Some experiments of LVQ training applied to mixture Gaussian HMMs.

Mikko Kurimo

See also:
Computer Speech and Language (1997),
Volume 11, Number 4, pages 321-343.

http://www.idealibrary.com/links/
doi/10.1006/csla.1997.0034/

Learning problems with large HMM systems

Attempts to improve the performance of models often crash into:

- too many parameters to estimate
- huge amount of samples to scan and learn
- training methods do not scale up well
- "the curse of dimensionality"
- the recognition speed might decline as well

Some special problems

- Segmental K-means tunes only the bmu ⇒ some units adapt (too) well and some are left over
- Embedded Baum-Welch adapts all units ⇒ computationally heavy and practical convergence difficulties
- The computation of gradients in most discriminative training methods is expensive and the required initial training complicates the process
- Some smoothing is required to prevent to accurate training data adaptation
- If the initialization is poor, it usually takes too long to converge into good results

What has SOM to offer?

- Suitable initialization for the mixtures
- Neighbor adaptation brings all the mixtures to effective areas
- The trade-off between smoothing and fitting accuracy is controlled by the width of the neighborhood
- By gradually reducing the width, the best density approximation accuracy occurs in the areas that get most hits by training samples
- Smoothing of the parameters occurs in a very natural way using the samples falling into nearby clusters
- Fast winner search methods used in the density function approximations can exploit the ordering of the mixtures

Segmental training by SOM

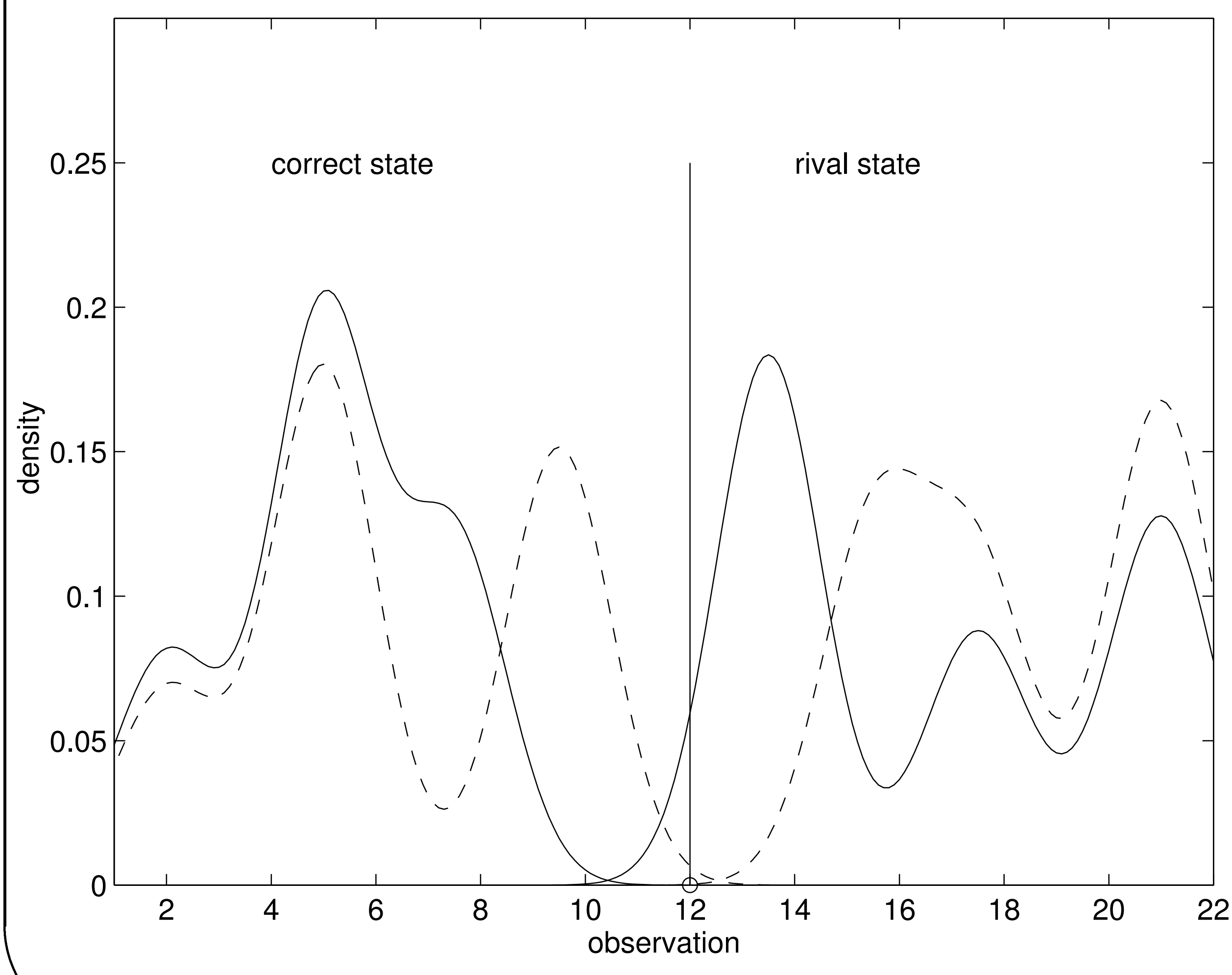
1. Train one SOM for each phoneme
2. Initialize the centroids of the Gaussian kernels and the state dependent weights using the obtained SOM units
3. Do the Viterbi segmentation as usual
4. Adapt the parameters using the associated set of samples by batch SOM and repeat from step 3. The difference to K-means is that *each data vector updates also the topological neighbors of the best matching mixture*
5. Fine tune the parameters with other training methods (SGPD, SLVQ), if necessary

Minimizing the error rate by segmental LVQ3

- For the best phoneme recognition accuracy the adaptation phase in the HMM training can be done by the segmental LVQ3
- Two rival segmentations (i.e. state paths) are computed for the data samples
- First one fits the known phonetic transcription, as usual, but the second one assumes it unknown.
- For feature vectors, where the phoneme labels of the two segmentations coincide, the state parameters are adapted to maximize the data likelihood, as usual.

Phoneme discrimination

- For frames, where the phoneme labels of the rival segmentations differ, the parameters are adapted to increase discrimination, as in the LVQ2
- The state on the correct path is tuned closer to the observed feature
- The corresponding state on the other path is tuned away from the observed feature



Segmental LVQ3

- Two rival phoneme decodings based on the best path for each are examined
- In SLVQ3 the states on the path for the given correct decoding are adapted as in SKM
- If the incorrect decoding is more likely, the states different from the correct path are adapted to lower the likelihood

Differences to GPD

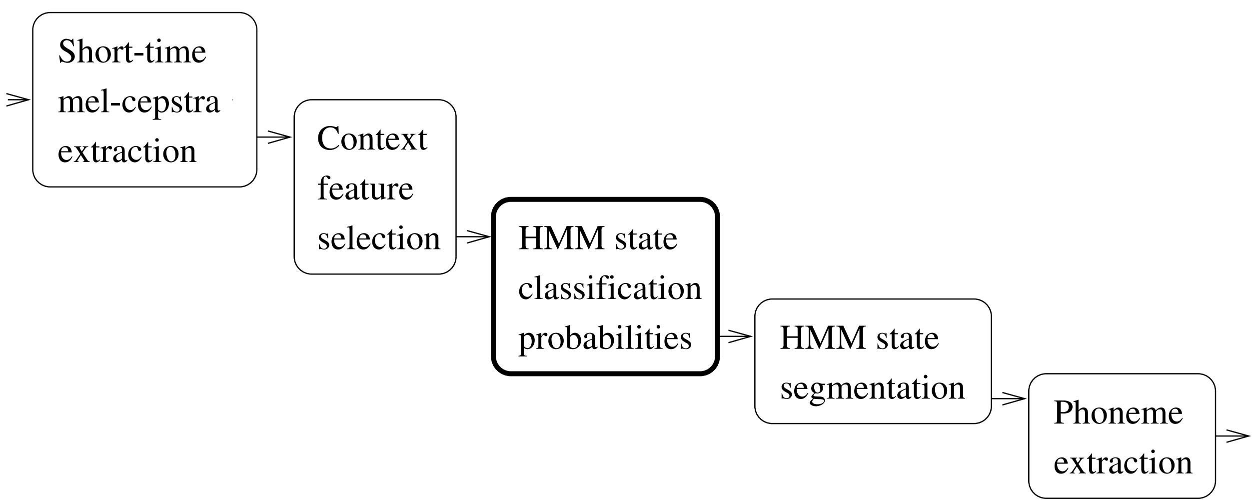
- Faster and more robust convergence is sought by decreasing state likelihoods only when it is absolutely necessary to avoid a misrecognition
- Unlike in GPD the adaptation step size does not depend directly on the extent of the misclassification of the whole path
- In general, robustness is sought in SLVQ3 by using as simple learning rules and as few control parameters as possible
- Segmental GPD:[W. Chou et. al., Proc. ICASSP, 1992]

Corrective tuning by LVQ2

- A pure corrective training algorithm
- The parameters of the states are modified stochastically in small steps after each incorrectly recognized feature.
- The learning rate decreases gradually
- Otherwise similar as the segmental LVQ3 without the likelihood maximization option
- A fine tuning method only applicable after the HMMs are already trained well by another method
- The error rate on data not used in training will eventually start to increase, if this method is used too many epochs.

Framework of experiments

- Finnish speech recognition for unlimited vocabulary
- Phoneme models using mixture density HMMs
- Using a large number of Gaussian mixtures with the help of SOM
- Minimization of recognition errors by applying LVQ training



Speech material

- Each speaker has dictated a list of 350 words on 4 different days
- The list is balanced to contain the most common phoneme combinations of the Finnish language
- The data is collected from 20 speakers
- The speaker-dependent models are trained by 3 word sets and tested on the remaining set
- Most of the results are given as an average error rate of 7 speakers
- For verification the most important results are computed as well for an older slightly different speech database of 3 speakers

Results

Init.	HMM training	Error rate 5 ep	Error rate 10 ep
KM	SKM	6.2	6.1
KM	SKM+SGPD		5.4
KM	SGPD	5.8	5.6
SOM	SSOM	5.9	5.5
SOM	SSOM+SGPD		5.1
SOM	SSOM+SLVQ3		5.3
SOM	SLVQ3	5.3	5.3
SOM	SLVQ3+SGPD		4.8

- 5 epochs by SLVQ3 gave the fewest errors in average
- The error rates did not improve significantly after 5th epoch except for method combinations (and for SSOM)
- Lowest rate was then obtained by SLVQ3+SGPD

Results for a larger model

Init.	HMM training	Error rate	
		5 ep	10 ep
KM	SKM	5.6	5.6
KM	SKM+SGPD		5.0
KM	SGPD	5.5	5.4
SOM	SSOM	5.2	4.9
SOM	SSOM+SGPD		5.5
SOM	SSOM+SLVQ3		5.0
SOM	SLVQ3	4.8	4.7
SOM	SLVQ3+SGPD		4.8

- 140 Gaussians per phoneme (instead of 70)
- The more detailed model did not drop the SGPD error rate as much as for the others (e.g. SSOM)
- Training after 5th epoch does not seem to give lower error rates than the SLVQ3
- 140 mixtures might be too much for this training data

Conclusions of SOM-LVQ tests

- The segmental LVQ3 seems to do best in this comparison test
- The combination of using first the (perhaps more robust) SLVQ3 and then the SGPD gave the lowest error rate
- The smoothness obtained by SOM training seems to help in training larger models
- A proper comparison between the methods would require several different databases, however
- Here, the averaged results on the Finnish database are used for a tentative ranking
- The obtained error rate can be much improved for practical recognition tasks