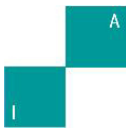
Inteligencia Artificial 0(0), 1-10



INTELIGENCIA ARTIFICIAL

http://erevista.aepia.org/

Minimum Classification Error Training of Hidden Markov Models for Sequential Data in the Wavelet Domain

Diego Tomassi, Diego Milone, Liliana Forzani

Laboratorio de Investigación en Señales e Inteligencia Computacional
FICH, Universidad Nacional del Litoral - CONICET, Argentina
diegotomassi@gmail.com

Laboratorio de Investigación en Señales e Inteligencia Computacional
FICH, Universidad Nacional del Litoral - CONICET, Argentina
dmilone@fich.unl.edu.ar

Instituto de Matemática Aplicada Litoral
FIQ, Universidad Nacional del Litoral - CONICET, Argentina
liliana.forzani@gmail.com

Abstract In the last years there has been increasing interest in developing discriminative training methods for hidden Markov models, with the aim to improve their performance in classification and pattern recognition tasks. Although several advances have been made in this area, they have been targeted almost exclusively to standard models whose conditional observations are given by a Gaussian mixture density. In parallel with this development, a special kind of hidden Markov models defined in the wavelet domain has found wide-spread use in the signal and image processing community. Nevertheless, these models have been typically restricted to fully-tied parameter training using a single sequence and maximum likelihood estimates. This paper takes a step forward in the development of sequential pattern recognizers based on wavelet-domain hidden Markov models by introducing a new discriminative training method. The learning strategy relies on the minimum classification error approach and provides reestimation formulas for fully non-tied models. Numerical experiments on a simple phoneme recognition task show important improvement over the recognition rate achieved by the same models trained under the maximum likelihood estimation approach.

Keywords: Dynamic pattern recognition, minimum classification error, discriminative training, hidden Markov models, wavelets.

1 Introduction

Hidden Markov models have been proven successful in dealing with sequential data, being at the core of state of the art methods for applications such as speech recognition [15] and sequence alignment in bioinformatics [3]. Within this modeling framework, maximum likelihood estimation has been the standard approach for learning parameters from data, taking advantage of the efficiency of the expectation-maximization algorithm (EM) [6]. The rationale behind this is that minimum Bayes risk can be attained by picking the class which maximizes the posterior probability given the observation sequence. This

ISSN: 1988-3064(on-line)

probability can be further replaced via Bayes' rule by the likelihood and an estimation of the class prior. Thus, within this framework the classifier design involves in fact a distribution approximation task.

The key observation to be noticed is that what is actually used in most cases is a plug-in maximum a posteriori approach: true class posterior probabilities are supposed to equal those for the models linked to each class. When this is true and the set of training signals is large enough, the above approach is in fact the best we can do. However, these assumptions usually do not hold for pattern classification tasks involving real-world data. When there is high variability in data or when training samples are limited, models posteriors cannot be expected to match the true class posteriors and Bayes risk becomes an unattainable lower bound.

To overcome these limitations, in recent years there has been a growing interest in discriminative training of hidden Markov models [8]. Unlike the previous distribution approach to parameter estimation, these methods aim to reduce the classification error by using training samples from all classes simultaneously and to maximize the dissimilarity between models of different classes. Several criteria have been proposed to drive the learning process, giving rise to methods such as Maximum Mutual Information [2] and Minimum Classification Error [9, 4].

The most widely used of those methods is Minimum Classification Error training (MCE). When applied to parameter estimation in hidden Markov models, this is a HMM-based discriminant analysis approach in which a soft approximation of the 0-1 loss is used to model the decision risk of the classifier. The learning problem becomes an optimization problem which directly links the design of the classifier to its expected performance and it is usually carried out by the generalized probabilistic descent (GPD) method [10].

MCE training has shown to outperform the conventional maximum likelihood approach in many applications. This success has also stimulated several efforts both to ground the method on a more principled basis [12, 1] and to improve its efficiency in real-world applications [7]. Nevertheless, most of these works deal only with standard hidden Markov models whose observation densities are given by Gaussian mixtures.

A very special kind of hidden Markov models comprises those defined in the wavelet domain. The best known of these models is the hidden Markov tree (HMT), which was introduced in [5] to account for statistical dependencies between coefficients in wavelet representations of signals and images. Although the HMT has found widespread use in applications, it is not well suited to sequential pattern recognition tasks because it cannot handle variable-lenght sequences. This is due to the use of the discrete wavelet transform, which makes the structure of the representation depend on the length of the signal. To relax this limitation, a composite HMM-HMT architecture was proposed in [13], in which an HMT models the observation density of each state of an external HMM. An EM algorithm for parameter estimation was derived in [13] for fully-coupled non-tied models and promising preliminary results both for signal denoising and classification were reported in [14] and [13], respectively.

In this paper we take a step forward in the development of sequential pattern classifiers in the wavelet domain by introducing a new discriminative training algorithm for the HMM-HMT model. It relies on the minimum classification error criterion and it is solved through the GPD approach. The proposed algorithm focusses in fully non-tied models in the wavelet domain. Use of them instead of Gaussian mixtures as observation densities requires the introduction of modifications to the standard MCE approach in order to avoid numerical issues. We provide reestimation formulas for all the parameters in the model and carry out simple phoneme recognition experiments to compare the performance of the proposed algorithm against the same model trained by the standard EM approach.

The paper is organized as follows: Section 2 reviews the composite HMM-HMT model and notation; reestimation formulas for the proposed algorithm are given in Section 3 and experimental results for phoneme recognition are shown in Section 4. Conclusions and future works are outlined in Section 5.

2 The HMM-HMT model

The HMM-HMT architecture is a composition of two Markovian models in which the HMT serves as observation density for each state of the HMM. Long-term dependencies are modeled by the external HMM, while the HMT models short-term dependencies in the wavelet domain. To make the following

sections clear, we summarize next the main definitions and notation for the HMM-HMT model. Further details can be found in [13].

2.1 Model definition and notation

In order to model a sequence $\mathbf{W} = \mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^T$, with $\mathbf{w}^t \in \mathbb{R}^N$, we define a continuous HMM with the structure $\vartheta = \langle \mathcal{Q}, \mathbf{A}, \boldsymbol{\pi}, \mathcal{B} \rangle$, where $\mathcal{Q}$ is the set of states, $\mathbf{A} = \{a_{ij}\}$ is the matrix of state transition probabilities so that a_{ij} is the probability of transition from state i to state j; $\boldsymbol{\pi}$ is the initial state probability vector; and $\mathcal{B} = \{b_k(\mathbf{w}^t)\}$, is the set of observation densities. We will suppose that $\mathcal{Q}$ takes values $q \in 1, 2, \ldots, N_Q$. In addition, let $\mathbf{w}^t = [w_1^t, w_2^t, \ldots, w_N^t]$, with $w_n^t \in \mathbb{R}$, be the vector of coefficients of the wavelet representation of a signal [1]. The HMT in the state k of the HMM can be defined with the structure $\theta^k = \langle \mathcal{U}^k, \mathcal{R}^k, \boldsymbol{\kappa}^k, \boldsymbol{\epsilon}^k, \mathcal{F}^k \rangle$, where $\mathcal{U}^k$ is the set of nodes in the tree; $\mathcal{R}^k$ is the set of states in all the nodes of the tree; $\boldsymbol{\kappa}^k$ are the probabilities for the initial states in the root node; $\boldsymbol{\epsilon}^k = \left[\epsilon^k_{u,mn}\right]$ is the array whose elements hold the conditional probability of node u being in state m given that the state in its parent node $\rho(u)$ is n; and $\mathcal{F}^k = \left\{f^k_{u,m}(w_u)\right\}$ is the set of observation densities for the wavelet coefficients, that is, $f^k_{u,m}(w_u^t)$ is the probability of observing the wavelet coefficient w_u^t with the state m (in the node u). In particular, we assume that wavelet coefficients are conditionally Gaussian given the state in the node of the tree; so $f^k_{u,m}(w_u^t) = \mathcal{N}(w_u^t; \mu^k_{u,m}, \sigma^k_{u,m})$, where $\mathcal{N}(\cdot)$ denotes the Gaussian density. In later developments we will also denote with $\mathcal{R}_u^k$ the set of states in the node u, which takes values $r_u \in 1, 2, \ldots, M$.

2.2 Likelihood of the observations

The likelihood of the first order HMM for conditionally independent observations is given by [15]:

$$\mathcal{L}_\Theta(\mathbf{W}) = \sum_{\forall \mathbf{q}} \prod_t a_{q^{t-1}q^t} b_{q^t}(\mathbf{w}^t) \ , \tag{1}$$

where the observation density for each HMM state is given by (see [5]):

$$b_{q^t}(\mathbf{w}^t) = \sum_{\forall \mathbf{r}} \prod_{\forall u} \epsilon^{q^t}_{u,r_u r_{\rho(u)}} f^{q^t}_{u,r_u}(w_u^t) \ , \tag{2}$$

with $\mathbf{r} = [r_1, r_2, \ldots, r_N]$ a combination of hidden states in the HMT nodes. Thus, the complete likelihood for the joint HMM-HMT model is:

$$\begin{aligned}
\mathcal{L}_\Theta(\mathbf{W}) &= \sum_{\forall \mathbf{q}} \prod_t a_{q^{t-1}q^t} \sum_{\forall \mathbf{r}} \prod_{\forall u} \epsilon^{q^t}_{u,r_u^t r^t_{\rho(u)}} f^{q^t}_{u,r_u^t}(w_u^t) && (3)\\
&= \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \prod_t a_{q^{t-1}q^t} \prod_{\forall u} \epsilon^{q^t}_{u,r_u^t r^t_{\rho(u)}} f^{q^t}_{u,r_u^t}(w_u^t) && (4)\\
&\triangleq \sum_{\forall \mathbf{q}} \sum_{\forall \mathbf{R}} \mathcal{L}_\Theta(\mathbf{W}, \mathbf{q}, \mathbf{R}) \ , && (5)
\end{aligned}$$

where $a_{01} = \pi_1 = 1$. The sign $\forall \mathbf{q}$ denotes that the sum is over all possible state sequences $\mathbf{q} = q^1, q^2, \ldots, q^T$ and $\forall \mathbf{R}$ accounts for all possible sequences of all possible combinations of hidden states $\mathbf{r}^1, \mathbf{r}^2, \ldots, \mathbf{r}^T$ in the nodes of each tree. See [13] for details about the HMM-HMT model and the EM algorithm for training it. We will refer to $\mathcal{L}_\Theta(\mathbf{W}, \mathbf{q}, \mathbf{R})$ as the joint likelihood of the observations and the states of the model.

3 Algorithm formulation

The MCE approach for classifier design involves a set of discriminant functions optimized in a competitive way in order to achieve the least classification error over the training sample. Discriminant functions are

[1] For a wavelet analysis up to J levels and skipping the coarser approximation coefficient, $N = 2^J - 1$.

those functions which measure the degree of membership of an observation to a given class, thus characterizing the decision rule of the classifier. Let $\{g_j(\mathbf{W};\Lambda)\}$ be a parameterized set of such discriminant functions for a classification task at hand, $\mathbf{W}$ be an observation, Λ be the whole parameter set, and $C(\mathbf{W})$ be the decision of the classifier. The classifier will decide that observation $\mathbf{W}$ belongs to class i when

$$C(\mathbf{W}) = \arg\max_j g_j(\mathbf{W};\Lambda) = i \ . \tag{6}$$

To train a set of HMMs within this framework, the discriminant function of each class is chosen to be a function of the joint likelihood for the HMM of that class. In order to put in context the proposed algorithm for HMM-HMT models, we first review the basics of the MCE approach.

3.1 General MCE approach

A main feature of the MCE training method is that model update is competitive with regard to classes. That is, all models are updated simultaneously and the strenght of the update depends on how confusing the decision is to the classifier. Within this framework, minimization of the classification error is pursued through a three-step process:

1. *Simulation of the classifier decision.* This is carried out defining a function $d_i(\mathbf{W};\Lambda) : \mathbb{R} \to \mathbb{R}$ which is usually chosen to take a negative value when the classifier decision is right and a positive one otherwise. Following the decision rule (6), for a training sequence that belongs to class i, this function can be written as

   $$d_i(\mathbf{W};\Lambda) = -g_i(\mathbf{W};\Lambda) + \max_{j\neq i}\{g_j(\mathbf{W};\Lambda)\} \ .$$

   However, the max operation is not differentiable and so what is used in practice is a soft approximation to it. Function $d_i(\mathbf{W};\Lambda)$ is often referred to as the missclassification function.

2. *Soft approximation of the 0-1 loss*: the simulated classifier decision is embedded in a soft differentiable function which approximates the noncontinuous 0-1 loss. A common choice for this approximation is the sigmoid function defined as:

   $$\ell(d_i(\mathbf{W};\Lambda)) = \ell_i(\mathbf{W};\Lambda) = \frac{1}{1+\exp\left(-\gamma d_i(\mathbf{W};\Lambda)+\beta\right)} \ . \tag{7}$$

   Parameter γ controls the sharpness of the sigmoid and the bias β is usually set to zero.

3. *Minimization of the empirical classification risk*: let $\mathcal{M}$ be the number of classes in the problem. Let Ω_i stand for the set of patterns which belong to class i. The classification risk conditioned on $\mathbf{W}$ can be written as

   $$\ell(\mathbf{W};\Lambda) = \sum_{i=1}^{\mathcal{M}} \ell_i(\mathbf{W};\Lambda)\ \mathcal{I}(\mathbf{W}\in\Omega_i) \ , \tag{8}$$

   where $\mathcal{I}(\cdot)$ is the indicator function. The expected risk then reads

   $$\mathcal{L}(\Lambda) = \mathcal{E}_{\mathbf{W}}\left[\ell(\mathbf{W};\Lambda)\right] \ . \tag{9}$$

The GPD approach for MCE training is an on-line scheme which aims at minimizing (9) by updating the whole set of parameters Λ in the steepest-descent direction of the loss. Starting from an initial estimate $\hat{\Lambda}_0$, the τ-th iteration of the algorithm can be summarized as:

$$\hat{\Lambda} \longleftarrow \hat{\Lambda} - \alpha_\tau \left.\frac{\partial \ell(\mathbf{W}_\tau;\Lambda)}{\partial\Lambda}\right|_{\Lambda=\hat{\Lambda}_\tau} \ . \tag{10}$$

The updating process is often carried out with each training signal. Under mild conditions, it is shown that $\hat{\Lambda}$ converges to Λ^* with probability one provided the learning rate $\alpha_\tau \to 0$ as $\tau \to \infty$ [10].

3.2 Proposed algorithm

We start by choosing the functional form for the discrimination functions $g_j(\mathbf{W};\Lambda)$. In order for the method to be useful for training the model, we must preserve some link between these functions and the HMM. A common choice is to define $g_j(\mathbf{W};\Lambda)$ as a function of the joint likelihood $\mathcal{L}_\Theta(\mathbf{W},\mathbf{q},\mathbf{R})$ [4]. In particular, we will initially consider the following functional form based on Viterbi decoding:

$$\begin{aligned} g(\mathbf{W}|\Lambda) &= \log\left(\max_{\mathbf{q},\mathbf{R}}\{\mathcal{L}_\Theta(\mathbf{W},\mathbf{q},\mathbf{R})\}\right) \\ &= \sum_t \log a_{\bar{q}^{t-1}\bar{q}^t} + \sum_t\sum_{\forall u}\log \epsilon^{\bar{q}^t}_{u,\bar{r}^t_u\bar{r}^t_{\rho(u)}} + \sum_t\sum_{\forall u}\log f^{\bar{q}^t}_{u,\bar{r}^t_u}(w^t_u)\ . \end{aligned}$$

In the expression above, $\bar{q}^t$ and $\bar{r}^t$ refer to states that achieve maximum joint likelihood. Next, we must define the missclassification function $d_i(\mathbf{W};\Lambda)$. For HMMs with Gaussian mixture observations and the discriminant functions defined as above, it is a standard practice to choose it as

$$d_i(\mathbf{W}) = -g_i(\mathbf{W};\Lambda) + \log\left[\frac{1}{\mathcal{M}-1}\sum_{j\neq i} e^{g_j(\mathbf{W};\Lambda)\eta}\right]^{1/\eta}\ . \tag{11}$$

As η becomes arbitrarily large the term in brackets approximates, up to a constant, the supremum of $\{g_j(\mathbf{W};\Lambda)\}$ for all j different than i. However, likelihoods for the HMT model are tipically much smaller than those found for Gaussian mixtures. As a result, $g_j(\mathbf{W};\Lambda)$ often takes extremely low values for $\mathbf{W}\notin\Omega_j$ and the exponentiation gives rise to numerical underflow. Therefore, we define the missclassification function to be:

$$d_i(\mathbf{W};\Lambda) = 1 - \frac{\left[\frac{1}{\mathcal{M}-1}\sum_{j\neq i} g_j(\mathbf{W};\Lambda)^{-\eta}\right]^{-1/\eta}}{g_i(\mathbf{W};\Lambda)}\ . \tag{12}$$

To avoid restricting η to be an even integer, we also redefine the discriminant functions to be positive-valued:

$$g_i(\mathbf{W};\Lambda) = -\log\left(\max_{\mathbf{q},\mathbf{R}}\{\mathcal{L}_\Theta(\mathbf{W},\mathbf{q},\mathbf{R})\}\right)\ . \tag{13}$$

For an approximation of the 0-1 loss, we follow the standard practice and choose a sigmoid function as defined in (7).

As GPD is a gradient-based optimization method, we must introduce some transformation of the parameters to allow for such an unconstrained optimization to be valid [9]. To constrain a_{ij} to be a probability measure, we define $\tilde{a}_{ij}$ so that $a_{ij} = \exp\tilde{a}_{ij}/\sum_m \exp\tilde{a}_{im}$. A similar transformation is needed for the analogous probabilities in the internal HMTs. So, we define $\tilde{\epsilon}^k_{u,mn}$ so that $\epsilon^k_{u,mn} = \exp\tilde{\epsilon}^k_{u,mn}/\sum_p \exp\tilde{\epsilon}^k_{u,pn}$. We also need to constrain the Gaussian variances to be positive-valued. Thus, we define $\tilde{\sigma}^k_{u,m}$ so that $\tilde{\sigma}^k_{u,m} = \log\sigma^k_{u,m}$. Finally, we scale the Gaussian means in the conditional densities for the wavelet coefficients in order to improve numerical computations [4]. Following previous works, we define the transformed means $\tilde{\mu}^k_{u,m}$ to be $\tilde{\mu}^k_{u,m} = \mu^k_{u,m}/\sigma^k_{u,m}$.

3.3 Estimation of Gaussian means

Let assume that the τ-th training sequence $\mathbf{W}_\tau$ belongs to Ω_i and denote by $\Lambda^{(j)}$ the subset of Λ corresponding to the model for class j. To simplify notation, allow ℓ_i, d_j and g_j stand for $\ell_i(\mathbf{W};\Lambda)$, $d_j(\mathbf{W};\Lambda)$ and $g_j(\mathbf{W};\Lambda)$, respectively. The updating process works upon the transformed parameters $\tilde{\mu}^{(j)k}_{u,m}$ and is given by

$$\tilde{\mu}^{(j)k}_{u,m} \longleftarrow \tilde{\mu}^{(j)k}_{u,m} - \alpha_\tau \left.\frac{\partial \ell_i(\mathbf{W}_\tau;\Lambda)}{\partial\tilde{\mu}^{(j)k}_{u,m}}\right|_{\Lambda=\hat{\Lambda}_\tau}\ . \tag{14}$$

Applying the chain rule of differentiation we get for $j = i$:

$$\begin{aligned}\tilde{\mu}_{u,m}^{(i)k} \longleftarrow & \tilde{\mu}_{u,m}^{(i)k} - \alpha_\tau \gamma \ell_i (1-\ell_i) \frac{d_i - 1}{g_i} \times \\ & \times \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m) \left[\frac{w_u^t - \hat{\mu}_{u,m}^{(i)k}}{\hat{\sigma}_{u,m}^{(i)k}} \right] .\end{aligned} \tag{15}$$

For $j \neq i$, the same procedure leads to:

$$\begin{aligned}\tilde{\mu}_{u,m}^{(j)k} \longleftarrow & \tilde{\mu}_{u,m}^{(j)k} - \alpha_\tau \gamma \ell_i (1-\ell_i)(1-d_i) \frac{g_j^{-\eta-1}}{\sum_{k \neq i} g_k^{-\eta}} \times \\ & \times \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m) \left[\frac{w_u^t - \hat{\mu}_{u,m}^{(j)k}}{\hat{\sigma}_{u,m}^{(j)k}} \right] .\end{aligned} \tag{16}$$

3.4 Estimation of Gaussian variances

The updating process for Gaussian variances is completely analogous to the one shown above for means. Assuming again that the τ-th training sequence $\mathbf{W}_\tau$ belongs to Ω_i, the updating process for $j = i$ reads:

$$\begin{aligned}\tilde{\sigma}_{u,m}^{(i)k} \longleftarrow & \tilde{\sigma}_{u,m}^{(i)k} - \alpha_\tau \gamma \ell_i (1-\ell_i) \frac{d_i - 1}{g_i} \times \\ & \times \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m) \left[\left(\frac{w_u^t - \hat{\mu}_{u,m}^{(i)k}}{\hat{\sigma}_{u,m}^{(i)k}} \right)^2 - 1 \right] .\end{aligned} \tag{17}$$

For $j \neq i$, we get:

$$\begin{aligned}\tilde{\sigma}_{u,m}^{(j)k} \longleftarrow & \tilde{\sigma}_{u,m}^{(j)k} - \alpha_\tau \gamma \ell_i (1-\ell_i)(1-d_i) \frac{g_j^{-\eta-1}}{\sum_{k \neq i} g_k^{-\eta}} \times \\ & \times \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m) \left[\left(\frac{w_u^t - \hat{\mu}_{u,m}^{(j)k}}{\hat{\sigma}_{u,m}^{(j)k}} \right)^2 - 1 \right] .\end{aligned} \tag{18}$$

3.5 Estimation of state-transition probabilities in the HMT

Working as above, it can be shown that the updating formulas for the transformed parameters $\tilde{\epsilon}_{u,mn}^{(j)k}$ reads for $j = i$:

$$\begin{aligned}\tilde{\epsilon}_{u,mn}^{(i)k} \longleftarrow & \tilde{\epsilon}_{u,mn}^{(i)k} - \alpha_\tau \gamma \ell_i (1-\ell_i) \frac{d_i - 1}{g_i} \times \\ & \times \left\{ \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m, \bar{r}_{\rho(u)}^t - n) - \right. \\ & \left. - \sum_t \sum_p \delta(\bar{q}^t - k, \bar{r}_u^t - p, \bar{r}_{\rho(u)}^t - n) \hat{\epsilon}_{u,mn}^{(i)k} \right\} ,\end{aligned} \tag{19}$$

and for $j \neq i$:

$$\begin{aligned}\tilde{\epsilon}_{u,mn}^{(j)k} \longleftarrow & \tilde{\epsilon}_{u,mn}^{(j)k} - \alpha_\tau \gamma \ell_i (1-\ell_i)(1-d_i) \frac{g_j^{-\eta-1}}{\sum_{k \neq i} g_k^{-\eta}} \times \\ & \times \left\{ \sum_t \delta(\bar{q}^t - k, \bar{r}_u^t - m, \bar{r}_{\rho(u)}^t - n) - \right. \\ & \left. - \sum_t \sum_p \delta(\bar{q}^t - k, \bar{r}_u^t - p, \bar{r}_{\rho(u)}^t - n) \hat{\epsilon}_{u,mn}^{(j)k} \right\} .\end{aligned} \tag{20}$$

3.6 Estimation of state transition probabilities in the HMM

Similarly to section (3.5), updating formulas for the transformed state transition probabilities $\tilde{a}_{sj}^{(j)}$ using an i-class sequence reads:

$$\tilde{a}_{sj}^{(i)} \longleftarrow \tilde{a}_{sj}^{(i)} - \alpha_\tau \gamma \ell_i (1-\ell_i) \frac{d_i - 1}{g_i} \times \\ \times \left\{ \sum_{t=1}^{T} \delta(\bar{q}_{t-1} - s, \bar{q}_t - j) - \sum_{t=1}^{T} \delta(\bar{q}_{t-1} - s) \hat{a}_{sj}^{(i)} \right\} . \tag{21}$$

and for $j \neq i$:

$$\tilde{a}_{sj}^{(j)} \longleftarrow \tilde{a}_{sj}^{(j)} - \alpha_\tau \gamma \ell_i (1-\ell_i)(1-d_i) \frac{g_j^{-\eta-1}}{\sum_{k \neq i} g_k^{-\eta}} \times \\ \times \left\{ \sum_{t=1}^{T} \delta(\bar{q}_{t-1} - s, \bar{q}_t - j) - \sum_{t=1}^{T} \delta(\bar{q}_{t-1} - s) \hat{a}_{sj}^{(j)} \right\} . \tag{22}$$

4 Experimental results

In order to assess the proposed training method, we carry out a simple automatic speech recognition test using phonemes from the TIMIT database [16]. In particular, we use phonemes `'eh'`, `'ih'` and `'jh'` and compare recognition rates achieved by the proposed method against those for the same models trained only by the EM algorithm. In all the experiments we use left-to-right hidden Markov models with $N_Q = 3$. The observation density for each state is given by an HMT with two states per node. The sequence analysis is performed on a short-term basis using Hamming windows of 256-samples length, with 50% overlap between consecutive frames. On each frame, a full dyadic discrete wavelet decomposition is carried out using Daubechies wavelets with four vanishing moments [11].

In a first set of experiments, we show numerically that the recognition rate achieved with the EM algorithm attains an upper bound which cannot be surpassed neither increasing the number of reestimations of the algorithm neither enlarging the training set. We next test the improvement in recognition rate after adding a discriminative stage to the training process.

4.1 How much improvement can the EM algorithm achieve?

Discriminative training methods usually use maximum-likelihood estimates provided by the EM algorithm as initial values for the competitive process. Thus, it is fair to ask if better performance could be achieved just using more training sequences or increasing the number of reestimations in the EM algorithm only. To answer this question we first perform a two-phoneme recognition task using models trained with the EM algorithm only and training sets of increasing sizes. The number of reestimations was fixed to 5. Obtained results for the {`'eh'`,`'ih'`} pair are given in Fig. 1.a). Shown results are averages over ten trials for each size of the training set and error bars indicate standard deviations. Results suggest that performance is in fact improved when we enlarge very small training sets. However, adding sequences to the training set beyond 50 samples does not translate into models achieving higher recognition rates.

The effect of fixing the size of the training set and increasing the number of reestimations used in the EM algorithm is shown in Fig. 1.b). Given values correspond to a training sample comprising 50 sequences. It can be seen that recognition rates remain fairly the same with the increase in the number of reestimations. All of these results confirm that for models trained only with the EM algorithm, performance is upper bounded and no significant improvement can be expected just increasing the number of reestimations or adding sequences to the training set.

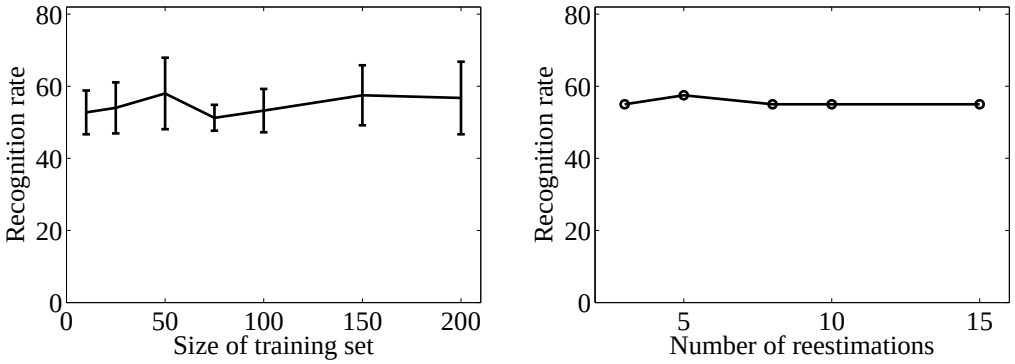


Figure 1: Recognition rates for EM training only: a) varying the size of the training set; b) increasing the number of reestimations.

4.2 MCE training for phoneme recognition

In order to get some insight into the learning process, we first consider a classification task comprising only two phonemes. It is straightforward to see that the proposed missclassification function reduces to

$$d_1(\mathbf{W};\Lambda) = 1 - \frac{g_2(\mathbf{W};\Lambda)}{g_1(\mathbf{W};\Lambda)} \ .$$

When the classifier decision is right, the second term in the rightside of the above expression is bigger than one and the missclassification function takes a negative value. As this decision is stronger, $d_1(\mathbf{W};\Lambda)$ becomes more negative and the resulting loss (7) goes to zero. We then see from the updating formulas in Sects. 3.3-3.6 that no updating is performed in such a case. So, the algorithm preserves model parameters that do well when classifying the current training signal. On the other hand, if the current training sequence is strongly missclassified, $d_1(\mathbf{W};\Lambda)$ will tend to 1. In this case, whether the algorithm update the parameters or not will depend on the value of γ in (7). As γ becomes larger, the loss approximation will go to one faster and even though the classifier is taking a wrong decision no parameter update is carried out. Thus, parameter update takes place only when models are confusable and it is the strongest when the current training sequence is equally likely for both of them.

Numerical experiments were carried out for each pair of the considered phonemes. Fifty sequences from each class were used for training and another set of twenty sequences from each class were used for testing. Five reestimation steps were used in the EM algorithm, along with Viterbi flat start. Parameters for the MCE learning stage were set to $\gamma = 1$, $\beta = 0$, and $\eta = 4$. The learning rate α_τ was linearly decreased during training, starting from $\alpha_0 = 2.5$. Five trials were performed, varying the number of competitive iterations through the whole training set. The first three rows in Table 1 show the recognition rates achieved for each pair of phonemes. Consistent performance improvements are obtained for the three pairs of phonemes. For pairs {`'eh'`,`'jh'`} and {`'ih'`,`'jh'`} the recognition rate increases monotonically to an upper bound as the number of iterations of the algorithm increases. Recognition rate for pair {`'eh'`,`'ih'`} shows some oscillations as the number of iterations increases. Nevertheless, it is clearly seen that discriminatively training the models significantly improves the recognition rate of the classifier. We next repeat the above experiment to consider the three phonemes jointly. Obtained results are shown in the last row in Table 1. Despite the recognition rate oscillates for increasing number of iterations, improvements remain bigger than 10% up to 35 iterations. Further MCE iterations seem to decrease performance. It should be noticed that adding phoneme `'jh'` to the classification task results in higher recognition rates over the {`'eh'`,`'ih'`} pair alone. This is because the former is an unvoiced phoneme and it is easier to discriminate from the pair of voiced phonemes.

Table 1: Recognition rates vs. MCE iterations over the whole training set.

Phoneme Set	EM Baseline	MCE Iterations 5	15	25	35	50
{'eh','ih'}	57.5	72.5	67.5	70.0	72.5	70.0
{'ih','jh'}	90.0	92.5	95.0	97.5	97.5	97.5
{'eh','jh'}	90.0	92.5	95.0	97.5	97.5	97.5
{'eh','ih','jh'}	65.0	75.0	73.3	75.0	73.3	68.3

5 Conclusions

This paper introduces a new method for discriminative training of hidden Markov models whose observations are sequences in the wavelet domain. The algorithm is based on the MCE/GPD approach and it allows for training of fully non-tied HMM-HMT models. Simple speech recognition experiments show that the proposed method achieves important improvements on recognition rates over training with the standard EM algorithm only. More extensive numerical experiments should be carried out in order to test the model with other speech material as well as with other patterns. In addition, further work should be targeted to optimally set the parameters for GPD optimization.

Acknowledgements

This work was carried out with financial support from UNL (CAI+D-012-72), ANPCyT (PAE-PICT-2007-00052) and CONICET.

References

[1] M. Afify, X. Li, and H. Jiang. Statistical analysis of minimum classification error learning for gaussian and hidden markov model classifiers. *IEEE Transactions on Audio, Speech, and Language Processing*, 15:2405–2417, 2007. doi: 10.1109/TASL.2007.903304.

[2] L.R. Bahl, P.F. Brown, P.V. De Souza, and R.L. Mercer. Maximum mutual information estimation of hmm parameters for speech recognition. In *Proc. of the Int. Conf. on Audio, Speech, and Signal processing (ICASSP86)*, pages 49–52, 1986.

[3] P. Baldi and S. Brunak. *Bioinformatics: The Machine Learning Approach.* MIT Press, Cambridge, Massachusetts, 2001.

[4] W. Chou. Minimum classification error rate (mce) approach in pattern recognition. In W. Chou and B.H. Juang, editors, *Pattern Recognition in Speech and Language Processing*, pages 1–49. CRC Press, 2003.

[5] M. Crouse, R. Nowak, and R. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Trans. on Signal Proc.*, 46:886–902, 1998. doi: 10.1109/78.668544.

[6] A.P. Dempster, N.M. Laird, and D.B. Durbin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[7] X. He and L. Deng. A new look at discriminative training for hidden markov models. *Pattern Recognition Letters*, 28:1285–1294, 2007. doi: 10.1016/j.patrec.2006.11.022.

[8] X. He, L. Deng, and W. Chou. Discriminative learning in sequential pattern recognition. *IEEE Signal Processing Magazine*, 25:14–36, 2008. doi: 10.1109/MSP.2008.926652.

[9] B.-H. Juang, W. Chou, and C.-H. Lee. Minimum classification error rate methods for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 5:257–265, 1997. doi: 10.1109/89.568732.

[10] S. Katagiri, B.-H. Juang, and C.H. Lee. Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method. *Proceedings of the IEEE*, 86:2345–2373, 1998. doi: 10.1109/5.726793.

[11] S. Mallat. *A Wavelet Tour of Signal Processing. Second Edition.* Academic Press, 1999.

[12] E. McDermott and S. Katagiri. A derivation of minimum classification error from the theoretical classification risk using parzen estimation. *Computers, Speech and Language*, 18:102–122, 2004. doi: 10.1016/S0885-2308(03)00037-8.

[13] D.H. Milone and L.E. Di Persia. An em algorithm to learn sequences in the wavelet domain. *Lecture Notes in Computer Science*, 4827:518–528, 2007. doi: 10.1007/978-3-540-76631-5-49.

[14] D.H. Milone, L.E. Di Persia, and D.R. Tomassi. Signal denoising with hidden markov models using hidden markov trees as observation densities. In *Proc. of the IEEE MLSP08 Workshop*, pages 374–379, 2008. doi: 10.1109/MLSP.2008.4685508.

[15] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition.* Prentice-Hall, New Jersey, 1993.

[16] V. Zue, S. Sneff, and J. Glass. Speech database development: Timit and beyond. *Speech Communication*, 9:351–356, 1990.