Global Optimization of a Neural Network–Hidden Markov Model Hybrid

Yoshua Bengio, Renato De Mori, *Senior Member, IEEE,* Giovanni Flammia, *Student Member, IEEE,* and Ralf Kompe

Abstract—**The subject of this paper is the integration of multilayered and recurrent artificial neural networks (ANN's) with hidden Markov models (HMM's). ANN's are suitable for approximating functions that compute new acoustic parameters, whereas HMM's have been proven successful at modeling the temporal structure of the speech signal. In the approach described here, the ANN outputs constitute the sequence of observation vectors for the HMM. An algorithm is proposed for *global* optimization of all the parameters. Results on speaker-independent recognition experiments using this integrated ANN–HMM system on the TIMIT continuous speech data base are reported.**

I. Introduction

In spite of the fact that speech exhibits features that cannot be represented by a first-order Markov model, hidden Markov models (HMM's) of speech units (e.g., phonemes) have been used with a good degree of success in automatic speech recognition (ASR) [1], [2]. Artificial neural networks (ANN's), in particular multilayer networks or recurrent networks trained with back-propagation [3], have proven to be useful for classifying speech properties and phonemes based on the analysis of a speech segment of limited duration [4]–[6]. Various attempts have been made to interpret the time evolution of ANN outputs. Worth mentioning is the postprocessor proposed by Robinson and Fallside [7], which uses dynamic programming with duration and bigram constraints. Along a similar line, researchers have attempted to combine the classification power of ANN's with the time-domain modeling capability of HMM's [8]–[13] or to formalize HMM's in the framework of ANN theory [14]–[16].

This paper is inspired by previous proposals for combining HMM's and ANN's [9], [12], [13], [15] and considers a novel architecture in which ANN's trained with the generalized delta rule [3] perform approximations of functions for computing acoustic parameters to be used as observations by continuous density HMM's (CDHMM's). It is shown how to perform a joint *global optimization* of both the ANN and the HMM parameter estimation by computing the gradient of the optimization criterion for the HMM with respect to the transformed observations. This gradient is sent to the ANN for the estimation of the weight associated with each connection of the network, as depicted in Fig. 1. No assumption need to be made or constraints imposed on the network outputs, except that the network output distribution can be modeled by a mixture of multivariate Gaussians. Another novelty introduced in this paper is that multiple ANN's are combined so that specialized networks for certain phoneme groups are fed by acoustic data pertinent to the characterization of phonemes in that group.

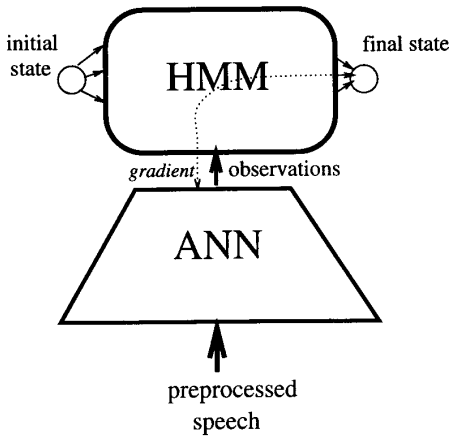


Fig. 1. Proposed ANN/HMM hybrid model: the outputs of the ANN constitute the observation sequence for the HMM. The parameters of both the ANN and the HMM can be estimated in order to perform a global optimization of a given criterion.

Section II relates the contents of this paper to the existing literature. Section III describes the gradient computation of the hybrid system consisting of ANN's providing signal transformations considered as observations of the HMM's. Section IV introduces the algorithms for ANN parameter estimation. Section V contains the details of the system architecture and reports on experimental results obtained for the recognition of the plosive sounds of the TIMIT data base [17]. An accuracy of 86% is obtained when the hybrid system is globally optimized, as opposed to an accuracy of 81% when ANN's and HMM's are trained separately.

Manuscript received March 18, 1991; revised September 6, 1991. This work is part of a project of the Institute for Robotics and Intelligent Systems, a Canadian Network of Centers of Excellence.

Y. Bengio was with the School of Computer Science, McGill University, Montreal, Quebec, Canada. He is now with the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA 02139.

R. De Mori is with the School of Computer Science, McGill University, 3480 University Street, Montreal, Quebec, Canada H3A 2A7.

G. Flammia was with the School of Computer Science, McGill University, Montreal, Canada. He is now with the Speech Technology Center, University of Aalborg, Denmark.

R. Kompe was with the School of Computer Science, McGill University, Montreal, Canada. He is now with the Institute for Pattern Recognition, University of Erlangen-Nürnberg, Germany.

IEEE Log Number 9105177.

II. Related Work

Interesting papers have been published recently describing attempts at combining ANN's with HMM's. In some of the proposed approaches (e.g., [10] and [15]) the activation value of each output node of the network corresponds to $P(observation(t)|state(i))$, the probability of observing a set of acoustic parameter values at time t conditional on the state i of the HMM (this probability will be indicated later as $b_{i,t}$). The ANN is trained to compute these observation probabilities for the best sequence of states produced by the alignment with the speech signal. In [10] the input data are aligned with the model of the spoken utterance with the Viterbi algorithm. In this case, the observation probabilities are approximated by the network outputs. Another approach is found in Bridle's alphanets [15] and consists in viewing the forward pass of the Baum–Welch algorithm [18] as a particular type of recurrent network with linear sums, products, and single delays. With this point of view, ANN's and HMM's can be seen as a single network for which the gradient of an optimization criterion with respect to all system parameters can be computed (although the HMM parameters require some normalization). The idea of considering a unified paradigm for ANN's and HMM's is also considered in [12]–[14]. An often cited advantage of such a combination is to make the HMM more discriminant [9], [13], [15]. This objective can be attained with the approach proposed in this paper when the hybrid is trained with the maximum mutual information estimation (MMIE) criterion. The ANN outputs are considered as observations for the HMM, and the HMM's are trained with methods that have already been proven very efficient for them.

Other hybrid systems combining ANN's with HMM's (e.g., [9] and [11]) have severe theoretical requirements (the ANN must have enough parameters and training has to converge to the global minimum) in order to express the posterior probability $P(state(i)|observation(t))$. Our previous work on hybrid models [8] used ANN's merely to compute an additional set of symbols considered as observations for a discrete HMM. A vector-quantized codebook was generated from these parameters and added to codebooks obtained for other acoustic parameter sets. This did not require any assumption on the network outputs but had the disadvantage that the ANN and the HMM were trained separately. The method described in the present paper makes it possible to perform global parameter optimization by transmitting to the ANN a gradient computed for the HMM. Furthermore, in order to achieve rapid convergence to a solution, in the experiments described here, global optimization is practically performed as a global tuning that starts from an initial point determined by prior (separate) training of the ANN and HMM. The ANN is first trained to approximate recognition of phonetic features, such as place of articulation. The HMM is trained with the Baum–Welch algorithm using the trained ANN outputs as observations. Only as a final step is a global tuning performed in order to optimize parameter estimation of the whole system.

III. Gradient Computation in the Hybrid ANN/HMM System

In this paper, only left-to-right HMM's with a single final state are considered. Let Y_t be the vector of ANN outputs at time t. These outputs are considered as observations of a CDHMM used in the scheme shown in Fig. 1. Let Y_1^T be the whole observation sequence for the HMM, T is the length of the observation sequence, and Y_t a particular observation, made when the HMM is in the state S_t at time t. Let a_{ij} be the transition probability from state i to state j. The probability that the HMM generates Y_t in state S_t at time t is denoted as $b_{i,t} = P(Y_t|S_t = i)$. Algorithms [19] allow one to recursively compute the following probabilities for partial sequences (up to time t, from time $t+1$ on), with appropriate boundary conditions assumed:

$$\alpha_{i,t} = P\left(Y_1^t \text{ and } S_t = i|\text{model}\right) = b_{i,t}\sum_j a_{ji}\alpha_{j,t-1}$$
$$\beta_{i,t} = P\left(Y_{t+1}^T|S_t = i \text{ and model}\right) = \sum_j a_{ij}b_{j,t+1}\beta_{j,t+1}. \tag{1}$$

If the task is to model isolated units (e.g., isolated words), there will be multiple models ω, one for each unit. In the case of continuous speech recognition, unit models (e.g., phones) are concatenated to make word or sentence models. The likelihood that an HMM has generated the observation corresponding to the pronunciation of the model ω is $L_\omega = \alpha_{F_\omega,T}$, where F_ω is the final state for model ω. HMM parameters can be estimated with different criteria. Two popular criteria are maximum likelihood (ML) and maximum mutual information (MMI). Modeling with these two criteria is discussed in [20]. The mutual information between the model c corresponding to the pronounced sequence of units and the observation Y_1^T is

$$I = \log\left(\frac{P\left(Y_1^T, \text{model}_c\right)}{P\left(Y_1^T\right)P(\text{model}_c)}\right) = \log\left(\frac{P\left(Y_1^T|\text{model}_c\right)}{\sum_\omega P\left(Y_1^T|\text{model}_\omega\right)P(\text{model}_\omega)}\right). \tag{2}$$

Maximum likelihood estimation (MLE) is based on the maximization of the criterion C, expressed as $C_{\text{MLE}} = L_c$, where c represents the pronounced sequence of units. Let us define

$$H_{\text{isolated}} = \frac{L_c}{\sum_\omega L_\omega}. \tag{3}$$

In the case of maximum mutual information estimation (MMIE) for isolated unit modeling, the following criterion can be used:

$$C_{\text{MMIE}} = \log\left(H_{\text{isolated}}\right) = \log\left(\frac{L_c}{\sum_\omega L_\omega}\right). \tag{4}$$

Assuming equal prior probabilities for each model, maximizing C_{MMIE} as in (4) also maximizes the mutual information I.

For continuous speech, we assume that there is a single HMM built by concatenating unit models. During *training*, we consider a constrained model τ that is made of the concatenation of the units that form the training sentence. On

the other hand, during *recognition* all the transitions from one unit to another are possible and we use an unconstrained model ρ, for example a loop model (see [2]). Hence, for continuous speech, C_{MMIE} can be expressed as

$$C_{\mathrm{MMIE}} = \log\left(H_{\mathrm{continuous}}\right) = \log\left(\frac{L_\tau}{L_\rho}\right), \quad \text{where } H_{\mathrm{continuous}} = \frac{L_\tau}{L_\rho}. \tag{5}$$

$L_\tau = \alpha_{F_\tau,T}$ denotes the likelihood of the training model and $L_\rho = \alpha_{F_\rho,T}$ denotes the likelihood of the recognition model. By optimizing one of the above described criteria with the hybrid system, we can replace the usual least mean square criterion and direct supervision for the ANN with a supervision which is derived from the temporal modeling in the HMM.

Assume $b_{i,t}$ can be represented by Gaussian mixtures as follows:

$$b_{i,t} = \sum_k \frac{Z_k}{\left((2\pi)^n |\sum_k|\right)^{1/2}} \cdot \exp\left(-\frac{1}{2}(Y_t - \mu_k)\sum_k^{-1}(Y_t - \mu_k)^T\right) \tag{6}$$

where n is the number of observation features of the HMM. The transition probabilities a_{ij}, normal distribution mean vectors μ_k, covariance matrices $\sum_k$, and gains Z_k can be estimated as in [19]. A derivative of the cost function C with respect to $b_{i,t}$ can be computed and used for estimating the parameters of the ANN, as will be shown in the next section.

IV. Estimation of ANN Parameters

As the optimization criterion C depends on the parameters Y_1^T computed by the ANN, it is possible to express C as a function of them and derive the following equation, using the chain rule:

$$\frac{\partial C}{\partial Y_{jt}} = \sum_i \frac{\partial C}{\partial b_{i,t}} \frac{\partial b_{i,t}}{\partial Y_{jt}} \tag{7}$$

for all the ANN output units j (Y_{jt} being the jth element of the network output vector Y_t). The negative of this gradient can be used with back-propagation[1] to estimate the ANN weights w_{mn}. In the case of MLE, the derivative of the criterion C_{MLE} with respect to $b_{i,t}$ is simply

$$\frac{\partial C_{\mathrm{MLE}}}{\partial b_{i,t}} = \frac{\partial L_{\mathrm{model}}}{\partial b_{i,t}} = \frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial b_{i,t}} \tag{8}$$

where *model* is the training model (the correct unit model, in the case of isolated units modeling). In the case of MMIE, the gradient of the optimization criterion C_{MMIE} with respect to the observation probabilities $b_{i,t}$ can be expressed as

$$\frac{\partial C}{\partial b_{i,t}} = \frac{1}{H}\frac{\partial H}{\partial b_{i,t}} \tag{9}$$

where H is defined as in (2) and (5) for isolated and continuous speech modeling, respectively. In the case of isolated unit modeling, for states i that are in a unit model ω, the following holds:

$$\frac{\partial H_{\mathrm{isolated}}}{\partial b_{i,t}} = \frac{(\delta_{c\omega} - H_c)}{\sum_\omega L_\omega}\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial b_{i,t}}. \tag{10}$$

For continuous speech, we have the following derivative:

$$\frac{\partial H_{\mathrm{continuous}}}{\partial b_{i,t}} = \frac{1}{\alpha_{F_\rho,T}}\frac{\partial \alpha_{F_\tau,T}}{\partial b_{i,t}} - \frac{\alpha_{F_\tau,T}}{\alpha_{F_\rho,T}^2}\frac{\partial \alpha_{F_\rho,T}}{\partial b_{i,t}}. \tag{11}$$

In general, for every optimization criterion C that can be expressed as a differentiable function of the likelihood L, it is possible to compute $\partial C/\partial L$. By differentiating (6), $\partial b_{i,t}/\partial Y_{jt}$ can be expressed as follows:

$$\frac{\partial b_{i,t}}{\partial Y_{jt}} = \sum_k \frac{Z_k}{\left((2\pi)^n|\sum_k|\right)^{1/2}}\left(\sum_l d_{k,lj}(\mu_{kl} - Y_{lt})\right) \cdot \exp\left(-\frac{1}{2}(Y_t - \mu_k)\sum_k^{-1}(Y_t - \mu_k)^T\right) \tag{12}$$

where $d_{k,lj}$ is the element (l,j) of the inverse of the covariance matrix $\left(\sum_k^{-1}\right)$ for the kth Gaussian distribution and μ_{kl} is the lth element of the kth Gaussian mean vector μ_k. Then, following Bridle [15], it is possible to compute the following derivative using (1) for any hidden Markov model, where *model* is ω for isolated unit modeling, or ρ (recognition model) or τ (training model) for continuous speech modeling:

$$\begin{aligned}\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial b_{i,t}} &= \frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{i,t}}\frac{\partial \alpha_{i,t}}{\partial b_{i,t}} \\ &= \left(\sum_j \frac{\partial \alpha_{j,t+1}}{\partial \alpha_{i,t}}\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{j,t+1}}\right)\left(\sum_j a_{ji}\alpha_{j,t-1}\right) \\ &= \left(\sum_j b_{j,t+1}a_{ij}\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{j,t+1}}\right)\left(\sum_j a_{ji}\alpha_{j,t-1}\right) \\ &= \beta_{i,t}\frac{\alpha_{i,t}}{b_{i,t}}.\end{aligned} \tag{13}$$

The equality on the previous line can be justified with the recursive definition of $\beta_{i,t}$ (see (1)), which is the same as the recursive computation of $\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{i,t}}$:

$$\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{i,t}} = \sum_j a_{ij}b_{j,t+1}\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{j,t+1}} \tag{14}$$

with

$$\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{F_{\mathrm{model}},T}} = \beta_{F_{\mathrm{model}},T} = 1 \tag{15}$$

so we have:

$$\frac{\partial \alpha_{F_{\mathrm{model}},T}}{\partial \alpha_{i,t}} = \beta_{i,t}. \tag{16}$$

In summary, the computation of the gradient of a training criterion for HMM's with respect to the parameters of the ANN has been introduced. In particular we have considered the MLE and the MMIE criterion for both isolated and continuous speech models. However, using the MLE criterion may yield

[1] It replaces the usual $\partial E/\partial Y_{jt} = (Y_{jt} - \mathrm{target}_{jt})$ for output units, as used in [3], where target_{jt} would be the desired output at time t for unit j.

a situation in which the likelihood is at a maximum but the hybrid is not doing any useful computation. This could occur if the ANN produced a constant output $\boldsymbol{Y}$ and the HMM Gaussians were all merged into a single distribution with mean $\boldsymbol{Y}$ and zero variance. This problem is avoided if the optimization criterion is the maximization of the likelihood of the *inputs* of the ANN (rather then its outputs). This approach is studied in [21].

In order to implement an ANN/HMM hybrid system, the following methodology was applied. First, ANN's were trained to recognize phonetically relevant features, such as place and manner of articulation. Second, the output vector of these networks was compressed by principal components analysis, in order to provide a smaller size input vector for the HMM. Third, a first estimation iteration computed initial values for the HMM parameters, keeping the ANN's fixed. Finally, the global optimization procedure was applied in order to tune both the HMM and the ANN parameters. In the next section, an application of this algorithm is described in more detail.

V. Experimental Results

A preliminary experiment has been performed using a prototype system based on the integration of ANN's with HMM's. Because of the simplicity of the implementation of the hybrid trained with MLE, this criterion was used in these experiments. Although such an optimization may theoretically worsen performance, we observed significant improvement after the final global tuning. This may be explained by the fact that a nearby local maximum of the likelihood is attained from the initial starting point based on prior and separate training of the ANN and the HMM.

The purpose of the experiment is to show the benefits of global optimization and of the use of suitable parameters for characterizing plosive phonemes. An effort is in progress to introduce and evaluate parameter sets suitable for other phoneme classes. The task is the recognition of plosive phonemes in every context and pronounced by a large speaker population. The 1988 version of the TIMIT continuous speech data base [17] has been used for this purpose. SI and SX sentences from regions 2, 3, and 6 were used, with 1080 training sentences and 224 test sentences, 135 training speakers, and 28 test speakers.[2] The following eight classes have been considered: /p/, /t/, /k/, /b/, /d/, /g/, /dx/,[3] /all other phones/. Speaker-independent recognition of plosive phonemes in continuous speech is a particularly difficult task because these phonemes are made of short and nonstationary events that are often confused with other acoustically similar consonants or may be merged with other unit segments by a recognition system.

As discussed in [22], speech knowledge is used to design the input, output, and architecture of the system and of each of the networks. The ANN's were trained with back-propagation and on-line weight update [3]. The nonlinearity of the hidden units was a symmetric sigmoid while that of the output units was a nonsymmetric sigmoid.

[2]The training speakers were those with initial between "a" and "r" inclusively; the remaining speakers were used for test.

[3]The flapped alveolar plosive /dx/ is considered a distinct phone in the TIMIT data base.

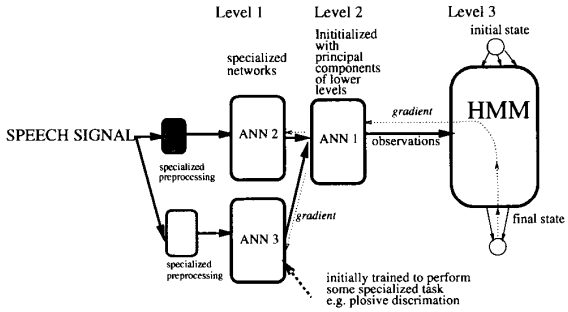


Fig. 2. Extension of the ANN/HMM hybrid to a hierarchy of modules, with three levels.

The experimental system is based on the scheme shown in Fig. 2. Rather than having a single ANN that computes the vector $\boldsymbol{Y}$ of parameters, we have a hierarchy of networks. Such an architecture is built on three levels. Input parameters are fed to the networks every 5 ms. At level 1, two ANN's are initially trained to perform plosive recognition (ANN3) and broad classification (ANN2) respectively. In the experiment described below, the combined network (ANN1 + ANN2 + ANN3) has 23 578 weights. Level 2 is made of a single ANN (ANN1) that acts as an integrator of parameters generated by the specialized ANN's of level 1. ANN1 is a linear network that initially computes the principal components of the concatenated output vectors of the lower level networks (ANN2 and ANN3). Level 3 contains the HMM's. In the following, we describe in some detail the input parameters and the encoding of the output nodes for each network. The approach that we have taken is to select different input parameters and different ANN architectures depending on the phonetic features to be recognized.

The broad classification net (ANN2) has five outputs, corresponding to five broad categories.[4] The 12 input nodes to ANN2 are the energies of five band-pass filters in the time domain covering the range up to 7 kHz, the signal total energy, and their six time derivatives. The filters were IIR (infinite impulse response) Butterworth band-pass filters with the following -3 dB bandwidth specifications: 150–350 Hz, 60–500 Hz, 500–2500 Hz, 2500–3500 Hz, and 4000–7000 Hz. The nonlinear phase response of the filters was not corrected. For the total energy and for the filters in the 150–350 and 60–500 Hz bands, an input window of 20 ms was used. A window of 5 ms was used for every other filter. The filter bandwidths were chosen based on acoustic-phonetics knowledge (see, for example, [23] and [24]). This input feature representation was found to perform better than other spectral representations based on the computation of energies from the fast Fourier transform of a fixed analysis window.

ANN2 has four fully connected layers (12–30–15–5) and time-delay links: from the input at frame t and frame $t-20$ ms to the first hidden layer, and from the second hidden layer at frame t and $t-20$ ms to the output layer. It was found that recurrence did not help for the performance. There were

[4]Nonnasal sonorant, nasal, plosive, fricative, and silence.

also direct links without any delay from the input layer to the second hidden layer and the output layer, and from the first hidden layer to the output layer. This architecture was optimized after a certain number of trials. The frame error rates obtained after parameter estimation of this network alone were 17.7% on the test set and 17.6% on the training set.

The plosive recognition net (ANN3) has 16 outputs, corresponding to place, manner, and degree of voicing, with different instantiations of each place of articulation depending on the right context.[5] The 74 inputs to ANN3 are the outputs of 32 Bark-scaled (logarithmic) triangular filters computed from the short-time fast Fourier transform of the windowed signal; 30 property detectors approximating a second-order derivative over short intervals of frequency and time;[6] seven slope coefficients describing the frequency derivative of the spectrum, the total energy and the voicing energy (in the 60–500 Hz band), and their time derivatives; and a measure of distance (dot product) between neighboring spectral frames. This particular selection of input parameters is the fruit of some preliminary experiments [26]. In general, we have found that using many correlated input parameters and using specialized ANN topologies with such a distributed output encoding improves both the phonetic classification performance and the convergence rate of the learning algorithm, compared with using the spectrogram as only input and the more traditional "on output node per phoneme" encoding.

The topology of ANN3 was optimized after a certain number of trials. Essentially it was a two-hidden-layers network with delays, with the addition of recurrent connections between the output layer and the second hidden layer, as shown in Fig. 3. At time t, three input frames (t, $t-15$, and $t-30$ ms) were used as input to the first hidden layer. To limit the number of parameters to be estimated, these input-to-hidden connections were localized in frequency. This means that the first hidden layer was divided into small groups of a dozen nodes, each being connected to a limited portion of the input vector.

ANN1 computes eight features for the continuous densities HMM. Each of the 11 unit models[7] has 14 states, 28 transitions, and three self-loops, without explicitly modeling the state duration, as shown in Fig. 4. Each HMM has tied distributions with three basic different distributions characterizing the beginning, middle, and final part of a segment modeled by the unit. Each of these distributions is modeled by a Gaussian mixture with five densities. The covariance matrix is assumed to be diagonal since the parameters are initially principal components and this assumption reduces significantly the number of parameters to be estimated. The hybrid system was trained in the final tuning step according to the equations

[5] Each of the four different places of articulation (labial, alveolar, velar, and flapped alveolar) corresponds to two different nodes, depending on whether the following phone has a front or nonfront place of articulation. The remaining eight nodes are labeled: unvoiced plosive, voiced plosive, vocalic front, vocalic nonfront, liquid, fricative, nasal, silence.

[6] This parameter is inspired by studies in acoustic-phonetics [25].

[7] In order to improve its modeling, the rejection class was composed out of four models: nasals, fricatives, nonnasal sonorants, and silence. The recognition results are obtained by merging these four subclasses, such that the total number of classes to recognize is eight.

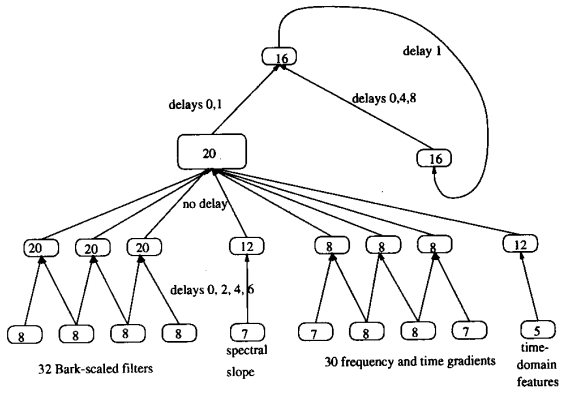


Fig. 3. Network architecture used for the recognition of plosives (ANN3).

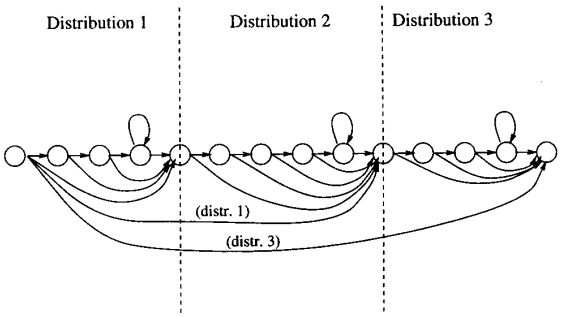


Fig. 4. Topology of the HMM's used in the experiments. Distributions are associated with transitions.

for continuous speech with the MLE criterion described in Sections III and IV.

In order to assess the value of the proposed approach as well as the improvement brought by the HMM as a postprocessor for time alignment, the performance of the hybrid system was evaluated and compared with that of a simple postprocessor applied to the outputs of the ANN's and with that of a standard dynamic programming postprocessor that models duration probabilities for each phoneme. The simple postprocessor assigns a symbol to each output frame of the ANN's by comparing the target output vectors with actual output vectors. It then smoothes the resulting string to remove very short segments and merges consecutive segments that have the same symbol. The dynamic programming (DP) postprocessor finds the sequence of phones that minimizes a cost. This cost depends on the product of the duration probabilities for each phone segment and of the conditional probability of the data (network output) given a phoneme. In the case in which bigram probabilities are also used, the conditional probabilities of a phone given the previous phone are also multiplied in the cost expression. The duration probabilities are modeled by a gamma distribution estimated with the TIMIT labeling for the training set. The observation probabilities are modeled by multivariate normal densities for each phoneme, estimated with the outputs of the network and the corresponding TIMIT labels for the training set.

TABLE I
COMPARATIVE RECOGNITION RESULTS: NEURAL NETWORKS ALONE, WITH DYNAMIC PROGRAMMING, WITH HIDDEN MARKOV MODELS, AND WITH GLOBAL OPTIMIZATION

	% rec	% ins	% del	% subs	% acc
ANN's	85	32	0.04	15	53
ANN's + DP (no bigrams)	88	16	0.01	11	72
ANN's + DP (bigrams)	88	14	0.01	11	74
ANN's + HMM	87	6.8	0.9	12	81
ANN's + HMM + global opt.	90	3.8	1.4	9.0	86

The comparative results for the three systems are summarized in Table I. The overall recognition rate (100% − % deletions − % substitutions) for the eight classes with hybrid system after two training iterations is 90% on a total of 7214 phones, and its accuracy (100% − % deletions − % substitutions − % insertions) is 86%. This is a significant improvement over the performance obtained with an HMM trained without global optimization (86% recognition and 80% accuracy), as well as with respect to the two DP systems (88% recognition and 72% accuracy without bigrams and 88% recognition and 74% accuracy with bigrams). The biggest improvement with respect to the ANN's comes from modeling the durations rather than the bigrams. The ANN's alone yielded 85% recognition but only 53% accuracy, because of the high number of insertions (32%), mostly because of short plosive segments. The ANN's perform a good classification but have a noisy output with many insertions. The HMM or DP duration modeling eliminates most of these insertions because of their better duration and temporal structure modeling. With global optimization, in addition to providing a good temporal model, the HMM provides more appropriate target values for the outputs of the ANN. With these target outputs for the ANN, the hybrid system significantly improves its performance. It is interesting to note that the effect of (7) and (12) is to generate a gradient that tends to bring the output of the ANN closer to the means of the normal densities which are close to the ANN output as well as consistent with the training string. This tends to reduce the variance of the ANN outputs with respect to those means, while allowing for a richer set of target vectors (Gaussian means) than the usual ANN supervision.

Our previous experience as well as other results [27], [28] indicates that on-line update of ANN weights yields faster convergence than batch update, especially for pattern recognition problems such as those in speech recognition. Comparative experiments performed with the hybrid system indicate that the on-line update for HMM parameters as well seems to yield better results. In Table II, the two update methods for the HMM parameters within the hybrid system are compared. Traditionally, the HMM parameters are updated after having compiled statistics over the whole training set. The alternative update method used in the experiments is a smoothed on-line parameter update:

$$\theta_{i,p} = (1-\alpha)\theta_{i,p-1} + \alpha\hat{\theta}_{i,p} \tag{17}$$

where $\theta_{i,p}$ is the new value of parameter i after sentence p, α is a small constant,[8] and $\hat{\theta}_{i,p}$ is the estimation of the parameter θ_i given the observations in sentence p, using usual HMM parameter estimation algorithms [19]. Table II also shows the evolution of generalization errors after one and two training iterations of the hybrid system with global optimization. In the experiments, a minimum of the error was reached after only two iterations. Further training only reduced generalization.

TABLE II
GENERALIZATION OF THE ANN/HMM HYBRID SYSTEM AS A FUNCTION OF THE NUMBER OF TUNING ITERATIONS AND THE HMM PARAMETER UPDATE METHOD

	% rec	% ins	% del	% subs	% acc
Iteration 0	87.6	6.8	0.9	11.5	80.7
Iteration 1 (batch)	87.1	3.6	2.2	10.7	83.5
Iteration 2 (batch)	87.7	3.8	1.9	11	83.4
Iteration 1 (on-line)	89.5	4.0	1.3	9.2	85.5
Iteration 2 (on-line)	89.6	3.8	1.4	9.0	85.8
Iteration 3 (on-line)	87.6	3.6	2.4	10	84.0

VI. CONCLUSION AND EXTENSIONS

A system has been proposed to combine the advantages of ANN's and HMM's for speech recognition. The parameters of the ANN and HMM subsystems can influence each other. We showed how to perform a global optimization of such a system by driving the network gradient descent with parameters computed in the HMM. Encouraged by the results of the above-described initial experiments, which indicate that global optimization of a hybrid ANN–HMM system gives some significant performance benefits, we will explore further the possibilities of such a hybrid system and extend it to the recognition of all American-English phonemes. We have seen how such a hybrid system could integrate multiple ANN modules, which may be recurrent.

An interesting extension would be to perform speaker adaptation with the hybrid system. This could be obtained by first training the system as previously described for multiple speakers, and in a second step, adapting *only the ANN parameters* with sentences from the new speaker. In such a system, the ANN adaptation represents a tuning of the feature space to the new speaker, whereas the temporal model remains unchanged (see [29] for a related speaker adaptation mechanism).

Another extension would be to replace the linear transformation performed in the second level (principal components) by a network with a hidden layer and symmetric sigmoids. This network could still be initialized to compute the principal

[8] We used $\alpha = 0.005$, except for the variances of the observation distributions which were updated with a semibatch algorithm, because the estimation of the second moment of the distributions requires more observations:

$$\sigma_{i,p} = \left(1 - \frac{p}{N}\right)\sigma_{i,p-1} + \frac{p}{N}\hat{\sigma}_{i,1,p},$$

where N is the number of sentences, and $\hat{\sigma}_{i,1,p}$ is the estimation of the parameter σ_i given all the observations from sentence 1 to sentence p, using standard HMM parameter estimation algorithms [19]. This method forces a slow initial adaptation of the variances but computes their final value using all the training data.

components of the outputs of the first level. This can be obtained by decomposing the principal components matrix into the product of two matrices, e.g., with LU decomposition [30], and multiplying the hidden layer weights by a small constant ϵ. Because the symmetric sigmoid is linear around 0, the network initially computes principal components, but it can be adapted with back-propagation and perform a nonlinear transformation after training.

Although the ANN's used in the experiments were recurrent, they did not capture the temporal structure of the speech signal as well as the hybrid system or the DP postprocessor. Notice that very few parameters were used in the HMM or the DP postprocessors to describe the temporal structure of the observations (transition probabilities or duration probabilities, respectively). This may indicate that current ANN topologies and related algorithms are inefficient in modeling temporal structures. It should be observed that HMM's generally used for speech recognition have a left-to-right structure rather than a full connectivity from state to state. It may be possible to improve the way in which temporal structures are modeled in ANN's by imposing appropriate constraints on their architecture for the particular problem of learning to recognize sequences.

REFERENCES

[1] L. R. Rabiner and S. E. Levinson, "A speaker-independent, syntax-directed connected word recognition system based on hidden Markov models and level building," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 3, pp. 561–573, 1985.

[2] K. F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, pp. 1641–1648, 1989.

[3] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representation by error propagation," in *Parallel Distributed Processing* vol. 1. Cambridge, MA: MIT Press, 1986, pp. 318–362.

[4] Y. Bengio, R. Cardin, R. De Mori, and E. Merlo, "Programmable execution of multi-layered networks for automatic speech recognition," *Commun. Ass. Comput. Mach.*, vol. 32, no. 2, pp. 195–199, Feb. 1989.

[5] R. P. Lippman, "Review of neural networks for speech recognition," *Neural Computation*, vol. 1, no. 1, pp. 1–38, 1989.

[6] P. Cosi, Y. Bengio, and R. De Mori, "Phonetically-based multi-layered networks for acoustic property extraction and automatic speech recognition," *Speech Commun.* (special issue on neurospeech), vol. 9, no. 1, pp. 15–30, 1990.

[7] T. Robinson and F. Fallside, "Phoneme recognition from the TIMIT database using recurrent error propagation networks," Engineering Dept., Cambridge University, CUED/F-INFENG/TR 42, 1990.

[8] Y. Bengio, R. Cardin, R. De Mori, and Y. Normandin, "A hybrid coder for hidden Markov models using a recurrent neural network," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* (Albuquerque, NM), Apr. 1990, pp. 537–540.

[9] H. Bourlard and C. J. Wellekens, "Links between Markov models and multilayer perceptrons," in *Advances in Neural Information Processing Systems 1*, (D. S. Touretzky, Ed. Los Altos, CA: Morgan Kauffman, 1988, pp. 502–510.

[10] M. Franzini, K.-F. Lee, and A. Waibel, "Connectionist Viterbi training: A new hybrid method for continuous speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* (Albuquerque, NM), April 1990, pp. 425–428.

[11] N. Morgan and H. Bourlard, "Continuous speech recognition using multilayer perceptrons with hidden Markov models," in *Proc. Int. Conf. Acoust., Speech, Signal Processing* (Albuquerque, NM), Apr. 1990, pp. 413–416.

[12] L. T. Niles and H. F. Silverman, "Combining hidden Markov models and neural network classifiers," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* (Albuquerque, NM), Apr. 1990, pp. 417–420.

[13] S. J. Young, "Competitive training in hidden Markov models," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* (Albuquerque, NM), Apr. 1990, pp. 681–684.

[14] J.-N. Hwang, J. A. Vlontzos, and S.-Y. Kung, "A systolic neural network architecture for hidden Markov models," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. 37, pp. 1967–1979, Dec. 1989.

[15] J. S. Bridle, "Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters," in *Advances in Neural Information Processing Systems 2*, (D. S. Touretzky, Ed. Los Altos, CA:) Morgan Kauffman, 1990, pp. 211–217.

[16] E. Levin, "Word recognition using hidden control neural architecture," in *Proc. Int. Conf. Acoust. Speech, Signal Process.* (Albuquerque, NM), Apr. 1990, pp. 433–436.

[17] V. Zue, S. Seneff, and J. Glass, "Speech database development: TIMIT and beyond," *Speech Commun.*, vol. 9, no. 4, pp. 351–356, Aug. 1990.

[18] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Statistic.*, vol. 41, pp. 164–171, 1970.

[19] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–285, Feb. 1989.

[20] A. Nadas, D. Nahamoo, and M. A. Picheny, "On a model-robust training method for speech recognition," *IEEE Trans. Acoust. Speech, Signal Process.*, vol. 36, no. 9, pp. 1432–1436, 1988.

[21] Y. Bengio, "Artificial neural networks and their application to sequence recognition," Ph.D. thesis, Dept. Computer Science, McGill University, Montreal Canada, 1991.

[22] Y. Bengio, R. De Mori, and R. Cardin, "Speaker independent speech recognition with neural networks and speech knowledge," in *Advances in Neural Information Processing Systems 2*, D. S. Touretzky, Ed. Los Altos, CA: Morgan Kaufmann, 1990, pp. 218–225.

[23] D. O'Shaughnessy, *Speech Communication—Human and Machine.* Reading, MA: Addison Wesley, 1987.

[24] K. N. Stevens and S. E. Blumstein, "The search for invariant acoustic correlates of phonetic features," in *Perspectives on the Study of Speech*, P. D. Eimas and J. L. Miller eds. Hillsdale, NJ: Lawrence Erlbaum, 1981, pp. 1–38.

[25] K. N. Stevens, "The potential role of properties detectors in the perception of consonants," in *Auditory Analysis and Perception of Speech*, G. Fant and M. A. Tatham, eds. London: Academic Press, 1975, pp. 303–330.

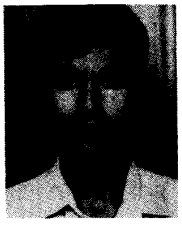
[26] Y. Bengio, R. De Mori, G. Glammia, and R. Kompe, "Phonetically motivated acoustic parameters for continuous speech recognition using artificial neural networks," in *Proc. Eurospeech 91* (Genova, Italy), 1991.

[27] L. Bottou, F. Fogelman, P. Blanchet, and J. S. Lienard, "Speaker-independent isolated digit recognition: Multilayer perceptrons vs. dynamic time warping," *Neural Networks*, vol. 3, no. 4, pp. 453–465, 1990.

[28] Y. Le Cun, "Generalization and network design strategies," in *Connectionism in Perspective*, Pfeifer, Schreter, Fogelman, and Steels, eds. New York: North Holland, 1989, pp. 143–155.

[29] J. S. Bridle and S. J. Cox, "RECNORM: Simultaneous normalization and classification applied to speech recognition," in *Advances in Neural Information Processing Systems 3*, D. S. Touretzky, Ed. Los Altos, CA: Morgan Kauffman, 1991.

[30] G. W. Steward, *Introduction to Matrix Computations.* London: Academic Press, 1973.



Yoshua Bengio was born in Paris in 1964. He received the B.Eng. degree in electrical engineering in 1986 and the M.Sc. and Ph.D. degrees in computer science in 1988 and 1991, respectively, all from McGill University.

He is currently with the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge. His research has focused on the combination of domain knowledge and learning from examples in artificial neural networks for sequence recognition, as well as on learning algorithms for recurrent networks and the combination of artificial neural networks with hidden Markov models. His interests also include the optimization of biologically constrained synaptic learning rules. He has been the recipient of National Science and Research Council of Canada postgraduate and postdoctorate scholarships.

Renato De Mori (M'83–SM'89) was born in Milan, Italy, in 1941 and received a doctorate degree in electronic engineering from the Politecnico di Torino, Torino, Italy, in 1967.

He became full professor in Italy in 1975. Since 1986, he has been Professor and the Director of the School of Computer Science at McGill University, Montreal, Quebec, Canada. Since 1987, he has been Vice-President of the Centre de Recherche en Informatique de Montréal, a research center involving seven universities and more than 40 industries. In 1991, he become an associate of the Canadian Institute for Advanced Research and project leader of the Institute for Robotics and Intelligent Systems, a Canadian Center of Excellence. He is the author of many publications in the areas of computer systems, pattern recognition, artificial intelligence, and connectionist models. His research interests are now stochastic parsing techniques, connectionist models, and reverse engineering.

Dr. De Mori has been member of various committees in Canada, Europe, and the United States; he has served as an associate editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and is now on the board of the following international journals: *Signal Processing, Speech Communication, Pattern Recognition Letters, Computer Speech and Language,* and *Computational Intelligence.*

Giovanni Flammia (S'90) received the M.S. degree in electrical engineering from the INFOCOM Department, La Sapienza University, Rome, Italy, in 1988. He then did research on speech analysis at the Centre National d'Études de Télécomunications, Lannion, France, for a semester. In 1991 he obtained the M.S. degree in computer science from McGill University, Montréal, Canada. He is now a guest researcher at the Speech Technology Center of the University of Aalborg, Denmark. His interests include speech communication, pattern recognition, statistics, and artificial neural networks.

Ralf Kompe was born in Karlsruhe, Germany, in 1963. He received a diploma degree in computer science from the University Erlangen-Nürnberg, Erlangen, Germany, in 1989. From September 1989 to December 1990 he was a research assistant at the School of Computer Science of McGill University, Montreal, Canada.

Since March 1991, he has been with the speech group of the Institute for Pattern Recognition at the University Erlangen-Nürnberg. His current research interests are speech recognition, knowledge-based speech understanding, natural language, and neural networks.