SLHMM: A CONTINUOUS SPEECH RECOGNITION SYSTEM BASED ON ALPHANET-HMM.

J.E Díaz-Verdejo, J.C. Segura-Luna, P. García-Teodoro, A.J. Rubio-Ayuso

Dpto. Electrónica y Tecnología de Computadores. Facultad de Ciencias.
Universidad de Granada. 18071 - GRANADA (Spain)

ABSTRACT

This paper presents a new framework developed to apply Alphanets to CSR. For this purpose, a modular system is proposed. This system is made up by three different modules: LVQ module, SLHMM module and DP module. The SLHMM module is an expansion of an Alphanet, and therefore, can be interpreted as a HMM. The system can be trained globally applying backpropagation techniques. The used prunning procedure is based upon recognized units instead of observations, which reduces the number of nodes needed to recognize a sentence, compared to HMM-based systems using the same parameters for the models in both systems. Besides, the training procedure re-adapts the weights according to the new architecture in a few iterations since the initial parameters can be estimated from a classical HMM CSR system.

1. INTRODUCTION

Sime recent works [1],[2] have shown strong relations between the RNNs and the well-known HMMs. In fact, this work proved that a RNN, with an appropriate topology, is equivalent to a HMM.

Our purpose is to use RNNs as the kernel of an ANN-based continuous speech recognition system maintaining a HMM meaning. This fact can allow us to improve the performance of both systems, the ANN-based and the HMM-based, by applying ANN techniques to HMM and vice-versa. On the other hand, if we maintain the HMM meaning for this module of the recognizer, we are able to train its parameters with usual HMM algorithms, simpler than those of ANN, and simply download this set of parameters into the net.

In order to recognize continuous speech using ANN, we have to evaluate probabilities of the speech sequence locally, due to the fixed structure of the net. This problem arises when using ANN since it is not possible to construct a "super-model" of the speech sequence by simply concatenating elementary models, as usually done in HMM-based systems.

To solve this problem we have developed a formalism which is able to obtain the optimal path from the set of local probabilities. This formalism allows us to develop a modular system made up by several types of ANN that can be trained globally using standard ANN algorithms as backpropagation.

*This work has been supported by Spanish CICYT under project TIC92-0662.

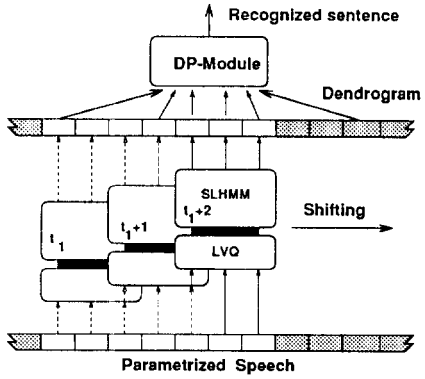

Figure 1. Temporal sliding of a SLHMM over a speech sequence.

2. SYSTEM STRUCTURE

The proposed system is composed by 3 main modules (fig. 1). The modules are: a LVQ module, a SLiding HMM (SLHMM) module and a DP module. The first ones are directly ANN. The third can be implemented as an ANN, although it is not trainable.

The LVQ module is an ordinary quantizer, although the system structure allows a training procedure for this quantizer that optimizes the whole system performance. Besides, this structure also allows the use of MVQHMM [3,4] in an easy way. In a MVQHMM recognizer, each HMM has its own VQ codebook, and the recognition is performed taking into account the distortion measure and the generation probabilities. In a first phase, the LVQ module has been replaced by a standard VQ because it does not affect the kernel of the system and, at this moment, our interests focuss on developing the SLHMM and DP modules.

The SLHMM module is the kernel of the system. Its structure will be described in detail in the next section. The SLHMMs are used to obtain the sets of all the possible segmentations of the sentence (*dendrogram*) by simply evaluating the partial probabilities of the subsequences of symbols that composes it.

0-7803-1775-0/94 $3.00 © 1994 IEEE

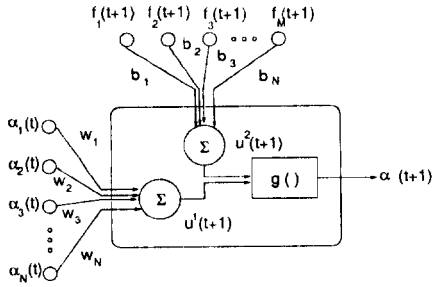

Figure 2. Generalized neuron used to build a SLHMM

The DP module is used to obtain the optimal path from the dendrogram by applying Dynamic Programming techniques. This optimal path along the dendrogram will yield the recognized sentence.

3. THE SLHMM

Basically, a SLHMM consists in a development of an Alphanet [2] obtained by expanding it into a fixed number of layers. In this way, each temporary step in the evaluation of an Alphanet has been replaced by a layer in the SLHMM structure. Obviously, this expansion of the net imposes temporary constraints: the maximum length of the sequence being evaluated is the number of layers in the SLHMM.

Each SLHMM stands for one of the selected recognition units (i.e. phonemes), and so, the SLHMM-module is built up with so many SLHMMs as different units are being considered.

The SLHMM is composed by generalized neurons [2], depicted in fig. 2. This neuron can be used to implement a Discrete or a Semicontinuous Alphanet, and so, the SLHMM can be used for both Discrete and Semicontinuous versions.

The neuron processing equations for the i-th neuron in the $l+1$ layer are:

$$\left.\begin{aligned} u_i^{(1)}(l+1) &= \sum_{j=1}^{N(l+1)} w_{ij}(l+1)\cdot\alpha_j(l) \\ u_i^{(2)}(l+1) &= \sum_{j=1}^{M(l+1)} b_{ij}(l+1)\cdot f_j(l+1) \\ \alpha_i(l+1) &= u_i^{(1)}(l+1)\cdot u_i^{(2)}(l+1) \end{aligned}\right\} \quad (1)$$

where $N(l+1)$ is the number of neurons and $M(l+1)$ the number of external inputs $f_j(l+1)$, obtained from the observed vector, at layer $l+1$. $w_{ij}(l+1)$ and $b_{ij}(l+1)$ are the weights of the links between neurons and inputs.

The processing carried out by a SLHMM is as follows: given an input sequence

$$X_1^T = \{X_1, X_2, \cdots, X_T\} \quad (2)$$

the SLHMM selects a subsequence of $L+1$ symbols starting at an arbitrary time t_1. The input subsequence is

$$X_{t_1}^{t_1+L} = \{X_{t_1}, X_{t_1+1}, \cdots, X_{t_1+L}\} \quad (3)$$

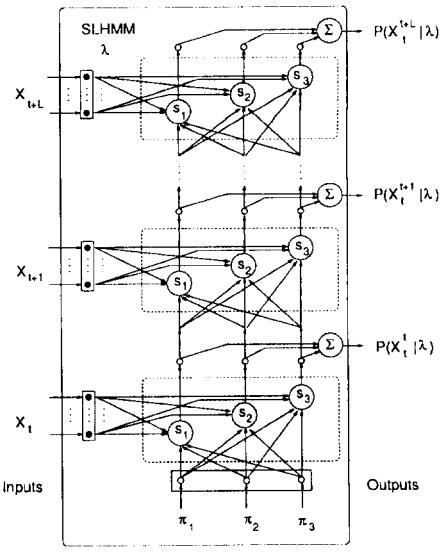

Figure 3. SLHMM scheme.

Each SLHMM presents L outputs. Each of them is the sum of the outputs of the neurons at each of the layers. For the layer l, this output is

$$\sum_{i=1}^{N(l)} \alpha_i(l)$$

If the parameters of the net are properly selected [1,2], the α's are the forward probabilities defined in HMMs. In this way, the output of the SLHMM at layer l is the probability of the subsequence starting at t_1 and ending at t_1+l-1

$$P(X_{t_1}^{t_1+l-1}|\lambda) = \sum_{i=1}^{N(l)} \alpha_i(l) \quad (4)$$

Since the SLHMM has $L+1$ layers, the output is the set of probabilities of all the subsequences starting at time t_1 with maximum length L

$$\{P(X_{t_1}^{t_1}|\lambda), P(X_{t_1}^{t_1+1}|\lambda), \cdots, P(X_{t_1}^{t_1+L}|\lambda)\} \quad (5)$$

The structure of the whole net is shown in fig. 3.

By sliding the net over the whole sequence (fig. 1) we can obtain the dendrogram of the signal since we will evaluate all the probabilities

$$P(X_{t_1}^{t_2}|\lambda) \quad with \quad 0 \le t_1 \le T;\ t_1 \le t_2 \le t_1+L \quad (6)$$

4. THE DP MODULE

Once the dendrogram is obtained, the DP-module is used to obtain the optimal path from it. This optimal path is determined by the number of units, P (i.e. the number of phones), the sequence of optimal units, $\{\lambda_1, \lambda_2, \cdots, \lambda_P\}$ and the final time indexes of these units $\{t_1, t_2, \cdots, t_P\}$.

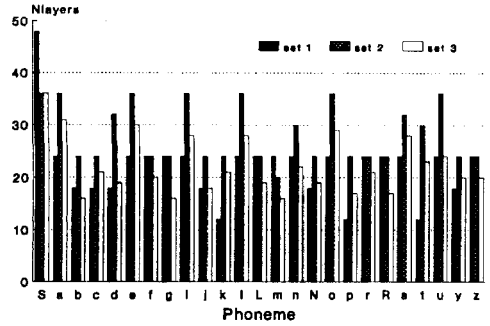

Figure 4. Number of layers in each SLHMM for the 3 sets of models

4.1. Training mode

The behavior of the DP module in training is simpler than for a speech signal evaluation.

In training mode the units that compose a sentence are known. Therefore, the DP module performs a standard DTW procedure in order to obtain the time alignment between the speech input and the elementary units in the sentence. The resulting optimal path is determined by the final time indexes of the units, since the number of units and the units are known.

If we call

$$\mathcal{L}(t,p) \equiv \log P(X_1^t|\lambda_1\lambda_2\cdots\lambda_p) \tag{7}$$

the alignment equations are

- Beginning
$$\mathcal{L}(1,1) = \log P(X_1^1|\lambda_1) \tag{8}$$
- Recursion
$$\mathcal{L}(t,p) = \max_{1\le\Delta t\le L_\lambda}\{\mathcal{L}(t-\Delta t, p-1) + \log P(X_{t-\Delta t+1}^t|\lambda_p)\} \tag{9}$$
- End
$$\log P(X_1^T|\lambda_1,\lambda_2,\cdots,\lambda_P) = \mathcal{L}(T,P) \tag{10}$$

The path obtained from this alignment can be used to train the SLHMM and LVQ modules by applying backpropagation techniques.

4.2. Evaluation mode

In the evaluation mode, the DP module must perform a process quite similar to a Viterbi Beam Search, due to the fact that we do not know neither the number of units in the sentence nor which these units are. Of course, to carry out this process a prunning procedure is needed.

If we call,

$$\mathcal{L}(t,l,\lambda_p) \equiv \log P(X_1^t|\lambda_1\lambda_2\cdots\lambda_p) \tag{11}$$

the recognition procedure is as follows:

- The initialization is
$$\mathcal{L}(1,1,\lambda_l) = \log P(X_1^1|\lambda_l) \tag{12}$$
- Given a known value for $\mathcal{L}(t,p,\lambda_p)$, we evaluate all the possible values, $\mathcal{L}(t+\Delta t, p+1, \lambda_{p+1})$, according to a given grammar
$$\mathcal{L}(t+\Delta t, p+1, \lambda_{p+1}) = \mathcal{L}(t,p,\lambda_l) + \log P(X_t^{t+\Delta t}|\lambda_{p+1}) \tag{13}$$

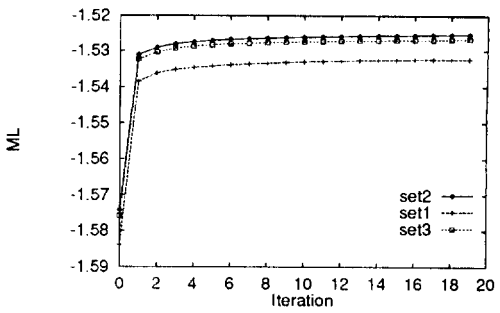

Figure 5. Likelihood vs. training iteration for the three set of SLHMMs.

- In the final state of the grammar and in time step T, the maximum value for $\mathcal{L}(T, P_i, \lambda_{end})$ is chosen and the optimal alignment path is obtained from backtracking.

5. SYSTEM TRAINING

The SLHMM can be trained by applying backpropagation techniques. The backpropagation signal is defined as,

$$\delta_i(t) = \frac{\partial E}{\partial \alpha_i(t)} \tag{14}$$

and can be recurrently computed,

$$\delta_i(t) = \sum_{j=1}^{N} \delta_j(t+1)\cdot u_i^{(2)}(t+1)\cdot w_{ji} \tag{15}$$

The utilized error measures are the Likelihood and a measure very similar to Mutual Information [1], given by the probability of selecting the correct model

$$\mathcal{J} = -\log\left[\frac{P(X_1^T|\lambda_c)}{\sum_\lambda P(X_1^T|\lambda)}\right] \tag{16}$$

where λ_c is the correct model.

To obtain the error for each unit of a sequence, it is necessary to know where this unit starts and ends. This is performed by the DP-module, that obtains the alignment between the sequence and the units. This way, the training procedure consists of two steps:

- a first step in which the sentence is processed by the whole system in order to obtain the alignment,
- and a second step in which each SLHMM-LVQ pair is used to process the subsequence corresponding to each of the units and train them according to the results and the selected measure.

The backpropagation signal can be used to train not only the SLHMM, but also the LVQ associated to it. It is also possible to model units duration by simply allowing the training procedure to adjust the weights of each layer independently. This way, these weights will depend on the layer of the SLHMM, which is equivalent to consider a set of weights that change with time and, obviously, will model the duration probabilities of the units.

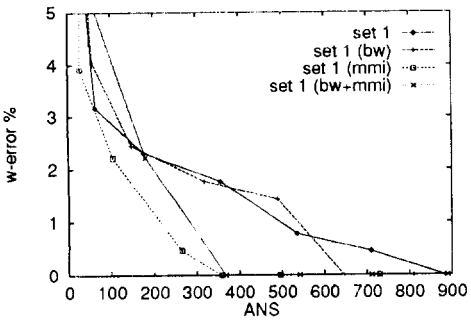

Figure 6. Error rate vs. average number of states (ANS) used to recognize a sentence. The oscillations are due to sentence rejection for small pruning thresholds in the SLHMM-based system.

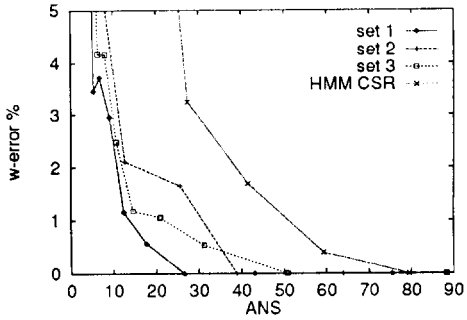

Figure 7. ANS for some pruning thresholds for HMM-based and SLHMM-based systems.

6. EXPERIMENTAL RESULTS

The initial values for the parameters of all the SLHMMs have been estimated from a classical HMM CSR system with the hope of diminishing the number of iterations needed for training.

The system was evaluated with a set of 170 sentences. The data was recorded by 4 speakers (2 males + 2 females). Therefore, a total of 720 sentences are used. A set of 120 sentences was used to train the models, while the other 50 sentences have been used to test the system. The speech signal was segmented into frames of 32 ms overlapped 8 ms, and parametrized with 16 cepstral coefficients, 16 Δcepstral coefficients and the Δenergy. The result is decimated to 16 ms [3]. The perplexity of the word-grammar associated to the set of test sentences is about 1.6. Although small, it can be used to test the behavior of the system and compare it to a HMM-based one.

The recognition units in our experiments are phonemes. A key point to obtain good results is the selection of maximum and minimum duration allowed for each phoneme. The number of layers of the SLHMM associated to a phoneme is the maximum duration the phoneme can have. Besides, the bigger the number of layers, the bigger the computational complexity and the larger decision tree. Therefore, a preliminary experiment was carried out to obtain the durations of the phonemes by simply obtaining the alignments of the training sentences. This way, 3 sets of phoneme durations were used for training and testing experiments. For set 1, a really short number of layers have been selected. For set 2, all the SLHMMs have a number of layers that allows all phoneme durations observed in the training set, while for set 3, the number of layers was chosen to include all durations in the training set, except the 5 longest phonemes. The resulting durations are shown in figure 4. The minimum durations were chosen to be the number of states in the phoneme.

The 3 sets of SLHMMs were trained using a standard BW reestimation algorithm, a MMI descent reestimation algorithm and a combination of both [1,2]. Figure 5 shows the evolution of Likelihood measure vs, training iteration. As it can be seen from this plot, the system re-adapts the parameters of the models to the new architecture used for SLHMM, although the models were previously trained with the same algorithm in a HMM CSR system. All the models so obtained were used to recognize the test sentences. For this test, a sentence is considered valid even when the final word does not correspond to the final state of the grammar . The word error rates for these experiments are shown in figure 7. This figure shows that the Average Number of Active States (ANS) used by the system to recognize a sentence decreases as the models are trained with BW, MMI and BW plus MMI.

The proposed system has the advantage that the prunning procedure is performed on a recognized unit basis, not on a temporary step basis as done with standard HMM CSR systems. Therefore, the prune procedure is asynchronous (the nodes of the decision tree correspond to different time indexes) instead of synchronous (all the nodes correspond to subsequences with the same number of elements). This way, the ANS needed to recognize a sentence is reduced because the depth of the decision tree is the number of units (phonemes) that is much more smaller than the number of symbols, as used for classical systems. Anyway, the number of sons each node has is bigger in the SLHMM-based system, although the reduction in depth is more important and compensates this effect (fig. 6).

REFERENCES.

[1] Bridle, J.S. "Alpha-nets: A Recurrent Neural Network Architecture with a Hidden Markov Model interpretation", Speech Communication, vol. 9, 1990.

[2] Diaz-Verdejo, J. et al. "A New Neuron Model for an Alphanet-Semicontinuous HMM". Proc. ICASSP-93, vol. 1, pp. 529-532, 1993.

[3] Segura-Luna, J.C. "Modelos de Markov con Cuantización Dependiente para Reconocimiento de Voz", Thesis Dissertation. Universidad de Granada, 1990.

[4] Peinado, A. et al. "Using Multiple Vector Quantization and Semicontinuous Hidden Markov Models for Speech Recognition". To be published in Proc. ICASSP-94.