SEGMENTAL LVQ3 TRAINING FOR PHONEME-WISE TIED MIXTURE DENSITY HMMS

Mikko Kurimo
Helsinki University of Technology
Neural Networks Research Centre
Rakentajanaukio 2 C
FIN-02150 Espoo
Finland

email:Mikko.Kurimo@hut.fi

This talk

1. Segmental LVQ3 in brief
2. Motivation for this research
3. Important references
4. Details of the algorithm
5. About the phoneme model
6. Experimental results

Segmental LVQ3

1. Tunnistetaan opetusaineisto alustavilla foneemimalleilla
2. Jaetaan data foneemeita vastaaviin osiin (segmentointi)
3. Opetetaan malleja segmentoidulla aineistolla
4. Kullekin havainnolle lahinta oikeaa tn-jakaumaa lahemmas (SKM)
5. Myos lahinta vaaraa poispain, jos tuli tunnistusvirhe
6. Toistetaan parannetuilla malleilla

Motivation

- Segmental Kmeans training is in many ways better HMM training than the Baum-Welch reestimation
- LVQ initialization for mixture means [Kurimo94b] requires another method to complement the HMM training
- Segmental GPD [Chou92] is reported to work well, but needs another method to start the training
- LVQ in corrective tuning [Kurimo94a] improves the models, but is applicable only for fine tuning
- The phoneme-wise tied mixture density HMMs seem to favor phoneme-wise discrimination

References

- SOMs and LVQ:
  [Kohonen95] Teuvo Kohonen. *Self-Organizing Maps.* Springer, Berlin, 1995.

- SCHMMs:
  [Huang90] X.D. Huang and Y. Ariki and M.A. Jack. *HMMs for Speech Recognition.* Edinburgh University Press, Edinburgh, 1990.

- Segmental GPD:
  [Chou92] W.Chou and B.H. Juang and C.H. Lee. Segmental GPD Training of HMM Based Speech Recognizer. In *Proceedings of the ICASSP-92*, pages = 473–476, 1992.

References to my own work

Old

- LVQ tuning:
  [Kurimo94a] Mikko Kurimo. Corrective tuning by applying LVQ for continuous density and semi-continuous Markov models. In *Proceedings of International Symposium on Speech, Image Processing and Neural Networks*, pages = 718-721, 1994.

- TMHMM system:
  [Kurimo94b] Mikko Kurimo. Hybrid training method for tied mixture density hidden Markov models using Learning Vector Quantization and Viterbi estimation. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, pages 362-371, 1994.

References to my own work

New

- Segmental LVQ:
  [Kurimo96a] Mikko Kurimo. Segmental LVQ3 Training for Phoneme-wise Tied Mixture Density HMMs. In *Proceedings of the EUSIPCO-96*, 1996.

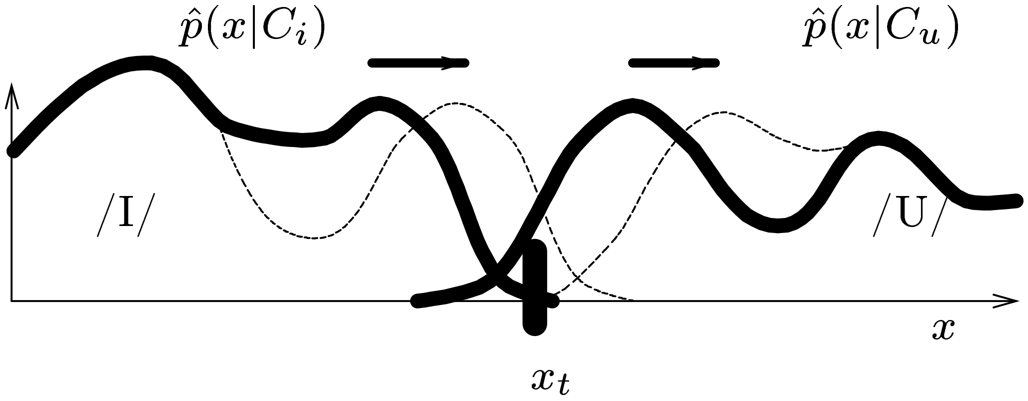
- Likelihood approximations using SOM:
  [Kurimo96b] Mikko Kurimo and Panu Somervuo. Using the Self-Organizing Map to Speed up the Probability Density Estimation for Speech Recognition with mixture density HMMs. In *Proceedings of the ICSLP-96*, 1996.

Segmental LVQ3 training

	Path	Phonemes
Best	KKKKEEEEEEEMMMMMMUUUUUUU	KEMU
Correct	KKKKEEEEEEEMMMMMMIIIIIII	KEMI

- A modification of the standard Viterbi training (SKM)
- Two segmentations: The Best ($\bar{q}$) and the Correct (q)
- If $\bar{q}$ gives wrong phonemes, find also q, else set $q = \bar{q}$

Segmental LVQ3 training



- Tune the state pdfs along q closer to observations ($\boldsymbol{x}_t$)
- If $\bar{q}$ is incorrect, tune state pdfs along $\bar{q}$ away from $\boldsymbol{x}_t$ for all frames where $\bar{q}$ and q give a different phoneme

Updating a density function by LVQ

Mixture means $\boldsymbol{\mu}$

- Find the nearest mixture density (o) used for the likelihood computation
- Tune its mean vector $\boldsymbol{\mu}_{io}$ closer to $\boldsymbol{x}_t$, if the segmentation is correct ($\eta = 1$)
- If incorrect ($\nu = 1$), tune away from $\boldsymbol{x}_t$
- $$\hat{\boldsymbol{\mu}}_{io} = \frac{\sum_{t=1}^{T}[\eta_t(i,o)\boldsymbol{x}_t + \nu_t(i,o)(2\boldsymbol{\mu}_{io} - \boldsymbol{x}_t)]}{\sum_{t=1}^{T}[\eta_t(i,o) + \nu_t(i,o)]}$$
  .
- Note that $o = f(\lambda, \boldsymbol{x}_t)$

Updating a density function by LVQ

Where does the $2\boldsymbol{\mu} - \boldsymbol{x}$ come from?

- Training by batch of T_1 correct vectors
- $\hat{\boldsymbol{\mu}} = \frac{1}{T_1} \sum_{t=1}^{T_1} (\boldsymbol{\mu} + \alpha(\boldsymbol{x}_t - \boldsymbol{\mu}))$
- In batch training, the old value of $\boldsymbol{\mu}$ is often forgotten ($\alpha = 1$)
- So, $\hat{\boldsymbol{\mu}} = \frac{1}{T_1} \sum_{t=1}^{T_1} \boldsymbol{x}_t$
- Similarly, for T_2 incorrect vectors
- $\hat{\boldsymbol{\mu}} = \frac{1}{T_2} \sum_{t=1}^{T_2} (\boldsymbol{\mu} - \alpha(\boldsymbol{x}_t - \boldsymbol{\mu}))$
- So, $\hat{\boldsymbol{\mu}} = \frac{1}{T_2} \sum_{t=1}^{T_2} (2\boldsymbol{\mu} - \boldsymbol{x}_t)$

Updating a density function by LVQ

Mixture weights c

- c_{io} is the weight of the nearest mixture density for state i
- Increase c_{io}, if the segmentation is correct ($\eta = 1$)
- If incorrect ($\upsilon = 1$), increase c_{im}, $\forall m \neq o$
- $$\hat{c}_{im} = \frac{\sum_{t=1}^{T} [\eta_t(i,m) + \upsilon_t(i,m) \frac{c_{im}}{1-c_{io}}]}{\sum_{t=1}^{T} \sum_{m=1}^{M} [\eta_t^l(i,m) + \nu_t^l(i,m)]}$$

  .

Complete update conditions

- State sequence q is the correct path and $\bar{q}$ is the best path
- $\eta_t(i,m) = 1$, if $q_t = i$ and $\boldsymbol{\mu}_{im} = \boldsymbol{\mu}_{io}$
- $\eta_t(i,m) = 0$, otherwise
- $\nu_t(i,m) = 1$, if $\bar{q}_t = i$ and $q_t \neq i$ and $\boldsymbol{\mu}_{im} = \boldsymbol{\mu}_{io}$
- $\nu_t(i,m) = 0$, otherwise
- $\upsilon_t(i,m) = 1$, if $\bar{q}_t = i$ and $q_t \neq i$ and $\boldsymbol{\mu}_{im} \neq \boldsymbol{\mu}_{io}$
- $\upsilon_t(i,m) = 0$,otherwise

Some convergence considerations

- The training affects to only the best-matching mixture for each codebook, but the densities are computed using k-nearest ones
- The second best mixture is not modified in the case of the correct classification like in normal LVQ3
- In asr experiments, the models converge to a good accuracy (on the validation set) consirably faster than by, e.g. Segmental K-means or SKM-LVQ2 hybrids
- As an approximation of the GPD framework the convergence properties of the segmental GPD supports also the segmental LVQ.

Differences to rival methods

To the corrective tuning by LVQ2

- Here, the densities are also updated in correct classifications
- Here, with batch training mode, there are no learning rate problems caused by the gradually improving segmentation
- Here, the mixture weights can be simultaneously adjusted
- Due to the increased robustness of the training, no extra method is required to produce the initial models

Differences to rival methods

To the segmental GPD

- Here, no rival mixtures representing false classifications for already correctly recognized phonemes will be tuned away
- This decreases the amount of tuning operations pointing away from the "observation hyperplane"
- Here, the magnitude of parameter modifications is not directly related to the amount of misclassification of the whole word
- In GPD, the amount of misclassification is usually determined using the differences of the log-likelihood scores over the whole paths

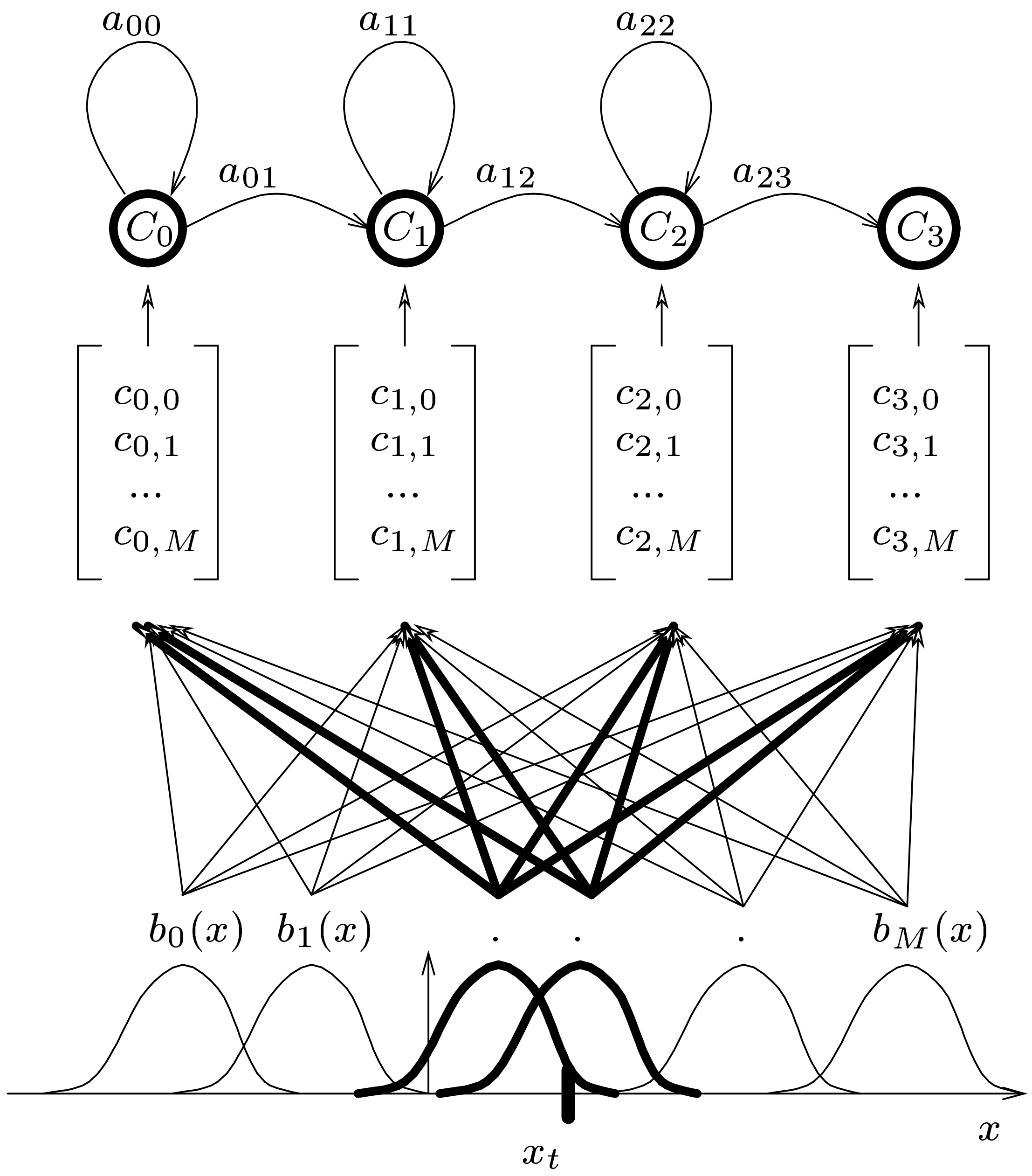
The current method has some contact points also to many other HMM training methods like SKM, ODT, EC, MMI, MLP-HMM, RBF-HMM etc.

Phoneme model

Phoneme-wise tied Mixture Density HMM

- A special form of a semi-continuous HMM
- Each state has not a unique set of mixture densities as in CDHMM
- An extension of discrete HMMs by using a weighted sum of multiple winners per codebook
- The density codebooks are initialized by training small SOMs producing a representative and suitably smoothed set of reference vectors for each phoneme
- The mixture densities for a phoneme is formed using the corresponding SOM units as the centroids of the Gaussian kernel functions
- The likelihood of a state is computed using the k best-matching mixtures of the corresponding phoneme

Phoneme-wise Tied Mixture Density HMM



Recognition task

- Speaker dependent and vocabulary independent recognition of Finnish words, "Phonetic typewriter"

Experiments on "Data95"

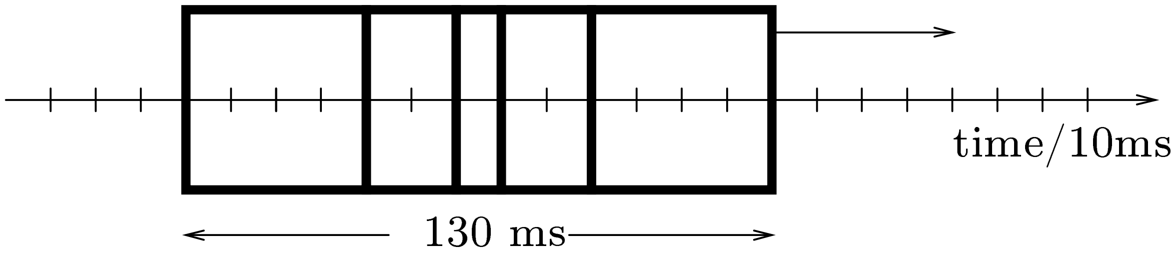
- HMMs for 20 phoneme classes (+ silences)
- 3 speakers and for each speaker 4 sets of 350 words
- 3 sets for training and one for testing, an average error rate of 12 tests used
- Error rate $\equiv \dfrac{\text{changed+missing+extra}}{\text{correct number of phonemes}}$

Feature vectors

Feature vector	Vector dimension			
	cepstra	Δ	other	total
normal	20		rms	21
delta21	10	10	rms	21
delta42	20	20	rms, Δ-rms	42
context	5x20		5xrms	105

- 16 kHz sampling rate
- The "normal feature vector" is a 20-component mel-cepstrum concatenated with energy of the signal
- A new cepstrum for every 8 ms using a 256 point FFT
- The "context vector" is formed by using concatenated averages of several successive short-time features

Context vector



- Concatenated averages of one, two and four successive short-time feature vectors
- Successive context vectors are formed by sliding the window through the sequence of the short-time features.

Testing the Segmental LVQ3

Feature vectors	Mixt /phon	Error rate % on Data 95			
		km	+lvq2	lvq3	+lvq3
normal	70	7.4	7.2	7.4	7.3
normal	140	7.2	7.1	7.1	6.9
context	70	4.9	4.6	4.9	4.8
context	140	-	-	4.5	4.3

- Average phoneme recognition error rates for Viterbi training by segmental K-means, LVQ3 and LVQ2.
- The default amount of training epochs is 10 for SKM and 5 for SLVQ3 and SLVQ2.

Testing the Segmental LVQ3

Feature vectors	Mixt /phon	Error rate % on Data 90			
		km	+lvq2	lvq3	+lvq3
normal	70	5.7	5.5	5.6	5.5
normal	140	5.3	5.4	5.3	5.2
context	70	3.6	3.4	3.5	3.4
context	140	-	-	3.3	3.1

- Average phoneme recognition error rates for Viterbi training by segmental K-means, LVQ3 and LVQ2.
- The default amount of training epochs is 10 for SKM and 5 for SLVQ3 and SLVQ2.

Testing the Segmental LVQ3

A	B	MP risk	MN risk
10x lvq3	lvq3	>0.05	>0.05
km + lvq2	lvq3	>0.05	>0.05
km + lvq2	km	0.005	0.005
km	5x km	0.025	0.005
lvq3	5x km	0.05	0.005
lvq3	km	>0.05	>0.05

- The average error rate by training A is lower than by B, but the difference is statistically significant only if the risk level is not larger than 0.05 in Matched-Pairs (MP) and McNemar's (MN) statistical significance tests.
- The risk levels are the average of the results on Data 95.

About the test results

A	B	MP risk	MN risk
10x lvq3	lvq3	>0.05	>0.05
km + lvq2	lvq3	>0.05	>0.05

- Five training epochs is good enough in the segmental LVQ3
- A slight, but not statistically significant improvement between five and ten epochs
- The results correspond to those by the previously best system (the base line Viterbi training with additional corrective LVQ2 tuning
- Now, the required amount of segmentation-estimation epochs on the training data is only about one third of the previous system.

About the test results

A	B	MP risk	MN risk
km + lvq2	km	0.005	0.005
km	5x km	0.025	0.005

- For the old system the recognition errors decrease from five to ten training epochs of traditional segmental K-means training
- The decreasing continues, when the corrective tuning by LVQ2 is applied.

About the test results

A	B	MP risk	MN risk
lvq3	5x km	0.05	0.005
lvq3	km	>0.05	>0.05

- When started from the same initial vales, the LVQ3 converges considerable faster to the good error rates than the segmental K-means.
- After the first five training epochs the difference is already significant
- If the LVQ training is then stopped, but the K-means still iterated for another five iterations, the results by the segmental K-means have still not yet improved enough to beat those of the segmental LVQ3.

Extending the vector dimension

Feature vectors	Mixt /phon	Rate% on Data 95*	Risk level	Time factor
normal	24	9.8	-	0.5
normal	70	7.7	0.005	1.0
normal	140	7.3	0.01	1.7
delta21	70	6.8	0.025	1.2
delta42	70	6.4	0.025	1.9
delta42	140	5.7	0.005	3.5
context	70	5.3	0.05	3.1
context	140	4.7	0.005	6.0

Extending the vector dimension

Feature vectors	Mixt /phon	Error rate% on Data 90	Risk level	Time factor
normal	24	6.9	-	0.5
normal	70	5.5	0.005	1.0
normal	140	5.2	0.01	1.7
delta21	70	4.7	0.025	1.2
delta42	70	4.0	0.025	1.9
delta42	140	3.6	0.005	3.5
context	70	3.4	0.05	3.1
context	140	3.1	0.005	6.0

Extending the vector dimension

- The HMMs are trained by iterating the segmental LVQ3 for ten epochs over the whole training data.
- The Data 90 error rate is an average of 12 tests by 3 experienced speakers.
- In the Data 95* there are 7 mostly inexperienced speakers, one test per speaker.
- The risk level is for the rejection of the hypothesis: "This result is not significantly different than the result on the previous row".
- Risk level 0.05 corresponds to 95 % confidence, which can be considered as limit of the significancy, so all improvements here are significant steps.
- The recognition time factor is the recognition time per word divided by that of the baseline system (the second row).

Conclusions

- The segmental LVQ3 training links together the traditional Viterbi training by the segmental K-means algorithm and the corrective tuning by LVQ into an efficient training algorithm.
- Provides lower error rates than the conventional Viterbi training and similar rates as the additional LVQ2 based tuning, but with a considerably smaller training effort.