BMBF

Verbmobil
Verbundvorhaben

A New Model-Discriminant Training Algorithm for Hybrid NN-HMM Systems

W. Reichl
P. Caspary
G. Ruske

Technische Universität München

Vm
Report 108
April 1996

April 1996

W. Reichl
P. Caspary
G. Ruske

Forschungsgruppe Sprachverarbeitung
Lehrstuhl für Mensch-Maschine-Kommunikation
Technische Universität München
Arcisstraße 21
80290 München

Tel.: (089) 2105 - 8554
e-mail: reichl@e-technik.tu-muenchen.de

Gehört zum Antragsabschnitt: TP3 Spracherkennung und Sprecheradaption

Das diesem Bericht zugrundeliegende Forschungsvorhaben wurde mit Mitteln des Bundesministers für Bildung, Wissenschaft, Forschung und Technologie unter dem Förderkennzeichen 01 IV 102 C/6 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei den Autoren.

A NEW MODEL-DISCRIMINANT TRAINING ALGORITHM FOR HYBRID NN-HMM SYSTEMS

W. Reichl *P. Caspary* *G. Ruske*

Lehrstuhl für Datenverarbeitung, Technische Universität München
Franz-Joseph-Str. 38, D-80801 München, Germany

ABSTRACT

This paper describes a hybrid system for continuous speech recognition consisting of a neural network (NN) and a hidden Markov model (HMM). The system is based on a multilayer perceptron, which approximates the a-posteriori probability of a sequence of states, derived from semi-continuous hidden Markov models. The classification is based on a total score for each hybrid model, attained from a Viterbi search on the state probabilities. Due to the unintended discrimination between the states in each model, a new training algorithm for the hybrid neural networks is presented. The utilized error function approximates the misclassification rate of the hybrid system. The discriminance between the correct and the incorrect models is optimized during the training by the 'Generalized Probabilistic Descent Algorithm', resulting in a minimum classification error. No explicit target values for the neural net output nodes are used, as in the usual backpropagation algorithm with a quadratic error function. In basic experiments up to 56 % recognition rate were achieved on a vowel classification task and up to 69 % on a consonant cluster classification task.

1. INTRODUCTION

Most current speech recognition approaches use a hidden Markov model based system (HMM), consisting of a parametric production model for each particular speech segment such as phonemes, syllables or words. These models depend on some assumptions on the statistical independence of the features and special pattern distributions, e.g. Gaussian pdfs. Normally the individual HMMs are trained separately by maximum likelihood estimation, or alternatively discriminant training algorithms as e.g. maximum mutual information optimization are used. Discriminant training is useful, in cases where the training set is small.

Connectionist architectures have been successfully applied for different classification tasks. Their learning algorithms are inherently discriminative. Usually layered feedforward nets, called multilayer perceptrons (MLP), are employed. Since these neural nets are not able to handle the dynamic distortion occurring in the speech process, a combination with a DP algorithm for optimal time warping is required. Using the backpropagation algorithm with a quadratic error function results in an approximation of the a-posteriori probability for the assigned class, conditioned by the input of the net [1].

In most hybrid NN-HMM systems the target values for the output nodes are derived from HMMs or from hand labelled phonetic speech segments. Normally they consist of fixed values for the 'active' state. This is the state an input vector is assigned to by the Viterbi alignment. In this way the different states in each model are trained discriminatively. Finally the neural net approximates a-posteriori probabilities, which can be used as emission probabilities for the underlying HMMs [1]. The output values of the neural net are processed by the Viterbi algorithm to calculate a score for each model within the boundaries of the phonetic segment. This can be done by multiplying the probabilities along the best sequence of states [1], [2]. In addition transition probabilities between the states can be used.

Another approach for the calculation of the model scores is to add an additional connectionist word unit, summing up the values of the output nodes along the Viterbi path. This method is used in [3] for a word level training with fixed word targets ('0.0' or '1.0'). Using the 1-stage DP algorithm for the whole speech utterance, an optimal segmentation and classification in a continuous speech recognition task is achieved.

In this paper a new training method for hybrid NN-HMM systems is proposed. An error function, which approximates the misclassification rate of the hybrid system, is optimized by the 'Generalized Probabilistic Descent Algorithm'. This is achieved iteratively by maximizing the discriminance between the correct and the wrong models and results in a minimum error classifier [5]. Some results for two continuous speech recognition experiments, a syllable based vowel and consonant cluster task, using this model-discriminant training algorithm are presented in the following.

2. THE HYBRID NN-HMM SYSTEM

For a continuous speech classification task with N different speech segments and fixed segment boundaries a hybrid NN-HMM system, consisting of a multilayer perceptron and a Viterbi algorithm, was employed. One hybrid model W_n is made up of 3 to 5 states q_{nm}, as in the underlying HMM, with $m = 1, \ldots, 3, (4), (5)$ and $n = 1, \ldots, N$. Each state q_{nm} of each model is assigned to one output node O_{nm} of the neural net. The activation of the neurons is computed according to the usual summing rule: $O_{nm} = f\left(\sum_i W_{nmi} o_i\right)$ with the sigmoid transfer function $f(a) = (1 + e^{-a})^{-1}$. The weight W_{nmi} connects the output node O_{nm} with the node o_i in the hidden layer. The nodes in the hidden layers work in the same way as the output nodes do. The used neural net is a multilayer perceptron with 1 or 2 hidden layers and different numbers of neurons in the hidden layer(s). The input layer consists of 1 or 5 consecutive frames of the feature vector, which can be a combination of the 20-dimensional Bark-scaled loudness spectrum, the total loudness and the delta loudness spectrum. The features are computed every 10 ms from the acoustical preprocessing.

Presented at the *Proc. ICASSP-94*, Adelaide, Australia, April 1994, pp. 677-680.

For the initial training of the MLP the usual backpropagation algorithm with a quadratic error function was used. The targets for the output nodes were derived from a semi-continuous HMM (SCHMM) system for demi-syllable based units [4]. The best path of the Viterbi algorithm in the (given) correct model marks the sequence of 'active' states. Along this sequence the targets for the corresponding output nodes of the MLP were set to '0.95'. The targets for the inactive states of the correct model were '0.5' and for all incorrect models '0.05' [2]. Figure 1 shows the activations of the output units and the sequence of states for the correct and an incorrect model. Black boxes indicate activations up to '1.0' and white ones near '0.0'.

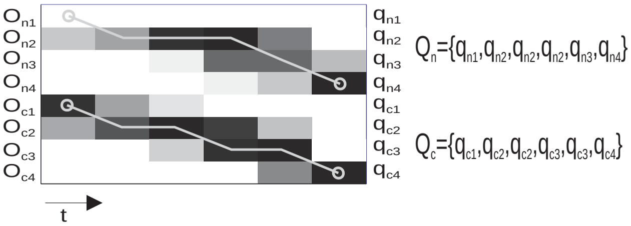

Figure 1. Activations of the output units and the sequence of states for the correct and an incorrect model

The outputs of the trained MLP approximate the a-posteriori probability of the individual states of the hybrid system conditioned by the actual feature vector x_t : $O_{nm} \approx P(q_{nm}|x_t)$ [1]. The output values of the neural net are used by the Viterbi algorithm under consideration of the best sequence of states $Q_n = \{Q_{n1}, \ldots, Q_{nT} | Q_{nt} \in \{q_{nm}\}\}$ to calculate the total score $g_n(X)$ of the model W_n for a feature vector sequence $X = \{x_1, \ldots, x_T\}$.

$$
\begin{aligned}
g_n(X) &= \prod_{t=1}^{T} O_{nQ_{nt}}(t) = \prod_{t=1}^{T} P(Q_{nt}|x_t) && (1) \\
&= P(Q_{n1}, \ldots, Q_{nT}|X) && (2)
\end{aligned}
$$

The total score is an approximation of the a-posteriori probability $P(W_n|X)$ of the model W_n under consideration of only the best path. The classifier decides for the class c with the maximum score $g_c(X) = \max_n\{g_n(X)\}$. The classification process is the same as for HMMs and can be connected with the well known 1-stage DP algorithm or beam search methods for continuous speech recognition purposes.

3. A NEW MODEL-DISCRIMINANT TRAINING ALGORITHM FOR HYBRID NN-HMM SYSTEMS

The explicit use of target values for the output nodes of the net entails an unintended discriminance within the models, because the neural net has to produce a '1'-output for the 'active' node and '0'-outputs for the others. This is an unnecessary condition and is difficult to achieve, especially if similar feature vectors are assigned to different states in the same model, e.g. if the same phonemes occur in the beginning and the end of demi-syllable or word models. The neural net is trained to approximate the target values on the frame basis but the classifier is working on the model scores. This discrepancy between the objective function during training and the decision function of the classifier leads to good frame based recognition performance, but causes problems in model-based recognition [1]. Furthermore, the minimization of MSE doesn't generally result in a minimum error probability classifier [5]. Therefore we developed an objective function which directly minimizes the decision error rate of the classifier without using explicit targets for the MLP. Between fixed segment boundaries $(t = 1, \ldots, T)$ we define, similar to [6] and [5], the misclassification measure for the correct class c

$$
d_c(X) = -r_c(X) + \log \left\{ \frac{1}{N-1} \sum_{n;n\neq c} e^{r_n(X)\eta} \right\}^{\frac{1}{\eta}} \qquad (3)
$$

where $r_n(X) = \log(g_n(X)) = \sum_{t=1}^{T} \log(O_{nQ_{nt}}(t))$ is the log score of the model n. A misclassification measure $d_c(X) > 0$ implies misclassification and $d_c(X) < 0$ indicates a correct decision. The misclassification measure makes use of the L_η-norm of the model scores g_n. The positive number η controls the number of incorrect models included in the training process. In the case of $\eta \to \infty$, only the wrong model with the highest score is considered. The misclassification measure is continuous with respect to the classifier's parameters and therefore suitable for a gradient-type optimization algorithm. The following cost function approximates the error rate of the classifier

$$
l_c(d_c(X)) = \frac{1}{1 + e^{-\gamma d_c(X)}} \qquad (4)
$$

with $\gamma > 0$. This is a smoothed zero-one cost function, which 'counts' the classification errors. If $d_c(X) > 0$ then $l_c \to 1$ and no cost occurs $(l_c \to 0)$ if $d_c(X) < 0$. The optimization of this objective function with respect to the parameters λ of the MLP results in an optimal minimum error classifier [5]. This optimization can be done iteratively by the 'Probabilistic Descent Algorithm' [7]. The parameters in the k-th iteration are updated according to the following rule

$$
\lambda_{k+1} = \lambda_k - \epsilon_k U_k \nabla_{\lambda_k} l_c(\lambda_k, X) \qquad (5)
$$

where U_k is a positive-definite matrix (here the identity matrix is used for U_k) and ϵ_k is a small positive number, controlling the step size. Each time a training segment X is presented, the parameters (weights) of the MLP are adjusted proportionally to the negative gradient of the objective function. The gradient of the objective function with respect to the weights of the MLP can be computed with the chain rule. The convergence of the 'Probabilistic Descent Algorithm' to the optimal Bayes classifier is ensured with proper initialization and appropriate selection of the learning step size ϵ_k [7], [5]. The partial derivative of the cost function with respect to the misclassification measure is

$$
\frac{\partial l_c(d_c(X))}{\partial d_c(X)} = \gamma l_c(1 - l_c) \qquad (6)
$$

This derivative has its maximum in $d_c(X) = 0$. In the vicinity of this point the scores of the models are similar and misclassification is likely to occur. If the model scores are different the derivative of the sigmoid-function declines rapidly, causing no further training. To compute the derivative of the misclassification measure with respect to the model scores the correct and the wrong models must be distinguished.

$$
\begin{aligned}
\frac{\partial d_c(X)}{\partial g_c(X)} &= -\frac{1}{g_c(X)} && (7) \\
\frac{\partial d_c(X)}{\partial g_n(X)} &= \frac{1}{g_n(X)} \frac{g_n(X)^\eta}{\sum_{n\neq c} g_n(X)^\eta} && (8)
\end{aligned}
$$

The score of the model $g_n(X)$ consists of the product of the output values along the Viterbi path (1). Therefore the derivative of the score with respect to one specific output node O_{nm} is

$$\frac{\partial g_n}{\partial O_{nm}} = \sum_{t:Q_{nt}=q_{nm}} \frac{\partial g_n}{\partial O_{nm}(t)} = \sum_{t:Q_{nt}=q_{nm}} \frac{g_n}{O_{nm}(t)} \tag{9}$$

The update rule (5) for the weight W_{nmi} of model n in the output layer of the neural net requires the following derivative

$$\begin{aligned}\frac{\partial l}{\partial W_{nmi}} &= \frac{\partial l}{\partial d}\frac{\partial d}{\partial g_n} \sum_{t:Q_{nt}=q_{nm}} \frac{\partial g_n}{\partial O_{nm}(t)}\frac{O_{nm}(t)}{\partial W_{nmi}} \\ &= \sum_{t:Q_{nt}=q_{nm}} -\Delta_{nm}(t)o_i(t) \end{aligned} \tag{10}$$

Using 'delta'-terms Δ_{nm}, similar to the usual 'deltas' in the backpropagation algorithm, the structure of the equation remains the same. The 'delta'-terms for the model-discriminant training procedure are different for the correct and incorrect models :

$$\Delta_{nm}(t) = \begin{cases} -\gamma l(1-l)\frac{g_n^\eta}{\sum_{n\neq c} g_n^\eta}(1-O_{nm}(t)) & \forall n \neq c \\ \gamma l(1-l)(1-O_{cm}(t)) & n = c \end{cases} \tag{11}$$

In case of an incorrect class the 'delta'-terms are weighted by their contribution to the L_η-norm in the misclassification measure. Only incorrect models with high scores, which are likely to be confused, are considered in the learning process. The differentials of the error measure with respect to the weights of the hidden layers are computed continuously using the chain rule as in the backpropagation algorithm.

The model-discriminant training procedure leads to minimum error classification without the utilization of targets. It increases the score of the correct model and decreases that of the others in the sense of a corrective training without causing discriminance within the models. Hence only 'active' states along the optimal Viterbi alignment are considered in the training procedure as well as in the classification process. The remaining states and their attached output nodes of the neural net are not trained by the learning process. The supervision and objective function works on model level. It is possible to connect the models with the Viterbi algorithm for training on word or sentence level. Like in HMMs, the model boundaries are then freely aligned in training as well as in the recognition process.

4. EXPERIMENTS

In the performed continuous speech recognition experiments a speaker independent database of 10 German speakers (Phondat: 'Berliner'-sentences) was used for training (dataset I) and cross-validation (dataset II). Each speaker uttered 2 versions of 100 German sentences. We used the utterances of 6 speakers for training and the remaining 4 speakers for cross-validation. The generalization performance of the hybrid NN-HMM system was tested with a third independent database (dataset III). This consists of another 2 versions of different 100 German sentences, spoken by 6 of the 10 speakers (training and test speakers) from the first database (Phondat: 'Marburger'-sentences). After a 256-point FFT with Hamming window the power spectrum was combined in critical bands. Every 10 ms a Bark-scaled loudness spectrum was computed, which was normalized to sum up to one. Additional features, as the total loudness, the delta-loudness spectrum and the zero-crossing rate of the signal, were also used in the experiments.

In the first phase of the training the MLPs learn to reproduce the sequence of states provided by a semi-continuous HMM system within fixed segment boundaries [4]. A modified MSE criterion was used in the standard backpropagation Algorithm. In the next step the model-discriminant training algorithm is used to adjust the weights of the MLPs for minimum error classification. This training step is very fast, only a few iterations are necessary, as compared to the standard backpropagation training, which needs some hundred iterations.

The syllable structure of speech has been successfully utilized for the recognition of continuous German speech [8]. The HMM system is based on parts of syllables and uses phonotactic constraints for the German language. The German language can be described with 50 initial consonant clusters (ICC), about 20 vowels and diphthongs (VOW) and 160 final consonant clusters (FCC), which are composed of 24 rudiments and 17 suffixes.

In the first experiment 17 vowels and diphthongs, occurring in the databases, were trained. Long and short variants of vowels were treated separately. The MLP consisted of 77 output nodes (states), different numbers of hidden layers (one and two hidden layers) and 50 or 100 nodes in the hidden layers. The input layer of the MLP net is made up of a sliding window of several frames. Each frame consists of different features derived from the acoustic processing. In Table 1 the results of some experiments with the hybrid MLP-HMM system are summarized. The number of neurons in the hidden layers are given in the first column. In nets with one hidden layer we used 50 and 100 neurons and in nets with two hidden layers we used 100 neurons in the first and 50 neurons in the second hidden layer. The next two columns of Table 1 show the rates of correctly recognized vowels (VOW). In column 2 the results of the hybrid system trained with a feature vector consisting of the 20 dimensional loudness spectrum, the total loudness and the zero crossing rate are printed. Here the MLP input layer is made up of a sliding window of 5 consecutive frames of the 22 dimensional feature vector and thus has a total of 110 nodes. The results of the nets which additionally utilize the delta-loudness spectrum are depicted in column 3. As the delta-loudness spectrum incorporates information about the temporal process of the spectrum, no temporal window in the input layer (1x42 neurons) was used. The recognition rates for the training data - dataset I - are given in the first lines for all the nets in Table 1. The results for the cross reference test - dataset II - and the results for the second test set - dataset III - are also illustrated.

For all 3 datasets the hybrid MLP-HMM system shows good recognition rates in the vowel and diphthong (VOW) experiments. These are about the same values as the results of the semi-continuous HMMs, which work very well on this task. The number of weights in the MLPs is between 6,000 and 14,000, which is considerably less as compared to the number of parameters in the SCHMMs. The results of the more complex nets are not better than those of the simpler nets with 50 neurons in the hidden layer. More layers and neurons are apparently not advantageous for this task. The MLP with 50 neurons in a single hidden layer and an input layer of 42 neurons shows the best results. About 58 % of the vowels and diphthongs in the training and cross-validation database and 55.5 % in the test set are correctly recognized by the hybrid system after the model-

No. Neurons			Dataset
Hidden Layer	Input Layer		
	5x22	1x42	
50	58.4	58.2	I
	56.4	57.9	II
	52.5	55.5	III
100	55.8	57.2	I
	55.0	54.5	II
	50.3	54.3	III
50,100	58.4		I
	53.5		II
	52.2		III

Table 1. Vowel (VOW) recognition rates for different hybrid MLP-HMM systems for the 3 datasets; all values in %

discriminant training. These results are about 6 % higher for the training, 3 % higher for the cross-validation and 3 % higher for the test data, as compared to the initial training results.

In the second experiment 29 of the most frequent initial consonant clusters (ICC) are modelled with different hybrid systems. The MLPs for these hybrid systems consist of 120 output nodes and 50 or 100 nodes in the hidden layers. Table 2 shows the results of the consonant cluster experiments for the 3 datasets.

No. Neurons			Dataset
Hidden Layer	Input Layer		
	5x22	1x42	
50	80.0	79.0	I
	62.7	63.6	II
	64.8	65.1	III
100	83.9	84.0	I
	65.8	64.5	II
	68.9	69.3	III
50,100	79.5		I
	61.0		II
	61.8		III

Table 2. Initial consonant cluster (ICC) recognition rates for different hybrid MLP-HMM systems for the 3 datasets; all values in %

The results for the hybrid MLP-HMM system on the initial consonant cluster (ICC) task show some improvement compared to the SCHMMs. The SCHMM recognition rates for the training database are about 80 % and about 60 % for the cross-validation and test datasets. The hybrid systems reach about 84 % for the training and about 66 % to 69 % for the cross-validation and test datasets after the model-discriminant training. The improvements, resulting in this training algorithm, are up to 9 % for the training and about 6 % for the cross-validation and test data. The utilization of more neurons in the hidden layers is advantageous for this task. More neurons enable the neural net to form finer representations of the features for the different initial consonant clusters. Using an additional hidden layer is not of advantage for this task.

5. CONCLUSIONS

In this paper a new model-discriminant training algorithm for hybrid NN-HMM systems was presented. On a frame basis this learning method uses no targets for the neural net. A model level supervision and objective function is used for minimum classification error training. This model-discriminant training algorithm was used in some basic experiments with syllable based units. The hybrid NN-HMM system showed very good performance on vowel and consonant cluster tasks in continuous speech recognition. In the future we plan to complete the syllable based inventory with nets for the final consonant clusters to build a complete hybrid speech recognition system. Furthermore we want to include our hybrid models in a beam search framework to recognize sentences by obtaining the best word chain or word lattice for the utterance.

This work has been partly carried out within the ASL/VERBMOBIL project, which is supported by the German BMFT.

REFERENCES

[1] H. Bourlard, C. J. Wellekens: *Links Between Markov Models and Multilayer Perceptrons*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 12, No 12, pp. 1167-1178, December 1990.

[2] M. Franzini, K.-F. Lee, A. Waibel: *Connectionist Viterbi Training: A New Method For Continuous Speech Recognition*, IEEE Proc. 1990 Int. Conf. Acoust. Speech Signal Process., Albuquerque, pp. 425-428, April 1990.

[3] P. Haffner, M. Franzini, A. Waibel: *Integrating Time Alignment and Neural Networks for High Performance Continuous Speech Recognition*, IEEE Proc. 1991 Internat. Conf. Acoust. Speech Signal Process., Toronto, pp. 105-108, May 1991.

[4] B. Plannerer, G. Ruske: *Recognition of Demisyllable Based Units Using Semicontinuous Hidden Markov Models*, IEEE Proc. 1992 Int. Conf. Acoust. Speech Signal Process., San Francisco, pp. 581-584, March 1992.

[5] B. H. Juang, S. Katagiri: *Discriminative Learning for Minimum Error Classification*, IEEE Trans. on Signal Processing, Vol. 40, No. 12, pp. 3043, December 1992.

[6] W. Chou, B. H. Juang, C. H. Lee: *Segmental GPD Training Of HMM Based Speech Recognizer*, IEEE Proc. 1992 Int. Conf. Acoust. Speech Signal Process., San Francisco, pp. 473-476, March 1992.

[7] S. Amari: *A theory of adaptive pattern classifiers*, IEEE Trans. on Elec. Comput., Vol. EC-16, pp. 299-307, June 1967.

[8] B. Plannerer, G. Ruske: *A Continuous Speech Recognition System Using Phonotactic Constraints*, EUROSPEECH 1993, Berlin, pp. 869-862, September 1993.