AUTOMATIC DERIVATION OF
HMM ALTERNATIVE PRONUNCIATION NETWORK TOPOLOGIES

Alessandro Falaschi, Massimo Pucci
La Sapienza University of Rome, INFO-COM Dpt
Via Eudossiana 18, 00184 Roma - ITALY

ABSTRACT

A new method for the automatic derivation of HMM topologies is presented. At first, the speech signal is segmented into acoustical units by using an Ergodic HMM. Then, a lattice structure is built from all the observed pronunciations. The lattice is thus pruned according to an information-theoretic criterion, aiming to preserve only the more characteristic event sequences. A circuit-free HMM topology is finally built after a proper state number assignment.

The method naturally permits the sharing of phonetically-motivated observation densities within different HMM and states. Results for a speaker-independent recognition experiment are given.

Keywords: Hidden Markov Models; Structure Inference; Information Measures; Alternative Pronunciations.

1. INTRODUCTION

As well known, the pronunciation of a word by different speakers varies not only because of voice quality changes, but also (and even more heavily) because of the speaker particular attitudes and idiosyncrasies. On the other hand, the phonetic units effectively produced by each talker constitutes a set of well defined acoustical events. These simple consideration make attractive the idea of describing words by an alternative pronunciation network. Although such a representation seems particularly well suited for HMM based speech recognition, the lack of deep knowledge about phonological processes often hinders the a priori definition of linguistically-motivated pronunciation networks, so that the acoustic variability of phonemes is often accounted for through approximate stochastic models, such as discrete, mixture of continuous, or semi-continuous observation densities.

The aim of this paper is to illustrate the possibility of deriving accurate multi-speaker pronunciation networks for words, in which an automatically defined, reduced set of acoustic-phonetic units are employed for the description of the acoustical alternatives. The word models topology inference method relies on dynamic programming techniques, making use of likelihood scores and information theoretical criteria. The use of these metrics is one of the main differences from [1]. Moreover, the criterion here exposed improves the technique introduced in [2].

At first, an ergodic HMM (EHMM) of speech is derived, accounting for the acoustical and phonotactical features of speech [3]. The EHMM is then used for the segmentation and labeling of the training data into a finite set of acoustic-phonetic elementary events, identified by the EHMM states. After having collected together all the label sequences for the same word, the graph structure building procedure can be executed. The first step is the alignment of all the collected label sequences for a word to the same length. Then, a lattice structure is built, in which all the (synchronously aligned) occurrences of the same label are collapsed in a single token, while its neighboring labels and occurrence count are kept updated. These latter are used in a forward-backward computation, aiming to ascribe the relevance of each branch of the lattice. Thus, a pruning procedure is executed, so that only typical pronunciation alternatives are retained. Finally, the pronunciation network is built by checking for the absence of loops within the resulting graph.

Several benefits come with the outlined approach. The first is an accurate and discriminative description of typical realization of words. The second is due to the use of an EHMM for the initial segmentation, which allows the sharing of the same acoustic descriptions by different states of different models, similarly to semi-continuous [4] or clustered [5] density approaches to hidden Markov modelling. Finally, the inferred network topology allows a clear interpretation of the results in terms of alternative phonetic realization.

The performances of the method are tested on a database constituted by 36 speakers uttering 33 words each.

2. THE ERGODIC HMM

An Ergodic HMM [6][7] is usually defined by a set of states $s_i$, $i=1..N$, a set of transition probabilities $a_{ij}$, a set of observation densities $b_j(y)$ and a set of initial probabilities $p_i$. We choose to add two more states, $s_0$ and $s_{N+1}$, which do not possess any observation density, and which can be regarded as the entry and exit states of the EHMM. In this case, the $p_i$ values are replaced by the first row of the transition matrix $a_{0i}$, and the EHMM is functionally equivalent to a left-to-right HMM, with the only difference that in this case circuits of states are allowed.

The adopted observation density function $b_j(y)$ are multivariate gaussian, described by two 8-dimensional vectors, which are the mean $\mu_j$ and standard deviation $\sigma_j$ of the Mel-Cepstrum
Coefficients observed for that state. The C_0 term has not been used. Moreover, the state dwelling time is implicitly modeled by a non-stationary assumption on the state self-transition probability \[8\], so that the state loop probability obeys an exponential decay law.

The initial estimates of the b_j(y) means are obtained through a binary-tree vector quantization [9] of the Mel-Cepstrum coefficients computed for the training data, and the variances estimated as the ones of the parent cluster. The VQ labelling of the same data is analyzed for gathering an initial estimate of the state probability transition matrix and of the duration parameters. All the parameters of the EHMM are then re-estimated by the Vouet method [10], which performs a forward-backward Viterbi algorithm for the derivation of the weight coefficients required for the parameters estimation. Finally, the transition matrix is gradually pruned during the re-estimation iterations [11], by eliminating the least probable transitions, so that a manageable complexity model is finally obtained.

3. THE TOPOLOGY INFERENCE PROCEDURE

In this section, the HMM topology inference procedure will be illustrated. It is based on the labeling of speech against the states of the EHMM, obtained by a modified version of the Viterbi algorithm [8]. The decoded EHMM state (or label) at instant \(n \) for the \( k^{\text{th}} \) occurrence of a word will be indicated in the following as \( S_k(n), n = 1, \ldots, L_k \).

3.1 Ranking of the training sequences

The \( \{S_k(n)\} \) sequences obtained by EHMM labeling of the same word uttered by different speakers are initially ranked according to a criterion which accounts for both their representativity and length deviation from the average. The labels occurrence count allows the evaluation of their unpredictability, which for label \( S_k(n) \) takes the value

\[
I_k(n) = - \log_2 p(S_k(n))
\]

so that the rank of the sequence \( \{S_k(n)\} \) can be assessed, in ascending order, through the value

\[
\text{Score}(k) = I_k(n) \times (1 + \text{abs}(L_k - L_m)) / L_m
\]

in which \( L_k \) and \( L_m \) are the \( k^{\text{th}} \) sequence length and the mean sequence length over all the \( k \), and \( I_k(n) \) is the average unpredictability of the labels occurring within the sequence.

3.2 First DP stage

This step aligns all the \( \{S_k(n)\} \) sequences to the same length, namely the length of the best sequence as given by (2). Each sequence is aligned against all the previously processed ones by a Dynamic Programming procedure which performs a one-to-many comparison, because the previous sequences have already been warped to the same length. Thus, the DP equation for the alignment of \( \{S_k(n)\} \) against the equal-length sequences \( \{S_h(W_h(m))\}, h \neq k, m = 1, \ldots, L_h \) is expressed as

\[
D(S_k(n), m) = d(S_k(n), m) + \min \{ D(S_k(n-1), m-1), D(S_k(n-1), m-2), \alpha D(S_k(n-1), m) \}
\]

in which \( \alpha \) takes values such that the Itakura warping constraints [12] are always satisfied. The value \( d(S_k(n), m) \) is the minimum Likelihood Distortion (LD) between the label \( S_k(n) \) and the labels aligned at time \( m \), as given by

\[
d(S_k(n), m) = \min_h \{ \text{LD}(S_k(n), S_h(W_h(m))) \}
\]

in which \( W_h(m) \) is the warping path that minimizes (3) for sequence \( S_h \) and \( \text{LD}(i,j) \) is evaluated as the logarithm of \( b_{ij} \), i.e. the probability of observing the mean of the density of the EHMM state \( S_i \) from the density of state \( S_j \). The analysis of \( W_h(m) \) allows one to keep track of how many frames have been associated to the same time position "m".

3.3 Building of the labels lattice

This stage collapses the matrix built by the aligned \( \{S_k(n)\} \) into a Lattice, \( L(n,m) \), in which all the occurrences of the same label (belonging to different sequences), warped to the same time epoch "m", are collapsed into a single entry, so that each label occurs only once per column.

During this process, two other data structures are kept updated: the Lattice Nodes Occurrence Count lattice NOC(n,m) which reports the number of labels associated to each node, and the Incidence Matrix IM(n,m,j) whose elements indicate if the nodes \( (n,m) \) and \( (j,m-1) \) are connected by any of the training sequences. Finally, a Partial Count array PC(m) stores the sum over \( n \) of NOC(n,m), thus indicating the total number of labels aligned at time \( m \).

3.4 Pruning of the pronunciation lattice

This stage performs the pruning of the derived lattice structure, so that a graph topology responsible only for the main phenomena occurring within the training data is obtained. The adopted method is substantially different from that exposed in [2]. The actual pruning strategy relies on a forward-backward computation operated on the lattice. It computes the probability of the best path connecting the left and the right edges of the lattice, and passing through any pair of contiguous nodes of it. Then, the transitions for which the best path probability is lower are deleted.

In order to describe the Fw-Bw computation, let us express the probability of a path crossing the lattice in terms of the information associated to the node \( (n,m) \) of the lattice, and given by

\[
I(n,m) = - \log_2 (\text{NOC}(n,m) / \text{PC}(m))
\]

The information quantity associated with the highest probability path connecting the lattice left edge to the node \( (n,m) \) can thus be computed by the DP recursion

\[
F(n,m) = I(n,m) + \min_{j \in \text{IM}(n,m,j)} \{ F(j,m-1) \}
\]
in which the incidence matrix constrains the allowable node sequences.

In the same way, the information measure related to the highest probability path starting from node \((n,m)\) and reaching the right edge of the lattice is evaluated by the backward relation:

\[
BI(j,m-1) = \min_{n:IM(n,m)\neq 0} \{ I(n,m) + BI(n,m) \} \tag{8}
\]

The sum

\[
I(n,m,j) = FI(n,m) + I(j,m+1) + BI(j,m+1) \tag{9}
\]

gives the information measure of the path which connects the left and right edge of the lattice and which has the maximum probability within all the paths passing through the node pair \((n,m);(j,m+1)\). After (7) and (8) have been evaluated for all the lattice nodes, (9) can be repeatedly computed for all the lattice allowed transitions. The node pair for which (9) takes the maximum is thus deleted by zeroing the relative entry in the incidence matrix, and the process repeated until a requested amount of transitions has been eliminated.

3.5 Network states numbering

The lattice pruning stage does not solve the problem of deriving a proper model topology. In fact, usable HMMs require the absence of loops spanning chains of states, and this fact must be checked before proceeding to build the model. This control basically consists of assigning the same state number to different nodes only if they are connected by a transition which has not been deleted at the previous stage, they are equally labelled, and no other (allowed) path connects the two. Finally, a transition matrix is built on the basis of the state number assignment and the allowed lattice transitions, by adding two more null (density-less) states which act as entry-exit points for the word HMM.

4. EXPERIMENTAL RESULTS

4.1 Data Base Composition

Thirty-six speakers, either male or female, uttered 33 words in isolation, comprising the digits set and a command words set. The voices were sampled at 8 kHz in a 8-bits PCM format, and successively expanded to a 12-bits linear format. The speech was analyzed using a sliding 30 m sec window at a 75 Hz frame rate. After pre-emphasis and Hamming windowing, 9 Mel-Cepstrum coefficients were computed for each frame. The speech material was then subdivided into two sets: 20 speakers served as training data, and 16 for the test.

4.2 Models complexity

An EHMM made by 34 states, 283 transitions and 32 densities has been estimated on the training set, and then used for its labeling. The topology inference procedure has been run on these label sequences, and four different sets of HMM of word have been derived, for various pruning ratios. Tab. 1 reports the complexity of each set of models, together with the average branching factor and average number of states per word model.

<table>
<thead>
<tr>
<th>States</th>
<th>N. of States</th>
<th>N. of trans.</th>
<th>ABF</th>
<th>States/Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>2646</td>
<td>4850</td>
<td>1.83</td>
<td>80.0</td>
</tr>
<tr>
<td>75</td>
<td>1391</td>
<td>2120</td>
<td>1.52</td>
<td>42.0</td>
</tr>
<tr>
<td>92</td>
<td>972</td>
<td>1240</td>
<td>1.27</td>
<td>29.3</td>
</tr>
</tbody>
</table>

Tab 1 - Models complexity

4.3 An example

Fig. 1 illustrates the resulting HMM topology for the word *copy* (copy), with different degrees of pruning. On the bottom, a phonetic interpretation of the different sections of the word model is given. As it can be observed, the stressed vowel exhibits a greater consistency of acoustical realizations between different speakers, while major variability is observed for the plosive, the diphthong and the fading of the word-ending vowel. At the 82% level of pruning, the model tends to be nearly unifiable, and it can be noticed that only the most relevant acoustic-phonetic features are retained, as for instance the burst of /k/, represented by the second state.

4.4 Recognition performances

The word uttered by the 16 test speakers have been used for evaluating the recognition performances of the system. A first experiment involved the HMM obtained after 82% of pruning, for which 81.9% recognition rate was observed. Then, the models from which 75% of transitions are pruned were experimented, resulting in a 80.8% recognition rate.

5. DISCUSSION AND CONCLUSIONS

The recognition results given above must be regarded as purely indicative, as the use of only 20 speakers as training set can not be truly significant. Moreover, a tendency has been noticed for the EHMM in being highly biased by the initial estimated of its parameters. Different initialization procedures probably will give better results. This section is mainly devoted to a discussion of the experiments here reported. At first, it must be stressed that only 32 densities accounted for the description of the acoustical properties of 36 words. Then, the lowering of the recognition rate for the more detailed models can be due to an higher probability of finding the proper density sequence also for the incorrect models. Again, a better definition of the EHMM could solve the problem. As probably better results could be obtained by using more densities, their number will remain at least one order of magnitude smaller than that for the un-shared densities case. This low dimensionality will allow the adoption of more computationally-intensive metrics, as full covariance matrices, and/or augmented observation vectors, as is the case when dynamic features are incorporated. A second observation is due to the fact that only densities are shared, and each state possesses its own duration model. This fact greatly contributes the improvement of discrimination within different words.
REFERENCES


Fig. 1 - The HMM network for the word *copia* (copy)

- a) - No pruning; b) - 50%; c) - 75%; d) - 82% of pruning.

The latter transition diagram is related to the state observation density mean.