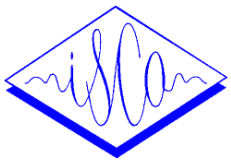

3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

A NEW PROBABILISTIC FRAMEWORK FOR CONNECTIONIST TIME ALIGNMENT

Patrick Haffner
France Télécom, Centre National d'Etudes des Télécommunications
CNET/LAA/TSS/RCP, BP 40, 22301 LANNION, FRANCE
email: haffner@lannion.cnet.fr

ABSTRACT

To build optimally effective word classifiers, one research direction in speech recognition is to train a connectionist architecture with a gradient back-propagation procedure that minimises the word error rate directly. The first step was the integration of the DTW alignment procedure into the architecture: the Multi-State Time Delay Neural Network (MS-TDNN[6]) architecture was successfully demonstrated on several large speech recognition tasks. In this paper, we provide an HMM probabilistic framework for the alignment procedure, with improved experimental results. Moreover, applying a unified HMM/connectionist formalation to global speech recognition systems suggests ways to exchange expertise between both fields.

1. INTRODUCTION

Current speech recognition techniques are mostly described as Hidden Markov Models (HMMs) and Neural Networks (NNs). Despite the fact that they are both based on the training of parameters on large speech corpora, they have different properties. HMMs are based on statistical models which compute the probability of a sequence of states and the corresponding acoustical parameters. This statistical interpretation permits the use of very efficient procedures to compute the optimal alignment path and reinforce the desired sequence of states. However, it relies on strong hypotheses on the distribution of the acoustical parameters. The discriminant training procedure of supervised NNs (for instance MLPs) makes their performance robust to the choice of acoustical parameters, which are not assumed to be produced by a probabilistic model. This absence of data generating model makes current connectionist models much more difficult to interpret under a formalism where temporal patterns are sequences of emitted symbols or vectors. Ideally, one would like to combine those properties. Several approaches attempt to combine Markov Modelling and discriminant/connectionist training procedures:

- HMM/MLP hybrid systems combine independent frame-level connectionist classification and HMM alignment, therefore, this approach lacks a unified formalism[2].
- Discriminant MMI and corrective training techniques are supposed to make HMMs more robust to incorrect modelling assumptions, but they are only a fine tuning procedure[4].
- MS-TDNNs incorporate time alignment into the connectionist architecture, however a probabilistic understanding of the alignment procedure has been lacking[6].

The paper is organized as follows. Section 2 introduces a static pattern classification formalism which exploits links between Gaussian Probability Density Functions (pdfs) and the Multi Layered Perceptron (MLP). Section 3 shows how the formalisms of section 2 provides a probabilistic framework for connectionist time alignment. Section 4 gives experimental results on a speaker independent speech recognition task.

2. THE STATIC PROBLEM

In the simplified case of static pattern classification, we try to combine MLPs and Gaussian pdfs. To use the an MLP as an adaptive pre-processor for a Gaussian mixture (one might equally the Gaussian mixture is a statistical post-processor for the MLP) has been suggested[1] as a way to perform density estimation with Neural Networks. We have a different goal here: to improve our probabilistic understanding of MLPs.

2.1. Links between Gaussian pdfs and MLPs

In this section, we show that there are strong links between the Maximum A Posteriori (MAP) probability criterion applied to Gaussian pdfs and MLPs.

Suppose we have N classes, each class being associated with a spherical Gaussian density, so that the probability (estimated with parameters θ) of the input vector y given the class c_i is:[1]

$$\begin{aligned} \Pr_\theta(Y{=}y|c_i) &= \frac{1}{Z}\cdot e^{-\frac{1}{2\sigma^2}\|y-m_i\|^2} \\ &= \underbrace{\frac{1}{Z}\cdot e^{-\frac{1}{2\sigma^2}\|y\|^2}}_{\text{Independent of } i} \cdot e^{\frac{1}{\sigma^2}m_i\cdot y-\frac{1}{2\sigma^2}\|m_i\|} \end{aligned} \quad (1)$$

In the rest of the paper, $\frac{1}{Z}$ always accounts for the normalization of the Gaussian pdfs to 1. Under Bayes' law

$$\Pr_\theta(c_i|Y{=}y) = \frac{\Pr_\theta(Y{=}y|c_i)\Pr(c_i)}{\sum_j \Pr_\theta(Y{=}y|c_j)\Pr(c_j)} \quad (2)$$

If all the classes c_i share the same variance, the first terms in equation (1) are independent of c_i, and can be simplified. We obtain an *a posteriori* probability which has the form of the connectionist *SoftMax*

$$\Pr_\theta(c_i|Y{=}y) = \frac{e^{h_i(y)+\log\Pr(c_i)}}{\sum_j e^{h_j(y)+\log\Pr(c_j)}} \quad (3)$$

We note that the connectionist weighted sum is

$$h_i(y) = \frac{1}{\sigma^2}m_i^T\cdot y - \frac{1}{2\sigma^2}\|m_i\|^2$$

- The dot product $\frac{1}{\sigma^2}m_i^T\cdot y$ is a weighted sum of the inputs y, with $v_i = \frac{1}{\sigma^2}m_i$ as a weight vector.

[1] Y denotes the corresponding random variable. In this section, we will explicitely define the different random variables which carry the input information.

- The regularization term $\frac{1}{2\sigma^2}\|m_i\|^2 = \frac{\sigma^2}{2}\|v_i\|^2$ has some similarities with weight decay.
- The bias $\log \Pr(c_i)$ takes into account the prior probability of class c_i.

This probabilistic formalism provides us with additional insight about the architecture and training of current (and extended) MLPs.

2.2. Extending the MLP architecture

When y is computed by a MLP from the actual input : $y = \mathcal{T}(x)$, equation (3) adds a last layer to this MLP which computes $\mathcal{T}$. Given a MLP with SoftMax outputs, we have shown there exist quantities inside the connectionist architecture which can be interpreted as Gaussian pdfs over some hidden parameters y (the activations of the last hidden layer). This suggests two ways to extend the current MLP architectures:

- The regularization term $\frac{\sigma^2}{2}\|w_i\|^2$ adds a penalty to the classes whose corresponding weights have a large magnitude[2]. As a consequence, the training algorithm will try to find a solution where each class in encoded with parameters of comparable size, carrying a comparable amount of information. Later in the paper, we will demonstrate advantageous uses of this "balanced output" property. The absence of such a term means, in our formalism, that the variance σ is zero.
- It is possible to let covariance matrixes be non spherical, or different from one class to another. Terms which are quadratic in y appear in the computation of $h_i(y)$, as they do in second order Neural networks. We have not explored this direction in much details, since the Back-Prop training algorithm tended to instability, especially when it was used to train the covariance matrix.

In the next subsections, we describe two versions of our probabilistic interpretation of MLP training.

2.3. A MMI understanding of the MLP training

Our goal is to maximize, over the training set, the amount of information provided by the random variable X about the random event C whose outcome is the class to recognize:

$$I(C;X) = \sum_{x,c_i} \Pr(c_i, X{=}x) \log \frac{\Pr(c_i, X{=}x)}{\Pr(c_i)\Pr(X{=}x)}$$

This is known as the MMI criterion. In practice, as the entropy $H(C) = -\sum_{c_i} \Pr(c_i) Log \Pr(c_i)$ is fixed, one minimizes the conditional entropy of C given X

$$H(C|X) = H(C) - I(C;X)$$

Here we have as an estimate for the *a posteriori* probability the MLP output $\Pr_\theta(c_i|Y{=}\mathcal{T}(x))$, the empirical error we minimize is:[3]

$$H_\theta(C|X) = -\sum_{x,c_i} \Pr(c_i, X{=}x) . \log \underset{\theta}{\Pr}(c_i|Y{=}\mathcal{T}(x))$$

It can be shown[4] that $H_\theta(C|X) \geq H(C|X)$ and that the minimal value is reached when

$$\forall x, c_i, \underset{\theta}{\Pr}(c_i|Y{=}\mathcal{T}(x)) = \Pr(c_i|X{=}x)$$

This is yet another proof of the fact that, with a proper error criterion, the MLP outputs converge, as learning proceeds, towards the class posterior probabilities[7].

Usually, probabilistic interpretations of the MLP outputs are valid only *after* learning has properly converged. Our formalism extends their scope to the whole learning process. At the beginning of the learning phase, the MLP output $\Pr_\theta(c_i|Y{=}\mathcal{T}(x))$ is an estimate of the probability of class c_i , given a transform of the input $\mathcal{T}(x)$ which does not transmit all the discriminant information about c_i. The same probabilistic interpretation of the MLP outputs is valid during the whole training phase. At the convergence of the learning process, we expect to have the following properties[4]:

1. The $\mathcal{T}(x)$ density is more adapted to Gaussian modeling than the x density.
2. The parameters θ are set to Maximize the Mutual Information, to model the actual probability distribution of C given Y : $\Pr_\theta(c_i|Y{=}\mathcal{T}(x)) = \Pr(c_i|Y{=}\mathcal{T}(x))$.
3. $\mathcal{T}$ preserves all the discriminant information: $\Pr(c_i|Y{=}\mathcal{T}(x)) = \Pr(c_i|X{=}x)$.

2.4. Back to the standard sigmoid

To compute an estimate of $\Pr_\theta(Y{=}\mathcal{T}(x))$, one usually sums the estimates of all the classes, as it is usually done with MMI algorithms:

$$\underset{\theta}{\Pr}(Y{=}\mathcal{T}(x)) = \sum_j \Pr(c_j) \underset{\theta}{\Pr}(Y{=}\mathcal{T}(x)|c_j) \qquad (4)$$

However, we can also model each class separately. We have one distinct probabilistic space per class, with, for an input $\mathcal{T}(x)$, two possible outcomes: the class c_i and the non-class $\bar{c}_i$ which is the complementary event.

$$\begin{aligned}\Pr_{\theta,i}(Y{=}\mathcal{T}(x)) = \Pr(c_i)\Pr_{\theta,i}(Y{=}\mathcal{T}(x)|c_i) \\ + \Pr(\bar{c}_i)\Pr_{\theta,i}(Y{=}\mathcal{T}(x)|\bar{c}_i)\end{aligned} \qquad (5)$$

If we use the latter estimation of $\Pr_{\theta,i}(Y{=}\mathcal{T}(x))$, the *a posteriori* probability can be expressed with the sigmoid function $f(x) = \frac{1}{1+e^{-x}}$,

$$\underset{\theta,i}{\Pr}(c_i|Y{=}\mathcal{T}(x)) = f\left(\log \frac{\Pr(c_i)\Pr_\theta(Y{=}\mathcal{T}(x)|c_i)}{\Pr(\bar{c}_i)\Pr_\theta(Y{=}\mathcal{T}(x)|\bar{c}_i)}\right) \qquad (6)$$

Equation (6) shows the similarity between the class posterior probability and the ith sigmoidal output of a connectionist classifier. Maximization of mutual information is performed for each class separately.[5] This is the standard connectionist *Cross-Entropy*. If c_i is the correct class

$$CE = \log \underset{\theta,i}{\Pr}(c_i|Y{=}\mathcal{T}(x)) + \sum_{j\neq i} \log \underset{\theta,j}{\Pr}(\bar{c}_j|Y{=}\mathcal{T}(x)) \qquad (7)$$

$\Pr_{\theta,i}(Y{=}\mathcal{T}(x)|c_i)$ is computed as in equation (1). We can consider the "non-class" $\bar{c}_i$ as another class with its own mean n_i:

$$\underset{\theta,i}{\Pr}(Y{=}\mathcal{T}(x)|\bar{c}_i) = \frac{1}{Z} \cdot e^{-\frac{1}{2\sigma^2}\|\mathcal{T}(x)-n_i\|^2} \qquad (8)$$

[2] If class c_i has a large prior probability, the resulting large bias is taken into account in the $\log c_i$ term, and is not penalized.

[3] For one training sample, this objective function is the SoftMax version of the connectionist Cross-Entropy., i.e. the Kullback-Liebler divergence between the SoftMax outputs of the MLP $\Pr_\theta(c_i|Y = \mathcal{T}(x))$ and the target outputs $\Pr(c_i|X = x)$.

[4] However, we are not guaranteed that $\mathcal{T}$ outputs are optimized for Gaussian modeling or that the m_i correspond to some "true" estimates.

[5] In theory, the different MLP outputs are not directly comparable, as they represent estimations in different probabilistic spaces. However, as learning converges, they should approximate the actual *a posteriori* probability, as it has been shown that the minimization of the Cross-Entropy criterion leads to the Bayesian optimum[7] (the proof is similar to the one which is usually given for the Mean Square Error criterion).

Equation (6) amounts to the application of the sigmoid function to a connectionist weighted sum:

$$\Pr_{\theta,i}(c_i|Y=\mathcal{T}(x)) = f\left(\frac{1}{\sigma^2}(m_i-n_i)^T\cdot y-\frac{1}{2\sigma^2}\|m_i\|^2+\frac{1}{2\sigma^2}\|n_i\|^2+\log\frac{\Pr_\theta(c_i)}{\Pr_\theta(\overline{c}_i)}\right)$$

2.5. Advantages of this approach

In this section , we have described two versions of a MLP architecture which implements Gaussian pdfs (G-MLP) in its last hidden layer. The *SoftMax G-MLP* is built to satisfy the statistical MMI training criterion. The *Sigmoid G-MLP* departs more from classical statistical models, and bears a strong resemblance to a MLP with sigmoidal outputs trained to minimize the Cross Entropy error.

To stress the links between the "Gaussian Mixture " and the "MLP" approaches is interesting in two respects:

- In the development of a connectionist architecture, a statistical interpretation of the weighted sums and the activations allows one to analyse what happens inside the "black box" and to make reasonable architectural and parametric choices. Moreover, as maintaining good control over the learning curve is a key factor in achieving good performance, the validity of this probabilistic interpretation during learning (and not only after learning has converged) is an highly desirable property.
- The connectionist interpretation of statistical models allows the application of efficient versions of the gradient back-propagation algorithm. This exploits one of the most useful findings in connectionist research: gradient descent is applicable (in terms of convergence and computing time) to non linear system with a large number of parameters and estimated over large database.

Gaussian pdfs with equal variance are known to generate class boundaries which are hyperplanes. Therefore, in our formalism, one can say that the first layers of the MLP transform the input into hidden activations which are separable by hyperplanes (connectionist view) or into parameters which are suitable for Gaussian modeling (standard statistical view)

Those are topical concerns for speech recognition, as two major obstacles to improving current systems are:

- In the case of HMM based systems, gradient descent is generally considered as slow and hazardous.
- Connectionist speech recognition does not model time alignment properly.

The next sections show how we can efficiently use these links between Gaussian mixtures and MLPs to unify the treatment of some HMMs and some global connectionist approaches in speech recognition.

3. OUR GLOBAL CONNECTIONIST APPROACH

In this section, we take a MMI trained HMM and show that it is equivalent to a MLP with an alignment layer.[6]

3.1. Links between HMMs and $\alpha\beta$-TDNNs

We now extend the models developed in the previous section by replacing the simple Gaussian pdfs with Hidden Markov Models, and we add delayed connections to the MLP, so that it becomes a TDNN[8]. The input sequence is $x_1^T = [x_1,\ldots,x_T]$, but the sequence of vectors we want model with the HMM is an non-linear transform of this input through a TDNN: $y_1^T=[y_1,\ldots,y_T]$ with $y_t=\mathcal{T}\left(x_{t-d}^{t+d}\right)$.Given a word w_l, the emission probability of y_1^T sums the probabilities along every possible path q_1^T:

$$\Pr_\theta\left(y_1^T|w_l\right)=\sum_{q_1^T}\Pr\left(q_1^T|w_l\right).\Pr_\theta\left(y_1^T|q_1^T,w_l\right) \qquad (9)$$

Path q_1^T is a sequence of states $\left\{q^1,\ldots,q^t\ldots,q^T\right\}$. The emission probability given path q_1^T is

$$\Pr_\theta\left(y_1^T|q_1^T,w_l\right)=\frac{1}{Z}\prod_{t=1}^T e^{-\frac{1}{2\sigma^2}\|y_t-m_{q^t}\|^2} = \underbrace{\frac{1}{Z}\prod_{t=1}^T e^{-\frac{1}{2\sigma^2}\|y_t\|^2}}_{\text{constant}}\cdot\underbrace{\left[\prod_{t=1}^T e^{m_{q^t}\cdot y_t-\frac{1}{2}\|m_{q^t}\|^2}\right]^{\frac{1}{\sigma^2}}}_{\text{actual score}} \qquad (10)$$

By substituting eq.(10) into eq.(9), and simplifying by $\frac{1}{Z}\prod_{t=1}^T e^{-\frac{1}{2\sigma^2}\|y_t\|^2}$, we obtain a word score which is proportional to the likelihood[7]

$$\Pr_\theta\left(y_1^T|w_l\right)\propto\sum_{q_1^T}\Pr\left(q_1^T|w_l\right)\cdot\left[\prod_{t=1}^T e^{(m_{q^t})^T\cdot y_t-\frac{1}{2}\|m_{q^t}\|^2}\right]^{\frac{1}{\sigma^2}}$$

For an efficient computation of this word score, a connectionist implementation of the Forward/Backward procedure is possible[3], and was implemented in the $\alpha\beta$-TDNN, which could be efficiently trained with the Back-Prop training procedure on speaker independent word recognition problems[5]. To compute the *a posteriori* probability $\Pr_\theta\left(w_l|y_1^T\right)$, as in Section 2, we have two ways to estimate $\Pr_\theta\left(y_1^T\right)$.

The *SoftMax G-TDNN* computes the usual ratio used with HMMs optimized with the MMI criterion and estimates $\Pr_\theta\left(y_1^T\right)$ by summing the likelihoods of all the words in the vocabulary, as shown in equation (4). The MMI training criterion, described in section 2.3, is applied at the word level, as described in [5]. If we note $v_{q_i}=\frac{1}{\sigma^2}m_{q_i}$ the weight vector for state q_i, the *G*-TDNN corresponds to an implementation of the $\alpha\beta$-TDNN with, as the frame level score for state q_i,

$$e^{(v_{q_i})^T\cdot y_t-\frac{\sigma^2}{2}\|v_{q_i}\|^2} \text{ instead of } f\left((v_{q_i})^T\cdot y_t\right)$$

So far, we found in our experiments the *G*-TDNN error rate to be half of the $\alpha\beta$-TDNN error rate, which shows that our new HMM interpretation of connectionist time alignment is a source of good architectural choices:

- The exponential transform seems to more appropriate than the sigmoid one.
- The $\frac{\sigma^2}{2}\|v_{q_i}\|^2$ regularization term implements what we called, in section 2, the "balanced output" property among the states of a given word. This term prevents one state from growing larger weights than the others, taking a larger proportion of the word duration and more temporal credit assignment.

[6] Historically, we proceeded the other way round, starting from a connectionist architecture and finding, after CPU decades of computers simulations, an architecture which had an HMM framework and could handle large speech corpora.

[7] The peak around the path of maximum likelihood will be sharper if path exponent $\frac{1}{\sigma^2}$ is larger. σ^2 is therefore equivalent to an alignment temperature.

The *Sigmoid G-TDNN* requires the definition of *non word* models $\overline{w}_l$ to compute $\Pr_{\theta,l}\left(y_1^T\right)$. As in the static case (eq.(8)), our experiments used a baseline *non-word* model which has the same number of states as its corresponding word model, but with different means for the Gaussians. The word score $\Pr_{\theta,l}\left(w_l|y_1^T\right)$ can again be expressed in a sigmoidal form (eq.(6)). The error criterion is global, word-level Cross-Entropy (CE). Word scores between 0 and 1 are computed independently form one another: no comparison to the other word outputs is necessary to determine whether the word has been detected or not. This approach should be more suited to word spotting problems.

In terms of learning, the procedure we use is fundamentally different from the classical HMM Baum-Welsh estimation. Learning adjusts simultaneously the hidden parameters y_1^T (whose initial values are 0) and the Gaussian means m_{q_i} (which are initially set to small random values). The variance σ^2 is fixed, and chosen to minimize the error rate on a cross-validation corpus. HMMs with MLP transforms at their input have already been described[1], but MLP training mostly happened before or after HMM training, not simultaneously.

4. EXPERIMENTS

The speech material was collected from about 750 speakers, each of whom uttered the 10 digits (French) in isolation over the long distance telephone network. As input parameters, we use 6 Mel Frequency Cepstral Coefficients (MFCC) computed at a 16 msec frame rate, the Energy. We also constrain the TDNN architecture to compute the derivatives and the 2nd derivatives of the input parameters. This is the only place where we help the learning process with some *a priori* knowledge we have on this task (i.e. very useful discriminant information is given by the derivatives). Without this constraint, we found the system to yield about the same performance with "good" initial weights, but to be much more sensitive to the choice in the initial weights.

MMI or CE training of the network on 3500 digits takes about 10 hours on a IBM RISC workstation. Word models with 10 states were used. Results obtained with the MS-TDNN[6] and the $\alpha\beta$-TDNN[5] are given (Table 1) for comparison, on a task where the word boundaries are known. The extension to unknown word boundaries (and connected speech) is not yet satisfactory, as we have to deal with events, such as silence, which are not trainable with our discriminant procedure. Table 2 gives preliminary results with separately trained silence models and compare them to the CNET state-of-the-art HMM, with 1 and 16 Gaussian densities per state. It can be noted that the G-TDNN requires significantly fewer parameters than the HMM.

System	Error rate.	# states
MS-TDNN	1.1	5
$\alpha\beta$-TDNN	0.7	5
SoftMax G-TDNN	0.4	10
Sigmoid G-TDNN	0.3	10

Table 1: known word boundaries

System	Error rate.	# param.
Single Density HMM	0.6	18326
Multiple Density HMM	0.4	249656
MS-TDNN	1.6	
Sigmoid G-TDNN	0.9	4188

Table 2: unknown word boundaries

5. CONCLUSION

In this paper, we introduced a unified MLP/Gaussian interpretation of pattern classifiers. We extended this interpretation to incorporate time alignment and unify the *G-TDNN* global connectionist speech recognition system to an HMM. Unlike standard hybrid systems[2], the resulting probabilistic framework is applicable *inside* the connectionist architecture. This theoretical framework was implemented *as is* and, with very little tuning, yields high performance on a classical speaker independent isolated word recognition task. This work suggests a formal framework to solve one of the main challenge in connectionist speech recognition: how does one model time alignment?

Furthermore, this work adds some contributions to two very challenging problems in HMM speech recognition. First, it is possible to train from scratch, with a purely discriminant MMI based training procedure, an HMM based system on a large speech task. The resulting system has a number of parameters which is much smaller than a system trained in a non-discriminant way. Second, an HMM input representation can be automatically extracted for a given task.

6. ACKNOWLEDGMENTS

The author would like to thank Denis Jouvet and Chafic Mokbel for their suggestions about this work, and Michael Witbrock and David Sadek for proofreading this paper.

REFERENCES

[1] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Neural network - gaussian mixture hybrid for speech recognition or density estimation. In *Advances in Neural Information Processing Systems*, volume 4, pages 175–182, 1992.

[2] H.A. Bourlard and N. Morgan. *CONNECTIONIST SPEECH RECOGNITION: A Hybrid Approach.* Kluwer Academic Publisher, Boston, 1994.

[3] J.S. Bridle. Alphanets: a recurrent 'neural' network architecture with a hidden markov model interpretation. *Speech Communication*, 1990.

[4] P.F. Brown. *The Acoustic-Modelling Problem in Speech Recognition.* PhD thesis, Carnegie Mellon University, Pittsburgh, 1987.

[5] P. Haffner. Connectionist speech recognition with a global MMI algorithm. In *Eurospeech*, Berlin, Germany, September 1993.

[6] P. Haffner and A.H Waibel. Multi-state time-delay neural networks for continuous speech recognition. In *Advances in Neural Information Processing Systems*, volume 4, pages 579–588. Morgan Kaufmann, San Mateo, 1992.

[7] J.B. Hampshire and B.A. Pearlmutter. Equivalence proofs for multi-layer perceptron classifiers and the bayesian discriminant function. In *Proceedings of the 1990 Connectionist Model Summer School*, pages 159–172, San Mateo, USA, 1990. Morgan Kaufmann.

[8] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang. Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37:328–339, 1989.