent: $\varphi_r(\mathbf{s}_r) = \sum_{m=1}^{M} \sum_{i=1}^{N} \frac{w_m \alpha_{mi}}{\sqrt{\lambda_m}} \varphi_r(\mathbf{x}_{ir})$
- Similarity between $\varphi_r(\mathbf{s}_r)$ and $\varphi_r(\mathbf{o}_t)$:

$$k_r(\mathbf{s}_r, \mathbf{o}_t) = \varphi_r(\mathbf{s}_r)' \varphi_r(\mathbf{o}_t) = A(r,t) + \sum_{m=1}^{M} \frac{w_m}{\sqrt{\lambda_m}} B(m,r,t)$$

  - $A(r,t) = \frac{1}{N} \sum_{j=1}^{N} k_r(\mathbf{x}_{jr}, \mathbf{o}_t)$
  - $B(m,r,t) = \left(\sum_{i=1}^{N} \alpha_{mi} k_r(\mathbf{x}_{ir}, \mathbf{o}_t) \right) - A(r,t) \left(\sum_{i=1}^{N} \alpha_{mi} \right)$

Maximum Likelihood Adaptation

- $k_r(\cdot,\cdot)$: e.g., isotropic kernels $k_r(\mathbf{s}_r, \mathbf{o}_t) = \kappa(\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r})$
  - e.g., Gaussian kernels: $k_r(\mathbf{s}_r, \mathbf{o}_t) = \exp(-\beta\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r})$
  - if κ is invertible, $\|\mathbf{o}_t - \mathbf{s}_r\|^2_{\mathbf{C}_r} \longrightarrow$ function of $k_r(\mathbf{s}_r, \mathbf{o}_t) \longrightarrow$ function of $\mathbf{w}$

- Substitute back to $Q(\mathbf{w})$ and differentiate to obtain $\partial Q / \partial w_j$

- No closed form solution for the optimal $\mathbf{w}$
  - use generalized EM algorithm (GEM)

- $\mathbf{w}(0)$: eigenvoice weights of the supervector composed from the speaker-independent model $\mathbf{x}^{(si)}$
  - $\mathbf{w}_m(0) = \mathbf{v}'_m \varphi(\mathbf{x}^{(si)})$ (can be obtained from kernel evaluations)

Incorporate the SI Model

- Interpolate $\varphi(\mathbf{s})$ with the φ-mapped SI supervector $\varphi(\mathbf{x}^{(si)})$ to obtain the final SA model (in the feature space):

$$\varphi^{(rkev)}(\mathbf{s}) = w_0\varphi(\mathbf{x}^{(si)}) + (1 - w_0)\varphi(\mathbf{s}), \quad 0 \leq w_0 \leq 1$$

  – w_0 estimated in the same manner as the other w_m's
  – robust kernel eigenvoice

- $\varphi^{(rkev)}(\mathbf{s})$ contains components in $\varphi(\mathbf{x}^{(si)})$ from eigenvectors beyond the M selected kernel eigenvoices for adaptation

  – preserve the speaker-independent projections on the remaining less important but robust eigenvoices in the final speaker-adapted model

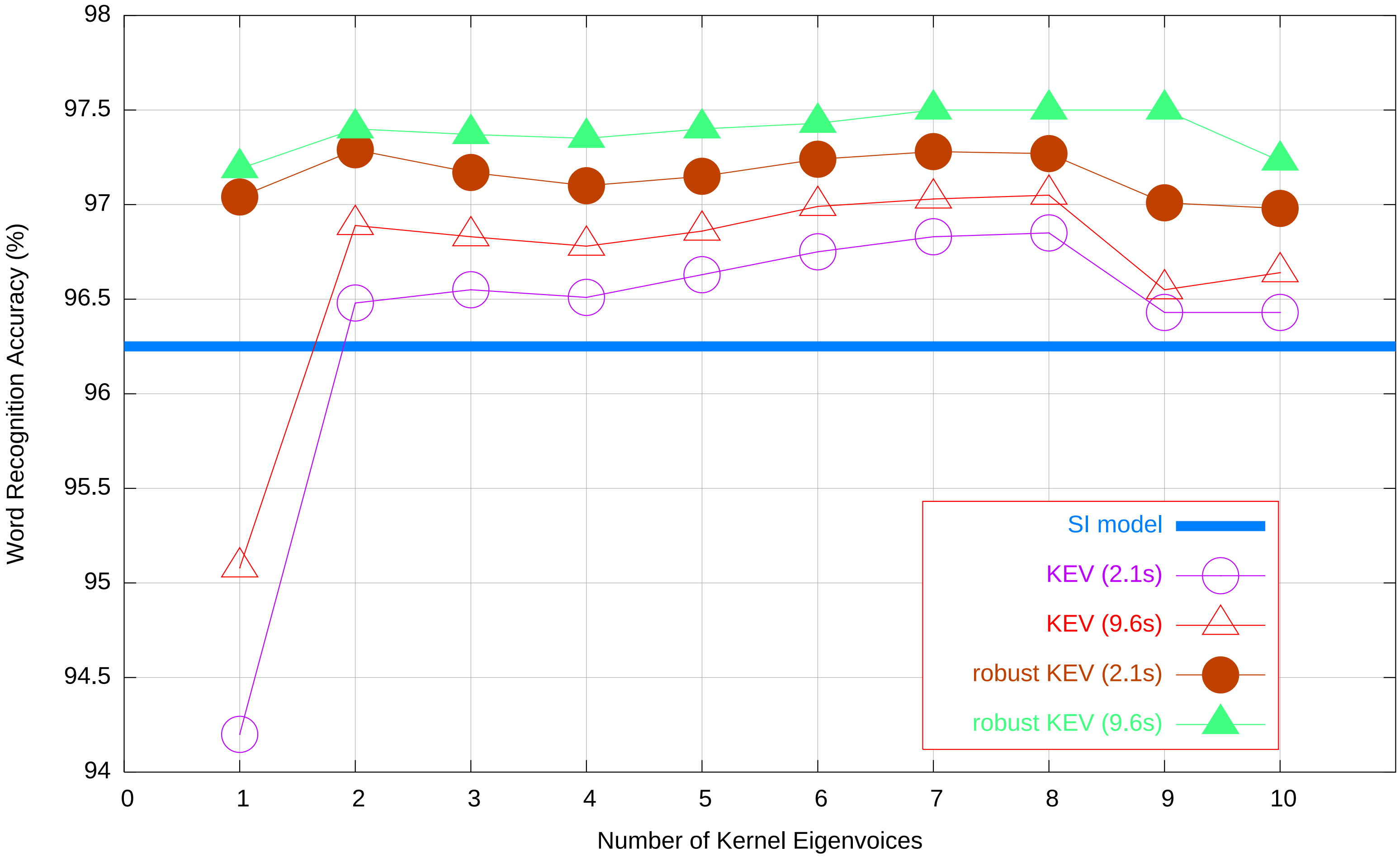
Experimental Setup: Data Set and HMM Models

- TIDIGITS corpus
  - 163 speakers (of both genders) in each (training and test) set, each pronouncing 77 utterances of 1-7 digits (out of: "0", "1", . . ., "9", and "oh")
- 12 mel-frequency cepstral coefficients and the normalized frame energy from each speech frame of 25 ms at every 10 ms
- Digit model
  - strictly left-to-right HMM with 16 states
  - one Gaussian with diagonal covariance per state
- A 3-state "sil" model to capture silence speech and a 1-state "sp" model to capture short pauses between digits

Adaptation

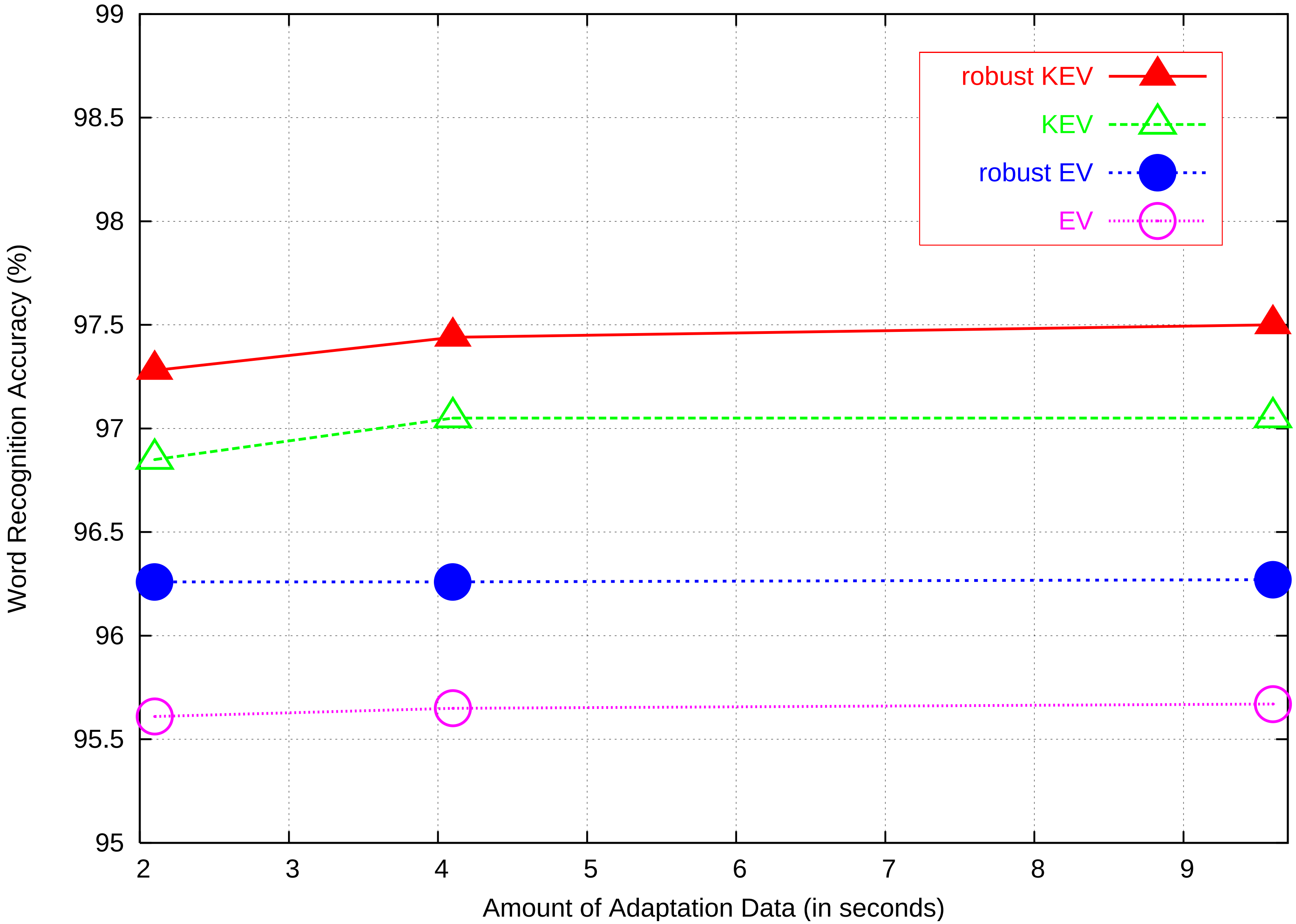
- SD digit model
  - one for each training speaker
  - variances and transition matrices are borrowed from SI models (only the Gaussian means are estimated)
- The "sil" and "sp" models are simply copied to the SD model
- 5, 10, 20 digits for adaptation ($\simeq$ 2.1s, 4.1s, and 9.6s of speech)
- Results are averages of 5-fold cross-validation over all test speakers
- (Testing) word accuracy of SI model: 96.25%

Experiment 1: Number of Kernel Eigenvoices



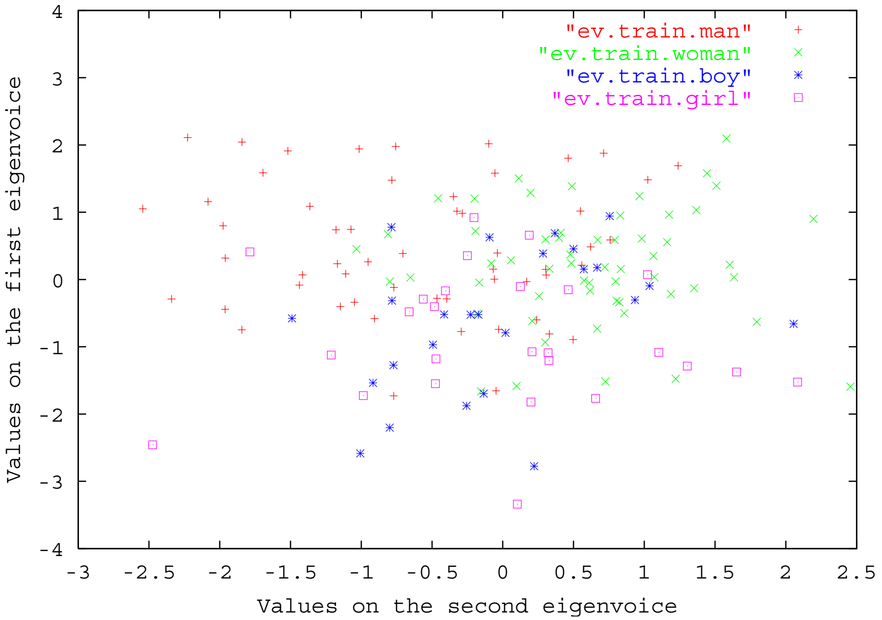
- KEV outperforms the SI model even with only two eigenvoices
- Robust KEV significantly improves KEV

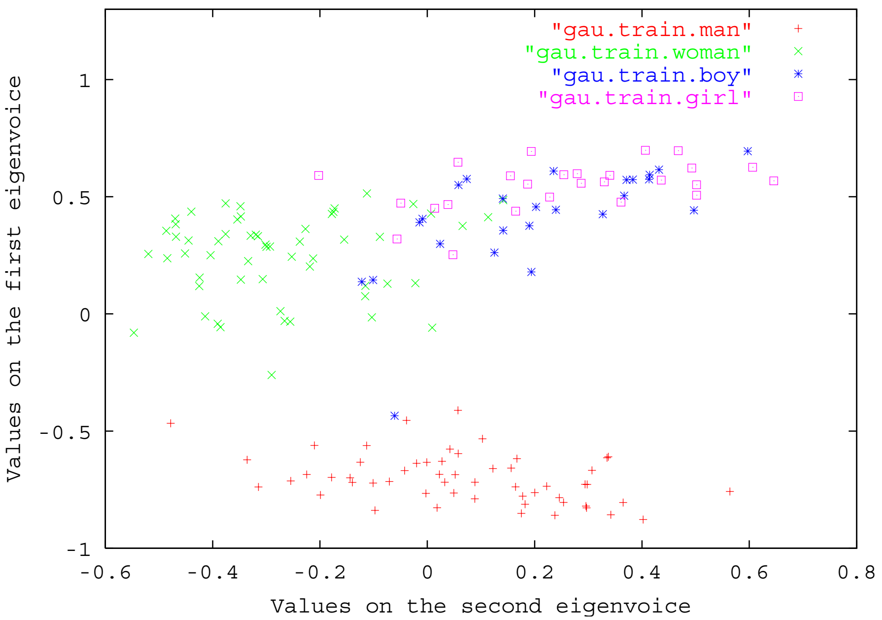
Experiment 2: KEV vs. EV



- (Robust) KEV always performs better than (robust) EV
- When only 2.1s or 4.1s of adaptation data are available
  $\text{EV} \simeq \text{MAP} \simeq \text{MLLR} < \text{SI} \simeq \text{robust EV} < \text{KEV} < \text{robust KEV}$
- With 9.6s of adaptation data
  - MLLR works marginally better than robust KEV (by an absolute 0.06%)
- Word error rate reduction over SI

	KEV	robust KEV
2.1s	16.0%	27.5%
4.1s	21.3%	31.7%
9.6s	21.3%	33.3%


"ev.train.man"
"ev.train.woman"
"ev.train.boy"
"ev.train.girl"
Values on the first eigenvoice
Values on the second eigenvoice
4
3
2
1
0
-1
-2
-3
-4
-3
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5


"gau.train.man"
"gau.train.woman"
"gau.train.boy"
"gau.train.girl"
Values on the first eigenvoice
Values on the second eigenvoice
1
0.5
0
-0.5
-1
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8

Conclusion and Future Work

- (Nonlinear) kernel PCA + composite kernel
  - better eigenvoices $\rightarrow$ improved speaker adaptation
- Interpolate the SI model with the speaker model found by KEV
- In the TIDIGITS task
  - standard EV does not help
  - KEV outperforms SI by 16–21% (word error rate reduction)
  - robust KEV: 28–33% word error rate reduction over SI
- Disadvantage: KEV is slower than EV
  - online computation of many kernel functions required during subsequent speech recognition
  - currently investigating speed-up techniques