ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

Self Organization in Mixture Densities of HMM based Speech Recognition

Mikko Kurimo

Helsinki University of Technology
Neural Networks Research Centre
P.O.Box 2200, FIN-02015 HUT, Finland

Abstract. In this paper experiments are presented to apply Self-Organizing Map (SOM) and Learning Vector Quantization (LVQ) for training mixture density hidden Markov models (HMMs) in automatic speech recognition. The decoding of spoken words into text is made using speaker dependent, but vocabulary and context independent phoneme HMMs. Each HMM has a set of states and the output density of each state is a unique mixture of the Gaussian densities. The mixture densities are trained by segmental versions of SOM and LVQ3. SOM is applied to initialize and smooth the mixture densities and LVQ3 to simply and robustly decrease recognition errors.

1. Introduction

The automatic speech recognition (ASR) has long traditions as a prototype application for ANNs. Huge efforts have also been invested to study the ASR itself, but it has turned out to be such a difficult problem that the satisfactory solution to the general task is still open. Despite scientific challenges the benefits of even slight improvements are evident, however, because there are already a substantial amount of different ASR applications going on.

The commonly applied approach for the speech to text transformation can be briefly characterized by three successive stages. The first stage is to extract the suitable features from the speech signal. The next stage is to match the features with the statistics of the expected models and construct the model sequences that could most likely correspond the obtained sequence of features. The last stage is then to select the best alternative text decoding by taking also account of the content restrictions of the current ASR task.

In the Neural Networks Research Centre at the Helsinki University of Technology the emphasis in ASR has been on the speaker-dependent recognition of unlimited vocabulary. The "Neural Phonetic Typewriter" [3] and its later developments have several years been a test bench for the SOM and LVQ algorithms [4]. Currently, the system models phonemes by HMMs (hidden Markov models) [9]. where the output densities are based on tied Gaussian mixtures. The parameters of the Gaussian mixtures are learned automatically from the training samples using the SOM [6, 5]. The segmental LVQ3 training [7] opti-

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

mizes the model parameters by learning from the simulated recognition results on the training data. This paper completes the comparison [7] to standard training methods by some new modifications and speech data. The last step in the learning process is to automatically generate DEC rules (Dynamically Expanding Context) [2] to correct the observed differences between the output strings of the recognizer and the correct written form of the phoneme combinations.

2. Training the HMMs

The objective is to use the collected speech samples to train the phoneme models to decode the samples with as few misrecognitions as possible. While using the training data extensively, special care must be taken to avoid overlearning. Another point is the strong interdependence between the segmentation of the data and the optimal model parameters. The commonly used segmental training optimizes simultaneously the segmentations and the parameters in an iterative process that performs segmentation and parameter adaptation steps in turns.

2.1. Segmental SOM

Due to various phenomena in speech the models capable of accurate recognition must usually have hundreds of thousands of parameters. The optimal parameter value estimation would require an excessive amount training data and very efficient training methods by which the essential features of data could be learned in a feasible time. In practice, the limits of data must be accepted and smooth representations learned instead of models full of very fine details.

In SOM the smoothness of representation is obtained by not tuning single units independently towards the special input space areas, but tuning the neighboring units on the SOM grid as well. The structure of the SOM will eventually fit the feature space structures as well as it can and reflect also the density function by the density of the SOM units. Because the neighborhood decreases gradually, the close adaptation to the training data occurs only for the input space areas that are well represented in the samples.

The first stage in the training of the mixture density HMMs by SOM is to initialize the mean vectors. One SOM is trained for each phoneme using presegmented training data. The SOM is then applied to divide the training data into smaller partitions, one for each SOM unit. Each of these data partitions initializes then one Gaussian mixture component. The second stage is to segment or re-segment all available speech data by the obtained initial MDHMMs and adapt the models by a batch SOM iteration. This second stage is iterated using gradually decreasing neighborhood until acceptable models are reached.

The batch SOM iteration for MDHMMs [5] resembles the parameter adaptation in the conventional Viterbi training by the segmental K-means (SKM) algorithm [9]. The difference is to adjust as well the neighboring mixture components (by the SOM topology) of the best-matching one. When the SOM neighborhood radius has decreased to zero the following iterations will equal to SKM iterations. A simple example of how to adjust the best-matching mixture

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

component and how it affects the total mixture density is shown in Figure 1a. For the adaptation of the mixtures by the segmental SOM c.f. [5].

The SKM is the simplest version of the set of conventional maximum likelihood training algorithms for MDHMMs. At the other extreme there is the iterative Baum-Welch (BW) algorithm [9], which directly maximizes the likelihood of the data given all the models, not only the sequence of best matching models and mixture components in them as in the SKM. The BW adapts the parameters of the states according to the posterior probabilities computed for each state in each frame of the sample. The posterior probability computations for all training samples to estimate all the parameters involves, however, laborious computations and often due to modeling assumptions and other reasons the obtained posteriors may be too inaccurate for successful estimation. In Table 2 a segmental version of the BW (BW s1, a.k.a. embedded BW), which adapts only the most probable states, is tested as well. It is rather near to the weighted adaptation of mixture components in SKM (sometimes called embedded BW as well). The fundamental difference between the BW and the segmental SOM is that BW adapts all the mixture components by their approximative probabilities, but in SOM the adaptation is localized around the best match, which maintains the smoothness and ordering of the map.

2.2. Segmental LVQ3

Since neither the Viterbi training nor the SOM is designed to minimize recognition errors, the error rate is optimized by applying discriminative training. Popular methods are, e.g. the corrective training c.f. [8], and the MCE/GPD (minimum classification error/generalized probabilistic descent) c.f. [1]. The segmental LVQ3 [7] resembles the former ones, but consists of simpler operations and local adaptations and is in that way a more suitable to be integrated to "neural" methods.

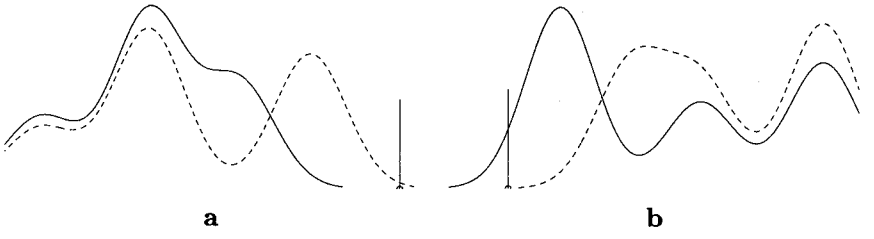

Figure 1: **a** Adjustment of a mixture density towards one observation (the short vertical line) to increase its likelihood for a HMM state (the basic K-means type adaptation). The modified parameters are the centroid of the nearest Gaussian and the mixture weights (the sum of weights is always one). The resulting new pdf for this simple one-dimensional three-mixture case is shown dashed. **b** A mixture density of an incorrect state is adjusted away from the observation to decrease the likelihood (the basic adaptation in discriminative training). The nearest Gaussian and the mixture weights are modified accordingly.

The segmental LVQ3 operates in two different modes depending on the outcome of the recognizer for the training sample (word). For incorrect re-

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

sult the discriminative learning is applied, but otherwise only the simple SKM adaptation. In principle, the adjustments are equal to performing a kind of batch version of the LVQ3 [4] to the parameters of the best state sequence that produces the correct result and of the corresponding state sequence for the incorrect result. A simple example of how to adjust mixture densities in one dimension is shown in Figure 1. For a correct state the adjustments of the output density occur as in the Figure 1**a** and for an incorrect state as in the Figure 1**b**. For the exact description of adaptation laws c.f. [7].

The segmental LVQ3 differs from the original corrective training and the MCE/GPD, mainly because it does not try to relate the extent of each individual parameter modifications to any exact measure of the misclassification of the whole training token. The non-discriminative mode is added instead to gain more robustness over the initialization and stability between the positive and negative adaptations. One motivation is to join the stability of the SKM to the corrective training. The segmental LVQ3 is as well very close to the extreme case of the MCE/GPD where the sigmoidal loss is approximated by a piece-wise linear function and the L_p norm by the L_∞ norm, and thus it approaches the similar convergence as the MCE/GPD.

2.3. SOM topology in HMMs

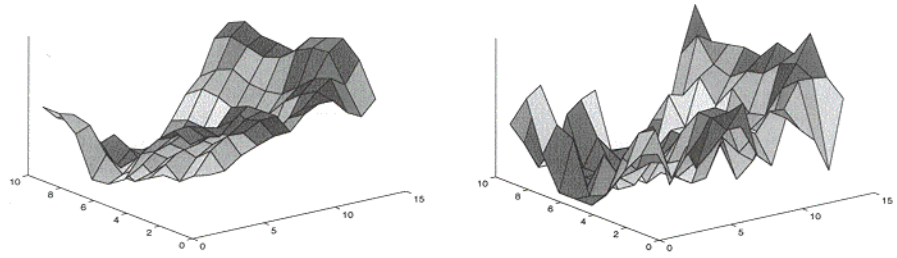

Figure 2: The structure of a high-dimensional mixture density is visualized by plotting the responses of the individual mixture components for one input. The 140 mixture components are organized into a 10x14 grid. In the rather smooth view **a** the SOM radius is decreased to one. When the neighborhood radius decreases to zero (**b**) only coarse ordered structures can be seen.

In a typical HMM based speech recognition process, the major computational effort is required for the mixture density values of each state for each new feature vector. Thus, in addition to the accuracy, the mixture densities must be simple to compute and flexible to approximate for the low-probability states. Since the pool of Gaussians for a HMM is rather large and usually only a small subset of Gaussians will have a significant value at a time, the mixture density can be well approximated by using only the significant mixture components. A couple of fast search methods that exploit the SOM topology of the mixtures are given in [6]. For example, the total average recognition time of one word can be decreased by 42 % [5] by using the so-called topological K-best search. The observed average increase of the error rate was from 5.9 % to 6.5 %. This

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

is due to the increased inaccuracy of the search when the ordering declines as in Figure 2. For the segmental LVQ3 the loss was not much bigger (from 5.3 % to 5.9 %) for the same speed improvement which suggests that some useful topological structure preserves even through the further training.

3. Experiments

Init.	HMM training	70 mixt.	140 mixt.
SOM	SSOM	5.6	4.9
SOM	SLVQ3	5.4	4.7
SOM	SSOM+SLVQ3	5.6	4.9
KM	SKM	6.3	5.5
KM	SGPD	5.8	5.5
KM	SKM+SGPD	5.6	4.9

Table 1: The average test set error rates for some alternative training methods.

In the experiments the speaker-dependent HMMs were trained for 10 Finnish speakers (6 males and 4 females) using three of the four recorded 350-word sets per speaker. The fourth set is left out for testing only. The 80-dimensional feature vector is a concatenation of five sets of 15 cepstrum coefficients and the power of the signal computed from 16 ms signal windows and averaged over different time spans and positions relative to the current window [7]. The HMM of each phoneme includes a chain of five states and a pool of maximum 140 Gaussians with a fixed diagonal covariance matrix. The error rate is the sum of inserted, deleted and changed phonemes divided by the correct number of phonemes and averaged over all the speakers. In the Table 1 the basic SOM is used for initialization [6] followed by 10 epochs of segmental training (or 5+5 by method combinations) using the whole training data as a batch. The reference method includes K-means initialization and SKM and/or segmental MCE/GPD training [1] thereafter.

In the Table 2 some modifications of the SKM and BW reference methods are tested: Adjusting of only the best state (BW s1, default for SKM) and adjusting the 3 closest mixtures per state without weights (k3) and with weights (w3). The adaptation weight in weighted SKM is the ratio of the contribution of a single mixture component to the whole density of the state. In BW the weights are the probability approximations.

4. Conclusion

This paper presents experiments and motivations for SOM and LVQ3 based MDHMM training methods and some reference methods. The experiments are made by testing the methods in a prototype speech recognition system, which transforms speech to text in a vocabulary independent way. The new tests complete the comparisons [7] by using more versions of the standard training methods and new speech data. The improvement from the best reference method in terms of the average reduction of phoneme errors is not very large,

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 227-236

HMM training	Rate%	HMM training	Rate%
SKM	5.5	BW	7.9
		BW s1	7.4
SKM k3	7.6	BW s1,k3	11.5
SKM w3	5.4	BW s1,w3	7.4
SKM w20	5.4	BW s1,k1	7.6

Table 2: The average test set error rates for some modifications of the reference training method using the same K-means initialization for 140 mixtures per phoneme. The results are averages for five speakers.

but still about 5 %. The test results show, however, that the segmental K-means or GPD/MCE alone cannot give the best results even after rather long training. Issues omitted here, such as comparisons of alternative feature vectors, initialization methods, convergence speeds to good error rates and the statistical significance of the differences have been presented in [6, 7]. Since the principles of the suggested segmental training methods suit to a wide range of other classification tasks involving sequential state models with stochastic observations, the further work is to proceed testing in different databases and applications.

References

[1] W. Chou, B.H. Juang, and C.H. Lee. Segmental GPD training of HMM based speech recognizer. In *Proc. of the ICASSP'92*, pages 473–476, San Francisco, USA, 1992.

[2] Teuvo Kohonen. Dynamically expanding context, with application to the correction of symbol strings in recognition of continuous speech. In *Proc. of the ICPR'86*, pages 1148–1151, Paris, France, 1986.

[3] Teuvo Kohonen. The 'neural' phonetic typewriter. *Computer*, 21(3):11–22, 1988.

[4] Teuvo Kohonen. *Self-Organizing Maps.* Springer, Berlin, 1995.

[5] Mikko Kurimo. SOM based density function approximation for mixture density HMMs. In *Workshop on Self-Organizing Maps*, pages 8–13, Espoo, Finland, 1997.

[6] Mikko Kurimo. *Using Self-Organizing Maps and Learning Vector Quantization for Mixture Density Hidden Markov Models.* PhD thesis, Helsinki University of Technology, Espoo, Finland, 1997.

[7] Mikko Kurimo. Training mixture density HMMs with SOM and LVQ. *Computer Speech and Language*, 1998. (accepted for publication).

[8] Shinobu Mizuta and Kunio Nakajima. An optimal discriminative training method for continuous mixture density HMMs. In *Proc. of the ICSLP'90*, pages 245–248, Kobe, Japan, 1990.

[9] Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.