LINEAR AND NONLINEAR PREDICTION FOR SPEECH RECOGNITION WITH HIDDEN MARKOV MODELS

Marco SAEVENS
IRIDIA Laboratory
Université Libre de Bruxelles, cp. 194/6
Av. F. Roosevelt 50, 1050 Bruxelles – BELGIUM
saevens@ulb.ac.be

Hervé BOURLARD
LENOURT & HAUSPIE SPEECH PRODUCTS
Rozendaalstraat 14
8900 Ieper – BELGIUM

Abstract. When using hidden Markov models for speech recognition, it is usually assumed that the probability that a particular acoustic vector is emitted at a given time only depends on the current state and the current acoustic vector observed. This model does not take account of the dynamic nature of the speech signal. In order to introduce time correlation between successive acoustic vectors, some authors have proposed to consider the time series of observations on a state to be generated by a nonlinear deterministic process corrupted by a Gaussian additive noise. This results in the introduction of the prediction error in the likelihood function. In this paper, we review the basic ideas underlying these models. Thereafter, we briefly introduce an extension of the linear case, i.e., we permit the autoregressive coefficients to be corrupted by noise. Indeed, when working at the speech samples level, this is a simple way to take the intra and inter speaker variability into account; that is, to allow variability in the transfer function. In fact, this is what we are doing when extracting LPC coefficients and clustering them with Gaussian distributions. The advantage here is that we directly introduce the variability at the sample level. This leads to processes that are known as AutoRegressive Conditional Heteroscedastic (ARCH) processes, with nonconstant variances conditional on the past.

Keywords. Hidden Markov models, autoregressive models.

1. INTRODUCTION

Hidden Markov models are widely used for speech recognition (Jelinek, 1976; Levinson, Rabiner & Sonish, 1983; Rabiner, 1989; Huang, Ariki & Jack, 1990). However, strong assumptions have to be made to render the model computationally tractable (see, for example, Boulard, 1992). One of these assumptions is the observation independence of the acoustic vectors. Indeed, it is usually assumed that the probability that a particular acoustic vector is emitted at a given time only depends on the current state and the current acoustic vector observed.

To overcome this problem, many authors have tried to take account of the dynamics of the speech. For instance, Furuì (1986, 1991) and Gurguen, Sagayama & Furuì (1990) introduce features including the time-derivative of the acoustic vectors. Deng (1992a,b) modelizes the temporal evolution of the acoustic feature inside a state by a given function of time, i.e. a polynomial trend function of time $t$ spend in the state. Saerens considers that the acoustic vectors are generated by a continuous-time Markov model (1993a) or a stochastic differential equation (1993b) in each state. Wollekens (1987) assumes explicit dependence between the current vector and the last observed vector. He shows that, in the case of a correlated Gaussian probability distribution function, the emission probabilities depend on the prediction error of a first order linear predictor. On the other hand, Portitz (1982), Juan (1984) and Juan & Rabiner (1985) (see also Kenny, Lennig & Mermelstein, 1990; Tishby, 1991; Woodland, 1992) use Gaussian autoregressive densities per state to model the speech dynamics. Once more the emission probabilities depend on the prediction error of a linear predictor.

More recently, some authors have considered the possibility of using nonlinear prediction models (mostly multi-layer neural networks) for speech recognition with hidden Markov models (Tsujii, Takada & Wakita, 1990; Levin, 1990, 1991; Tebelskis & Waibel, 1990; Tebelskis, Waibel, Petek & Schmidbauer, 1991; Petek, Waibel & Tebelskis, 1992; Issa & Watanabe, 1990, 1991; Deng, Hassanien & Elmasry, 1991). In this case, the acoustic vectors are assumed to be generated at each frame by a discrete nonlinear process, different for every state, corrupted by an additive uncorrelated Gaussian noise.

First, we will review the basic ideas underlying these models. Thereafter, we briefly introduce an extension of the linear case, i.e. we permit the autoregressive coefficients to be corrupted by Gaussian noise. In fact, this is exactly what we are doing when extracting LPC coefficients and clustering them with Gaussian distributions. This leads to processes that are known as AutoRegressive Conditional Heteroscedastic (ARCH) processes, with nonconstant variances conditional on the past. Unfortunately, the reestimation formulae for the variances of the coefficients are nonlinear, and have to be solved numerically. However, these models have been widely studied and methods for computing an estimate are available.

2. ACOUSTIC VECTORS GENERATED BY A PROCESS CORRUPTED BY ADDITIVE NOISE

Let us assume that, in each state $s \in \{0, 1, ..., q\}$, the acoustic vector $x(k) = [x_0(k), x_1(k), ..., x_p(k)]^T$ at time $k$ is generated by a linear or nonlinear process corrupted by additive noise:

$$x(k) = F_z(x_{k-p}; \theta_z) + \varepsilon(k)$$  \hspace{1cm} (1)

where $x_{k-p}^T$ represents the sequence of $p$ acoustic vectors $(x(k-1), x(k-2), ..., x(k-p))$ and $\varepsilon(k)$ is assumed to be an independent and identically distributed random sequence with probability density function $p_{\varepsilon}(\varepsilon \mid \lambda_\varepsilon)$, parameters $\lambda_\varepsilon$, and zero mean. $F_z(x_{k-p}; \theta_z)$ is a deterministic function of the $p$ last acoustic vectors only, and parameters $\theta_z$.

Let us now denote for each time $k$ the state occupancy by $s_k \in \{0, 1, ..., q\}$. From equation (1), the conditional probability density of the observation $p_{x_k \mid x(k) \mid x_{k-p}; \theta_{\lambda_z}}$ on state $s_k$ at time $k$, given the last $p$ acoustic vectors, is as follows:

EUROSPEECH 1993
pp 507-510 (BERLIN)
\[ p_{X|XX(k) | X_{k-1}^{k-1}, \Theta_{k-1}, \Theta_{k}} = p_{d}(x(k) - F_{s_2}(X_{k-1}^{k-1}, \Theta_{k}) \mid \lambda_{s_2}) \] (2)

where \( \Theta_{k} \) is the set of parameters appearing in the distribution: \( \{ \theta_k, \lambda_{s_2} \} \). This is simply because, from equation (1), the conditional distribution of the stochastic variable \( x \), given the past, is the same as the distribution of \( e^2 \), but with a shift \( F_s \) in the mean value.

We will formulate the algorithm in the framework of word recognition so that, for each word, we observe a sequence of acoustic vectors \( x(0), x(1), \ldots, x(N) \).

Once the segmentation is known, one can compute the likelihood function for the sequence of acoustic vectors and the uttered word. The sequence of states defining a word is \( s_0, s_1, \ldots, s_N \), or, if we take the same notations as for the acoustic vectors, \( S_0^N \). The total likelihood function for the observations is defined as:

\[
\mathcal{L}(x(0), x(1), \ldots, x(N); s_0, s_1, \ldots, s_N) = p(X_0^N | S_0^N) \quad p(S_0^N) \]

\[
= p(x(0) | X_0^{N-1}; S_0^N) \quad p(x(N) | X_0^{N-1}; S_0^N) \quad \ldots \quad p(x(1) | X_0^0; S_0^N) \quad p(x(0) | S_0^N) \quad p(S_0^N) \]

(3)

However, since the observations are only available from the sample \( x(0) \), the first \( p \) conditional probabilities \( p(x(k) | X_0^k; S_0^N) \) \((k = 0, \ldots, p-1)\) cannot be evaluated from expression (2) because of the problem of the initial values. We therefore consider the conditional likelihood that is defined as:

\[
\mathcal{L}(x(p), x(p+1), \ldots, x(N); s_p, s_{p+1}, \ldots, s_N | X_0^{N-1}; S_0^N) \quad \exp[- \frac{1}{2} \text{trace}(\Sigma_x)]
\]

\[
= p(x(N) | X_0^{N-1}; S_0^N) \quad p(x(N-1) | X_0^{N-2}; S_0^N) \quad \ldots \quad p(x(p) | X_0^0; S_0^N) \quad p(S_0^N) \quad p(S_0^N) \quad \frac{1}{2} \text{trace}(\Sigma_x) \quad p(x(k) - F_{s_2}(X_{k-1}^{k-1}, \Theta_{k}) \mid \lambda_{s_2}) \]

(4)

Instead of (3). To simplify the notations, we will omit the conditional term, keeping in mind that we are disregarding the \( p \) first observations (taken as initial value). If there is a large number of data, this has negligible effect (for the computation of the exact total likelihood in the linear autoregressive case, see Box & Jenkins, 1976; in the context of speech recognition, see Juang, 1984). Moreover, we will take \( p \) as the new origin for the indexation of the time, while \( N \) remains the last observed vector.

Now, since the process (1) is of order \( p \), the present value only depends on the \( p \) last values. Assuming further that the emission of the acoustic vector through the process defined by (1) only depends on the present state, we obtain

\[
\mathcal{L}(x(0), x(1), \ldots, x(N); s_0, s_1, \ldots, s_N) \quad \exp[- \frac{1}{2} \text{trace}(\Sigma_x)]
\]

\[
= p(x(N) | X_0^{N-1}; s_N) \quad p(x(N-1) | X_0^{N-2}; s_{N-1}) \quad \ldots \quad p(x(p) | X_0^0; s_p) \quad p(S_0^N) \quad p(S_0^N) \quad \frac{1}{2} \text{trace}(\Sigma_x) \quad p(x(k) - F_{s_2}(X_{k-1}^{k-1}, \Theta_{k}) \mid \lambda_{s_2}) \]

(5)

Furthermore, since the sequence of states is supposed to be generated by a first order Markov process, we can rewrite (5) as

\[
\mathcal{L} = \prod_{k=0}^{N} p_{s_2}(x(k) | X_0^{k-1}, \Theta_{k}) \quad \prod_{k=1}^{N} p(s_k | s_{k-1}) \quad \pi_0 (6)
\]

where the \( \pi(s_k | s_{k-1}) \) are the transition probabilities. Finally, from (2), we can rewrite the conditional likelihood for the whole utterance as

\[
\mathcal{L}(x(0), x(1), \ldots, x(N); s_0, s_1, \ldots, s_N) = \prod_{k=0}^{N} p_{s_2}(x(k) - F_{s_2}(X_0^{k-1}, \Theta_{k}) \mid \lambda_{s_2}) \quad \prod_{k=1}^{N} p(s_k | s_{k-1}) \quad \pi_0 (7)
\]

There is another, more straightforward, way to obtain the same result. Indeed, the likelihood of the parameters, given the values of \( (e^2(0), \ldots, e^2(N)) \), and conditional on the sequence of states is:

\[
\mathcal{L}(e^2(0), e^2(1), \ldots, e^2(N); s_0, s_1, \ldots, s_N) = \prod_{k=0}^{N} p_{s_2}(x(k) \mid \lambda_{s_2})
\]

and since the Jacobian of the transformation from the \( (x(k)) \) to the \( (e^2(k)) \) is unity, this joint probability function also represents the likelihood function of the parameters, given \( (x(0), x(1), \ldots, x(N)) \). Equation (7) follows directly.

From now, we have to make some assumptions about the distribution of the noise.

3. TRAINING AND RECOGNITION IN THE CASE OF GAUSSIAN ADDITIVE NOISE

Of course, the most studied case is the one for which the noise is Gaussian distributed with zero mean; that is, for each state \( s \),

\[
p_{s_2}(x(k) - F_{s_2}(X_0^{k-1}, \Theta_{s_2}) \mid \lambda_{s_2}) = \frac{1}{\sqrt{2\pi \sigma_x^2}} \exp[- \frac{1}{2} \sum_{k=0}^{N} (x(k) - F_{s_2}(X_0^{k-1}, \Theta_{s_2}))^2]
\]

During Viterbi training, one has to minimize the negative of the log likelihood within a constant term:

\[
\mathcal{L} = -\log(\mathcal{L}(x(0), x(1), \ldots, x(N); s_0, s_1, \ldots, s_N)) = \frac{1}{2} \sum_{k=0}^{N} [x(k) - F_{s_2}(X_0^{k-1}, \Theta_{s_2})]^2 \quad \frac{1}{2} \sum_{k=0}^{N} \log|\Sigma_{s_2}| - \sum_{k=1}^{N} \log p(s_k | s_{k-1})
\]

By introducing the additional constraint that the sum of the transition probabilities coming from the same state sum to one, one can easily derive restimation formulae for the variance-covariance matrix and the transition probabilities. Most of the authors assume, however, the variance-covariance matrix to be the identity matrix.

What concerns the predictor parameters \( \Theta_{s} \), the training method will, of course, be different for linear and nonlinear predictors. In the linear case, both standard techniques for the estimation of the parameters of autoregressive processes (for Viterbi algorithm), and forward-backward algorithm (Poritz, 1982; Juang, 1984; Juang & Rabiner, 1983; Kenny, Lennig & Mermeilestein, 1990; Tishby, 1991, and Woodland, 1992) can be used. In the nonlinear case, gradient descent is used, both for Viterbi algorithm (Levin, 1990, 1991; Tewksbury & Waibel, 1990; Tewksbury, Waibel, Petok & Schmidbauer, 1991; Petok, Waibel & Tewksbury, 1992; Iso & Watanabe, 1990, 1991) and forward-backward algorithm (Tsuboka, Takada & Wakisaka, 1990).
During recognition, the best state segmentation $S_N^N$, given the observation, is the one that maximizes

$$p(S_N^N \mid X_N^N) = \frac{p(X_N^N \mid S_N^N) p(S_N^N)}{p(X_N^N)}$$

(10)

And maximizing this last expression is equivalent to maximize (9). The segmentation is found by dynamic programming. Of course, to save computation time, dynamic programming must not necessarily be performed every sample. We can assume that the state is allowed to change only after a given duration $\tau$. In this case, the emission probability must be cumulated during duration $\tau$, after which transition is allowed.

4. MODELLING SPEECH AT THE SAMPLE LEVEL WITH AN ARCH MODEL

As mentioned before, the modelling can be done at the sample level or at the feature vector level. However, we have to keep in mind that, very roughly, there are two kinds of variability in the speech signal: (1) real noise that affects the speech signal itself and (2) inter and intra speakers variability that results in different vocal tract shapes, and therefore different transfer functions, related to the same phonetic unit. In this paragraph, we introduce a model that takes account of the two sources of variability.

The main idea is to assume that, in the linear case, the autoregressive coefficients, related to the transfer function, are also random variables, subject to fluctuations. Indeed, let us consider that the speech samples $x(k)$ are generated thanks to an autoregressive process corrupted by additive noise:

$$x(k) = \sum_{i=1}^{p} a_i^o x(k-i) + \varepsilon^o(k)$$

(11)

where the $\varepsilon^o(k)$ are assumed to be independent Gaussian distributed random variables with zero mean and variance $\sigma^o_\varepsilon$. We also assume that the autoregressive coefficients $a_i^o$ are corrupted by noise:

$$a_i^o(k) = a_i^o + \varepsilon_i^o(k)$$

(12)

where the $\varepsilon_i^o(k)$ are also independent Gaussian random variables with zero mean and variance $\sigma^\varepsilon_i^o$. In fact, this is almost what we are doing when extracting LPC coefficients and clustering them with Gaussian distributions. The difference here is that we directly introduce the variability at the sample level.

Equation (11) can be rewritten as:

$$x(k) = \sum_{i=1}^{p} a_i^o x(k-i) + \sum_{i=1}^{p} x(k-i) \varepsilon_i^o(k) + \varepsilon^o(k)$$

(13)

Now, since a sum of independent Gaussian random variables is also Gaussian distributed, one has:

$$p_s(x \mid X_{k-p}^{k-1}, \Theta_q^o) = \frac{1}{\sqrt{2\pi h^o(k)}} \exp \left[ -\frac{(x(k) - A^o \cdot X_{k-p}^{k-1})^2}{2 h^o(k)} \right]$$

(14a)

with

$$h^o(k) = \sigma^o_\varepsilon + \sum_{i=1}^{p} \sigma^o_i x_i^2(k-i)$$

(14b)

where $A^o$ is the vector of the autoregressive coefficients; and $(A^o \cdot X_{k-p}^{k-1})$ is the scalar product between the two vectors, i.e. the predicted value. $h^o(k)$ is the conditional variance at time $k$; the conditional mean is simply

$$m^o(k) = A^o \cdot X_{k-p}^{k-1}$$

(14c)

The process is called heteroscedastic because of the conditional variance that is varying in time (Engle, 1982; Gourieroux, 1992). AutoRegressive Conditional Heteroscedastic (ARCH) processes are means zero, serially uncorrelated processes with nonconstant variances conditional on the past, but constant unconditional variances.

Let us now compute the log-likelihood of the observed data from

$$f = -\log \left[ \mathcal{Z}(x(0), x(1), ..., x(N); x_0, s_1, ..., s_N) \right]$$

$$= \frac{1}{2} \sum_{k=0}^{N} \left[ x(k) - A \cdot X_{k-p}^{k-1} \right]^2 \frac{1}{h^o(k)} + \frac{1}{2} \sum_{k=0}^{N} \log(h^o(k)) - \frac{N}{2} \sum_{k=1}^{N} \pi[s_k \mid s_{k-1}]$$

(15)

As before, one can easily derive reestimation formulae for the transition probabilities.

Engle (1982) derived an interesting result concerning estimation of the parameters. He showed that, if the autoregressive coefficients $(A^o)$ only appear in the conditional mean $(m^o(k))$ and the variances $(\sigma^o_i)$ only in the conditional variance $(h^o(k))$, Fisher’s information matrix is block-diagonal in the autoregressive coefficients and the variances. This means that the estimation of the autoregressive coefficients and the variances can be undertaken separately without asymptotic loss of efficiency. Furthermore, either can be efficiently estimated based only on a consistent estimate of the other. So, a possible procedure is to initially estimate the autoregressive coefficients $a_i^o$ by ordinary least squares (saying that all the variances are equal to one). From these estimated coefficients, a new estimate of the coefficients – by solving the corresponding Yule-Walker equations – can be constructed, while the variances $(\sigma^o_i)$ have to be computed by a nonlinear optimization procedure.

Computing the partial derivative of the log-likelihood in terms of a particular autoregressive coefficient yields the corresponding Yule-Walker equations:

$$\sum_{k=0}^{N} \phi_i(j, k) a_i^o = \phi_i(0, j), \quad j = 1, ..., p$$

(16a)

where the $\phi_i(j, k)$ $(i = 0, ..., p; j = 1, ..., p)$ are defined for each state $s$ as

$$\phi_i(j, k) = \frac{1}{N} \sum_{k=0}^{N} x(k-j) x(k-j)$$

(16b)

This means that the linear equation (16a) has to be solved for all the states, after each iteration on the whole data set. Unfortunately, the Toeplitz property does not hold any more, but the $\phi_i(j, k)$ matrix $(i = 1, ..., p; j = 1, ..., p)$ is symmetric positive definite so that we can use Cholesky decomposition (as for the LPC covariance method; see Rabiner & Schafer, 1978). Indeed, assuming that there is no linear relationship between $p$ successive samples:

$$0 < \frac{1}{N} \sum_{k=0}^{N} h^o(k) \left( \sum_{j=1}^{p} a_i^o x(k-j) \right)^2 = \sum_{i=1}^{p} \sum_{j=1}^{p} \phi_i(j, k) a_i^o a_j^o$$

As mentioned before, the variances can be computed thanks to a gradient descent. Let us define the vector $\Sigma = (\sigma^o_1, \sigma^o_2, ..., \sigma^o_p)^t$.

Engle (1982) proposes to calculate the elements of Fisher’s information matrix related to the variance:

$$J^o = E \left[ \frac{\partial^2 f}{\partial \sigma^o_i \partial \sigma^o_j} \right]$$

(17)
where is the log-likelihood, as defined by (15). One can show (Engle, 1982; Gourieroux, 1992) that, if we define the vector \( \mathbf{x}(k) \) as \((1, x(1-1), \ldots, x(k-p))\), \( Y \) can be estimated for each state \( s \) by

\[
J_s^Y = \frac{1}{N^Y} \sum_{k \in s} \mathcal{L}(k) z(k)^2 \left[ \frac{2}{\mathcal{L}} \right]^2
\]

(18)

From this estimate, we can compute the variances \( \sigma_{ij} \), for each iteration \( i \) on the training set, by

\[
[\Sigma_l]^{(i+1)} = [\Sigma_l]^{(i)} + \frac{1}{N^Y} \left( [\Sigma_l]^{(i)} - \frac{2}{\mathcal{L}} \right) \sum_{k \in s} \frac{\partial \alpha}{\partial \mathbf{s}^2}
\]

(19)

where \( \mathcal{L} \) is the contribution to the log-likelihood associated with observation \( x(k) \), that is, \( \mathcal{L} = -\log p_x(x(k) | X_{k-p}, \Theta_{s(t)}) \), and where the parameters are taken from previous iteration \( i \).

Of course, any non-linear optimization procedure can be used (we intend to use Daviden-Fletcher-Powell minimization).

5. CONCLUSION

In this paper, we reviewed the basic ideas underlying the use of predictive models for speech recognition with hidden Markov models. These models assume that the acoustic vectors are generated by a linear or nonlinear process corrupted by additive noise. Both linear and nonlinear models can be used for prediction, and both forward-backward algorithm and Viterbi algorithm can be used for training. Thereafter, we introduce an extension of the linear case, i.e. we permit both the speech samples and the autoregressive coefficients to be corrupted by noise. Indeed, when working at the sample level, this is a simple way to take the intra and inter speaker variability into account, that is, to allow variability in the transfer function. In fact, this is exactly what we are doing when extracting LPC coefficients and clustering them with Gaussian distributions. The advantage here is that we directly introduce the variability at the sample level. This leads to processes that are known as AutoRegressive Conditional Heteroscedastic (ARCH) processes, with nonconstant variances conditional on the past.

REFERENCES


