Techniques For Modelling Phonological Processes In Automatic Speech Recognition

Harriet Jane Nock

Darwin College

May 31 2001

This dissertation is submitted to the University of Cambridge
for the degree of Doctor of Philosophy

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where stated. It has not been submitted in whole or part for a degree at any other university. The length of this thesis including footnotes and appendices does not exceed 29,500 words and includes no more than 40 figures.

Abstract

Systems which automatically transcribe carefully dictated speech are now commercially available, but their performance degrades dramatically when the speaking style of users becomes more relaxed or conversational. This dissertation focuses on techniques that aim to improve the robustness of statistical speech transcription systems to conversational speaking styles.

The dissertation shows first that the performance degradation occuring as speech becomes more conversational is severe and is partially attributable to differences in the acoustic realizations of sentences. Hypothesizing that the quantifiably wider range of pronunciation in conversational speech contributes to these differences, the dissertation then focuses on techniques for modelling the phonological processes underlying pronunciation change. Such techniques may be classified as *explicit* (operating at or close to the level of the word pronunciation dictionary) or *implicit* (operating at or close to the subword statistical models of the acoustic signal) and both types are considered.

An existing explicit technique, motivated by linear phonology and originally evaluated on a dictated speech task, has recently been extended for conversational speech tasks. Rather than model pronunciations using phonemic units (which are by definition abstract units with highly variable acoustic realizations), a statistical mapping is constructed from the abstract phonemic units to their context-dependent realizations as surface phonetic units (which are by definition less abstract and less variable in acoustic realizations). If the map from phonemic units to phonetic realizations is sufficiently accurate, the task of modelling the acoustic realizations of words should be simplified. Small but statistically significant performance improvements can be obtained on the SWITCHBOARD transcription task. However, further experiments by the author and by other researchers suggest that schemes modelling pronunciation change in terms of speech “segments” have only limited potential.

This analysis suggests a more implicit approach capable of describing variable degrees of pronunciation change at levels below the segment may be more appropriate. This motivates investigation into a family of statistical models that could form the basis of such an approach: *Loosely-coupled* or *Factorial Hidden Markov Models* (FHMMs). The theory of FHMMs is described and it is then shown that they generalize several standard speech models. Two specific FHMMs are investigated. Analysis of an existing FHMM in the literature - the *Mixed-Memory Assumption* FHMM - finds it has potential weaknesses for speech modelling. This leads us to propose a new FHMM - the *Parameter-Tied* FHMM - which makes fewer a-priori assumptions about the data to be modelled. Estimation and decoding of FHMMs is potentially computationally expensive, so approximate algorithms are also developed. Empirical studies using the ISOLET speech classification task show (1) FHMMs scale to speech modelling tasks (2) the Parameter-Tied FHMM achieves performance comparable to the Mixed-Memory Assumption FHMM for speech modelling and (3) identify an approximate algorithm for decoding and estimation that is adequate for more extensive experimentation. A short study using the TI DIGITS task shows that FHMMs can be scaled to continuous speech recognition whilst continuing to achieve classification performance competitive with more conventional models.

The thesis ends with a summary and possible directions for future research.

Keywords: speech recognition, pronunciation variability, pronunciation modelling, decision tree pronunciation model, Hidden Markov Model, Factorial Hidden Markov Model, multiple loosely-coupled time series, variational approximation, chainwise Viterbi algorithm, ISOLET, TI DIGITS, SWITCHBOARD, MULTIREG.

Acknowledgements

I would like to thank my primary supervisor Steve Young and my secondary advisors or mentors Bill Byrne, Mark Gales, Sanjeev Khudanpur, Mari Ostendorf and Mike Riley for their guidance at different points during my postgraduate career. It has been a privilege to work with them.

I am indebted to Fred Jelinek, Mari Ostendorf and Steve Young for giving me the opportunity to work in their research groups and at the WS97 workshop and also to the members of those groups for technical guidance and for broadening my academic and non-academic horizons.

The systems administrators and administrative staff at CUED, CLSP and SSLI have provided assistance beyond the call of duty, as have those who proof-read.

I am grateful to Cambridge University Engineering Department, the Center for Language and Speech Processing at The Johns Hopkins University, Darwin College, EPSRC, the Royal Academy of Engineering, the Signal, Speech and Language Interpretation Lab at the University of Washington and to Tony Robinson, Phil Woodland and Steve Young for financial assistance.

My final thanks are due to friends, housemates and family.

Abbreviations

ASR	Automatic Speech Recognition
DBN	Dynamic Bayesian Network
FHMM	Factorial Hidden Markov Model
HMM	Hidden Markov Model
LWER	Lattice Word Error Rate
ML	Maximum Likelihood
MM-FHMM	Mixed-Memory Assumption Factorial Hidden Markov Model
PDF	Probability Density Function
PER	Phone Error Rate
PMF	Probability Mass Function
PT-FHMM	Parameter-tied Factorial Hidden Markov Model
SWB	Switchboard Conversational Speech Corpus
WER	Word Error Rate
%C	% Correct (classification tasks)
%Acc	% Accuracy (recognition tasks) where % Acc = 100 - WER

Mathematical Notation

$\|A\|$	determinant of matrix A
$cof_{ij}(A)$	cofactor of A_{ij} in A
A^{-1}	inverse of matrix A
A^T	transpose of vector/matrix A
$Card(S)$	cardinality of a set S
$P(.)$	probability mass function
$p(.)$	probability density function

General Model Notation

λ	current values for all model parameters
$\hat{\lambda}$	updated values for all model parameters
$\hat{\theta}$	updated model parameter θ
i^k	state in k-th chain
j^k	state in k-th chain
θ_k	set of states in k-th chain
$N_k = Card(\theta_k)$	number of states in k-th chain
N	number of states in each chain, when all chains are assumed to have same number of states
$I = (i^1, \ldots, i^K)$	metastate
$J = (j^1, \ldots, j^K)$	metastate
Θ_{meta}	set of metastates in a loosely-coupled model
$\mathbf{S}\|s^k = j^k$	the set of $\mathbf{S} = (s^1, \ldots, s^K) \in \Theta_{meta}$ such that $s^k = j^k$
D_k	dimensionality of observations in kth time series
D	dimensionality of observations in each time series, when all streams are assumed to have same dimensionality
o_t^k	observation at time t in time series k
$\mathbf{O}_t$	combined observation vector $(o_t^1, \ldots, o_t^K)$
$\mathbf{O}$	observation sequence $\mathbf{O}_1, \ldots, \mathbf{O}_T$
s_t^k	state occupied in chain k at time t
$\mathbf{S}_t$	metastate $(s_t^1, \ldots, s_t^K)$ occupied at time t
$\mathbf{S}$	sequence of metastates $\mathbf{S}_1, \ldots, \mathbf{S}_T$
$\mathcal{S}$	set $\{\mathbf{S}\}$ of possible metastate sequences $\mathbf{S}$ of length T

MM-FHMM Notation

x_t^k	hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_{t-1}$ which determines the matrix used for the transition into s_t^k
$\mathbf{X}_t$	vector of hidden variables $(x_t^1, \ldots, x_t^K)$
$\mathbf{X}$	the sequence $\mathbf{X}_1, \ldots, \mathbf{X}_T$
$\mathcal{X}$	set $\{\mathbf{X}\}$ of possible $\mathbf{X}$ sequences of length T
y_t^k	hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_t$ which determines the output probability for o_t^k
$\mathbf{Y}_t$	vector of hidden variables $(y_t^1, \ldots, y_t^K)$
$\mathbf{Y}$	the sequence $\mathbf{Y}_1, \ldots, \mathbf{Y}_T$
$\mathcal{Y}$	set $\{\mathbf{Y}\}$ of possible $\mathbf{Y}$ sequences of length T
$\pi^{kl}(j^k)$	MM-FHMM cross-prior matrix
$a^{kl}(j^k\|i^l)$	MM-FHMM cross-transition matrix
$b^{kl}(o_t^k\|i^l)$	MM-FHMM cross-emission distribution
$\phi^k(l)$	MM-FHMM transition-related mixture weights
$\psi^k(l)$	MM-FHMM observation-related mixture weights

PT-FHMM Notation

$C^{obs,k}$	an equivalence class of metastates for stream k observation distributions
$\mathcal{C}^{obs,k}$	set of all such equivalence classes $C^{obs,k}$ for stream k
$C^{trans,k}$	an equivalence class of metastates for chain k transition distributions
$\mathcal{C}^{trans,k}$	set of all such equivalence classes $C^{trans,k}$ for chain k

Contents

List of Figures

1
Introduction

Voice-controlled systems and software which automatically produce transcriptions of speech are becoming increasingly common. The performance of these systems has improved sufficiently that people are often surprised to learn that the automatic speech transcription (or recognition) problem is far from solved. The reason for this discrepancy is clear once we examine the dimensions of difficulty in the task:

- *noise*: quiet ↔ environmental and channel noise;
- *speaker diversity*: small set of known speakers ↔ multiple, unknown speakers;
- *fluency of speech*: isolated words ↔ continuous speech;
- *speaking style*: carefully articulated speech ↔ casual or conversational speech;
- *accent*: native speaker ↔ non-native speaker;
- *vocabulary words and vocabulary size*: restricted, easily distinguishable words ↔ unconstrained vocabulary;
- *dialogue initiative* (for interactive systems): tightly constrained by system ↔ mixed- or user-controlled.

Today's systems achieve good performance by restricting tasks along one or more of these axes. Voice-controlled systems such as air ticket reconfirmation or cinema information lines typically use a highly restricted, system-controlled dialogue structure to ensure that responses are (mostly) short utterances using a restricted vocabulary. Transcription systems for desktop PCs perform best when used in a quiet environment by a known user employing carefully dictated speech. In contrast, the goal of current speech research is to design systems which operate well at the extremes of all dimensions of difficulty. Tasks being addressed by the research community at the time of writing include the transcription of real radio and television news broadcasts (eg. [57]) and the transcription of informal telephone conversations between strangers (eg. [54]). The most ambitious task under consideration requires the "processing (transcription, query, search and structural representation) of audio recorded from informal, natural, and even impromptu meetings ... where the conversation may take place without any preparation, so that we cannot require special instrumentation to facilitate later speech processing (such as close-talking or array microphones)" [111]. These types of problem are not yet solved: recently reported benchmark error rates are 12-15% for broadcast news transcription [66], 25-40% for conversational telephone speech transcription [67] and 46.5% for preliminary attempts at meeting transcription [111].

This thesis focuses on just one dimension of difficulty, *speaking style*, and more specifically issues associated with transcribing speech which is conversational, as opposed to read or dictated. It begins by showing that casual or conversational speaking styles pose greater difficulty to today's state-of-the-art transcription systems than do formal or dictated speaking styles; it then supplies evidence to suggest that this may be attributed, at least in part, to the increased *pronunciation* or *phonological variability* found in conversational speech. The thesis then identifies assumptions made by current recognizer designs that may underlie this lack of robustness to conversational pronunciation variability. The background is then set for the main part of the thesis, which proposes and evaluates schemes for improving transcription accuracy on this type of speech.

1.1 Detailed Organization Of Thesis

The next chapter introduces terminology by briefly reviewing the statistical framework for speech transcription used in almost all conventional systems. Chapter 3 begins by presenting an empirical study which shows conversational speaking styles are more "difficult" for current state-of-the-art transcription systems. It then discusses the differences between read or dictated speech and casual, conversational speech and uses the results of the empirical study to conclude that at least part of the difficulties associated with conversational speech transcription may be attributed to the increased phonological variability found in conversational speech. The chapter then highlights aspects of current recognizer designs that may be inadequate to model this increased variability: the *pronunciation dictionary* and the *stochastic model of acoustic observations*. Schemes having the ultimate goal of improving robustness to phonological variability fall into two broad classes. The distinction will be defined more carefully in Chapter 4 but in general terms *explicit* approaches modify the pronunciation dictionary to be more representative of the style of speech to be transcribed whereas *implicit* approaches operate closer to the level of the model of the acoustic observations. The main part of this thesis considers both explicit and implicit schemes. Chapter 4 reviews explicit pronunciation modelling schemes and Chapter 5 extends a standard technique for dictated speech pronunciation modelling to conversational speech. Analysis of the results motivates investigation into an implicit pronunciation modelling approach: specifically, an approach that would allow more information about speech production and phonological processes to be incorporated into the acoustic modelling scheme. Chapter 6 reviews related modelling schemes, many of which are generalized by the models discussed in Chapter 7. Parameter estimation and decoding for this family of models is potentially computationally intensive. New algorithms may be necessary to apply them to large vocabulary speech tasks and Chapter 8 presents some possible schemes. Chapter 9 evaluates the new models and algorithms. The thesis ends with conclusions and an outline of possible future work.

2

Statistical Framework For Speech Recognition

This chapter introduces terminology used in later chapters by briefly reviewing the statistical framework for speech transcription used in almost all conventional systems.

2.1 Basic Framework

The statistical formulation of the automatic speech recognition (ASR) problem is due to [9]. An *acoustic preprocessor* converts the speech waveform into a sequence of observations or acoustic vectors $\mathbf{O} = \mathbf{o}_1, \ldots, \mathbf{o}_T$, which represents the acoustic evidence upon which the recognizer will make a decision. The recognizer seeks the word sequence $\mathbf{W}^*$ such that

$$\begin{aligned} \mathbf{W}^* &= \arg\max_{\mathbf{W}} p(\mathbf{W}|\mathbf{O}) && (2.1) \\ &= \arg\max_{\mathbf{W}} p(\mathbf{O}|\mathbf{W}) P(\mathbf{W}) && (2.2) \end{aligned}$$

where $\mathbf{W} = w_1, \ldots, w_N$ denotes a valid word sequence. Probability $p(\mathbf{O}|\mathbf{W})$ is provided by an *acoustic model* and $P(\mathbf{W})$ by a *language model*; the parameters of these models are typically estimated independently. Maximization of Equation 2.1 over all $\mathbf{W}$ is referred to as *decoding*. These stages will now be described in more detail.

2.2 Acoustic Preprocessing

The acoustic preprocessing scheme is often dictated by the distributional assumptions made by the chosen acoustic modelling scheme. Thus, when the acoustic modelling

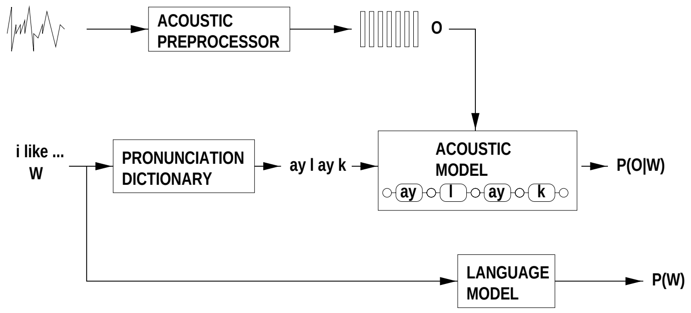


Figure 2.1 *Statistical Framework For Speech Transcription*

scheme is based on Hidden Markov Models (HMMs) using diagonal covariance, multivariate Gaussian observation distributions (see Chapter 6), the acoustic representation is generally chosen to be vectors of Mel-frequency Cepstral Coefficients (MFCCs) since the elements are approximately decorrelated.

MFCCs [32] are obtained using the following procedure. The digitized speech waveform is divided into (possibly overlapping) sections or *frames* of equal length. A window (eg. Hamming) is applied to each frame to remove boundary effects. The Fast Fourier Transform is then used to produce a spectral representation eg. [132], which is filtered using Mel-scale-spaced [159] filter banks, as in eg. [180]. The final step takes the log of the Fourier components, and then rotates the resulting vector using a Discrete Cosine Transform eg. [132]. An optional cepstral truncation step drops high order cepstral coefficients to reduce the dimensionality of the acoustic vector, which can be viewed simply as a procedure for spectral smoothing or as a means of retaining the more perceptually important parts of the speech spectrum. In a typical final step, first-order (*delta*)and second-order (*delta-delta*) regression coefficients are appended to the acoustic vector [4, 46]. This is a heuristic but effective technique compensating for conditional independence assumptions made by HMMs (which are discussed in more detail in Chapter 6).

2.3 Acoustic Modelling Using HMMs

The acoustic model estimates $p(\mathbf{O}|\mathbf{W})$. Most current commercial and research systems use HMMs, a stochastic model of discrete time series data. This is partly due to the existence of efficient algorithms for maximum likelihood parameter estimation and for recognition, as described in eg. [131].

For isolated word classification tasks with sufficient training data, each $\mathbf{W}$ is a single word and can be modelled by a single HMM. For tasks with insufficient data or for continuous speech tasks, an HMM is formed for each word or word sequence $\mathbf{W}$ by concatenating HMMs modelling subword units; these are generally *phoneme-like units*, although other units have been considered eg. syllables [50] and diphones [137, 134]. The mapping from word sequences to sequences of subword units is performed by a *pronunciation dictionary* or *lexicon*. This scheme for creating a sentence-level acoustic model is often termed the *beads-on-a-string* procedure.

Context-dependent models acknowledge the influence of context on the realization of sounds. A separate model is constructed for each subword unit in the context of an arbitrary number of units to the left and/or right. *Triphone*-based systems are common, in which sub-word units are phonemes and each HMM represents a phoneme in the context of a distinct preceding and following phoneme. Triphone models may or may not be used across word boundaries, and may or may not be made additionally dependent upon presence or absence of a word-boundary. Context-dependent modelling significantly increases the number of model parameters to be estimated: the most common solution uses some form of *parameter* or *distribution tying*, in which equivalence classes are defined between model constructs (eg. HMM states) and then constructs in the same class share the same parameters for associated distributions. Schemes for determining classes include [7, 95, 122].

Chapter 4 discusses formation of the acoustic model in more detail.

2.4 Language Modelling

The language model provides estimates of

$$P(\mathbf{W}) = \prod_{n=1}^{N} P(w_n | w_{n-1}, \ldots, w_1)$$

where this follows by the chain rule. Since typical vocabulary sizes V are in the tens of thousands or more, the parameter space is too large to allow robust estimation. The number of parameters can be reduced by defining equivalence classes between word histories using a function $h(w_{n-1}, \ldots, w_1)$ and assuming

$$P(\mathbf{W}) = \prod_{n=1}^{N} P(w_n | h(w_{n-1}, \ldots, w_1))$$

Language Modelling research considers functions h and methods for obtaining robust model parameters $p(w_n | h(w_{n-1}, \ldots, w_1))$ from finite training data. Most state-of-the-art recognizers use the N-gram model in which:

$$h(w_{n-1}, \ldots, w_1) \overset{\text{def}}{=} w_{n-1}, \ldots, w_{n-N+1}$$

Typically $N = 2, 3, 4$ (termed bi-, tri- or four-gram models respectively). Maximum-likelihood estimates for N-gram model parameters are simply relative frequency estimates. However, for a vocabulary of V words, there are V^N potential N-grams and training data is finite: even for $N = 2, 3$ it may be too sparse to obtain reliable estimates if V is relatively large. Smoothing techniques attempt to improve the robustness of parameter estimates; examples include deleted interpolation [74], discounting eg. [116], back-off [80] and maximum entropy models eg. [145]; an empirical comparison of techniques is provided by [24].

The N-gram model captures only local constraints and ignores higher-level structure. Many more sophisticated models have been investigated. Most have limited success eg. [8] although some recent papers report interesting results eg. [11, 179, 22].

2.5 Decoding

The decoder seeks $\arg\max_{\mathbf{W}} p(\mathbf{O}|\mathbf{W})P(\mathbf{W})$. Decoding using full acoustic likelihoods $p(\mathbf{O}|\mathbf{W})$ is possible eg. [9]. However, for efficiency most decoders make a *Viterbi approximation* ie. letting $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$ denote a state sequence through the HMM for word sequence $\mathbf{W}$, then it is assumed $p(\mathbf{O}|\mathbf{W}) \approx \arg\max_{\mathbf{S}} p(\mathbf{O}, \mathbf{S}|\mathbf{W})$. The maximizing state sequence can be obtained using the efficient Viterbi algorithm [168, 131, 181].

Decoding is a search problem. Since exhaustive search is generally intractable for large vocabulary tasks, search reduction strategies such as path pruning are employed and consequently the hypothesis obtained may not be optimal. Decoding research attempts to improve the organization and representation of the search space, the schemes for reducing search and the computation required to evaluate hypothesis costs under particular search schemes eg. [115, 122, 41]. A standard high-level scheme for reducing search costs uses a multiple stage approach to the search: *lattices* of alternative hypotheses are produced for each utterance using simple language and acoustic models; each lattice represents a reduced search space which can be explored using more sophisticated models eg. [112, 136, 152, 183].

2.6 Adaptation and Normalization

Models trained on domain-specific data generally outperform models trained on data from other domains where adequate domain-specific data is available. Similarly, models trained on speaker-specific data outperform models trained on data from several speakers where sufficient data exists. *Model-based adaptation* techniques attempt to modify generic language or acoustic models to achieve performance of domain- or speaker-specific models by adjusting model parameters to be more appropriate for the speech to be transcribed (with respect to eg. individual speaker or speaker group characteristics, noise characteristics, topic etc). *Normalization* techniques attempt to map all test speech to have characteristics closer to some canonical representation. Later chapters mention two techniques.

Maximum Likelihood Linear Regression (MLLR) is a successful acoustic model adaptation technique [35, 97]. Given data representative of the new speech to be transcribed (*adaptation data*), one or more linear transformations of model parameters are estimated using an ML criterion and then applied to transform the existing model set. These transformations capture general relationships between the original model set and the current speaker and (or) new acoustic environment and can adapt all model distributions based on limited adaptation data. Variations upon this basic theme include [49, 47, 61, 177].

Vocal Tract Length Normalization (VTLN) is a technique for normalizing the data from particular speakers to compensate for variability in vocal tract length eg. [3, 96]. A generic approach estimates "vocal tract length" parameters for each speaker and then warps test speech accordingly. As with MLLR, many variations have been investigated eg. [37, 170].

Many other adaptation and normalization techniques exist. Language model adaptation is considered in eg. [25, 71, 89, 90]. Acoustic model adaptation is considered in eg. [1, 48, 51, 106]. Normalization techniques are considered in eg. [88, 164]. Combinations of adaptation and normalization techniques, which may also be incorporated into model estimation, are discussed in eg. [129, 165, 182].

2.7 Obtaining State-of-the-Art Performance

Research systems typically incorporate many additional techniques to obtain state-of-the-art performance. Techniques falling into this category include (but are not restricted to): use of alternative criteria in estimation or decoding eg. [55, 176], more complicated acoustic models eg. [64, 103], more sophisticated language models eg. [117, 64, 89, 90], multiple pronunciation lexicons, possibly including pronunciation probabilities eg. [26, 139], confidence estimation eg. [21] and multiple recognizer hypothesis combination techniques eg. [43].

3

Speaking Style and Its Effects

The introduction stated that conversational (as opposed to read or dictated) speech poses serious problems for current transcription systems. This chapter begins by presenting experimental results to quantify that statement. It then discusses differences between dictated and conversational speech and, based on the experimental results, identifies increased pronunciation or phonological variability as one particular aspect of conversational speech that may create difficulties for current systems. This leads us to consider design assumptions in current recognizers that may result in an inadequate model of this type of variability. These potential weaknesses in current designs are used to motivate the research reported in the remainder of the thesis.

3.1 Effects Of Speaking Style On Transcription Accuracy

This section uses results obtained by the author and other researchers to demonstrate the effects of speaking style on current state-of-the-art transcription systems.

3.1.1 Evidence Obtained Using The MULTI-REG Corpus

The following experiment, performed by the author, illustrates the extreme variation in automatic transcription accuracy that results when speaking style changes from dictated to conversational. It investigates the hypothesis that relatively poor conversational speech transcription performance may be attributed to factors associated with speaking style.

The experiment extends work performed at SRI [171, 172] and is based upon their MULTI-REG corpus, which comprises conversations recorded in different speaking styles. The two-phase data collection procedure is outlined below; [171, 172] give more details.

The initial phase of data collection recorded fifteen *spontaneous* conversations on pre-assigned topics between newly acquainted individuals. *Narrow-band* (telephone bandwidth) and *wide-band* versions were recorded simultaneously[1]. The original speakers were later recalled to make two further recordings. They were handed transcripts of their original spontaneous conversations and asked to read them in two ways: first *reading* the transcript as if dictating it to a computer, then re-reading the same transcript as if *imitating* a conversation. Again, simultaneous narrow- and wide-band recordings were made for each of the two speaking styles. The resulting six renditions of the same conversation, controlled for two principle axes of variability (speaking style and recording bandwidth), were used to test the basic hypothesis.

[1] The narrow-band recordings thus resemble Switchboard in content and bandwidth [54].

Speaking Style	Wideband (WSJ models)	Narrowband (SWB models)
reading	26.2%	26.1%
imitating	39.7%	29.5%
spontaneous	62.4%	43.2%

Table 3.1 *WER for different speaking styles on the MULTI-REG test set*

The experiment uses two HMM-based acoustic model sets, trained using the standard HTK “recipe” as described in the HTK Book [180] or the Resource Management component of the HTK V2.2 release. The first model set is trained using narrow-band, conversational Switchboard (SWB) data [54]; the second is trained using wide-band, dictated Wall Street Journal (WSJ) data [128]. Both model sets were used to transcribe all of the MULTI-REG spoken utterances which have identical word-level reference transcriptions across the six conditions[2], whilst language model and decoder were fixed; the wide-band, WSJ models were used to transcribe the wide-band versions of the MULTI-REG recordings and the narrow-band, SWB models were used to transcribe the narrow-band (telephone-bandwidth) versions of the MULTI-REG recordings. Table 3.1 presents the corresponding word error rate (WER) results.

The first column shows the wide-band models trained on WSJ dictated speech are better at transcribing read (dictated) MULTI-REG data than the more spontaneous versions of the same utterances[3]. Note that the bandwidth of the test data is fixed (wide-band) across all three tests in the first column. The second column shows degradation cannot simply be attributed to the mismatch between the speaking styles in the training and test sets, since (reconfirming the results of [171]) narrow-band models trained on spontaneous speech are again better at transcribing read and imitated-spontaneous speech than truly spontaneous speech. Again, the bandwidth of the test data is fixed (narrow-band) across all three tests in this second column. Note that although general trends in results *degradation* can be compared between the wide and narrow-band experiments in the first and second columns, the absolute error rate results are not directly comparable across columns due to differences in the parameterization and model complexity of the WSJ wide-band and SWB narrow-band models.

Decreasing accuracy with increasingly casual speaking style is seen across both bandwidths, whilst the recording conditions and the words pronounced in the test data remained unchanged. The results suggest there are factors related to the acoustics associated with conversational speaking styles that are handled poorly by conventional HMM-based speech recognition systems. The results emphasise the importance of including spontaneous speech data in training when spontaneous data is to be encountered in testing, but it also seems that even training and testing on data with matched speaking styles offers only partial robustness to the degradation caused by style effects.

3.1.2 Evidence From Other Authors

The DARPA HUB4E Broadcast News Evaluation includes both spontaneous and more formal utterances in studio recording conditions. Every system entered into the 1998 evaluation produced less accurate transcriptions for the spontaneous speech condition

[2] To be precise, the utterances correspond to the `six_idnt.txt` subset of files on the MULTIREG CDs. These utterances have identical word-level reference transcriptions across the six conditions, excepting noises and special characters, and considering `UM` and `UH` identical.

[3] The word error rate for read speech under WSJ models in Table 3.1 is much higher than those typically reported on WSJ test sets. The material being read here however is the transcript of a spontaneous conversation which differs significantly from newspaper text [38].

(F1) than for the more formal, planned studio speech condition (F0) [126].

There are earlier papers examining the effects of speaking style upon recognition performance, but there is little focus on issues associated with natural conversational speech and large vocabulary tasks. The papers [133, 127] study small vocabulary, isolated and connected word recognition from a large range of speech styles (eg. normal, lombard, loud, soft), under motion and in different vibration conditions, and find that speaking style affects transcription performance. The studies by [169, 19] report that transcription performance improved as users interacted with one particular airline reservation system over time, and attribute the effect to changes in user speaking style.

3.2 Differences Between Dictated And Casual Speech

In the following discussion, typeface `AND` denotes a word, `/ae n d/` represents a phonemic baseform pronunciation and `[ax n]` denotes a phonetic realization of that pronunciation; phonemic and phonetic transcriptions use the ICSI Switchboard Transcription Project phone set (Appendix K).

The previous section showed there must be differences between dictated and conversational speech which make transcription more difficult for current transcription systems. However, whilst humans easily distinguish dictated and conversational utterances [99], linguists have found it hard to identify the distinguishing factors [185, 5]. The differences reported include:

- *phonetic and lexical deletions*: [12] reports that people asked to read text faster will increase the rate mostly by shortening each segment that is spoken; in contrast, a fast rate is accomplished in spontaneous speech by deleting phonemes eg. `NEXT+WEEK` `/n eh k s t/` + `/w iy k/` → `[n eh k s w iy k]`. [60] finds some words ‘swallowed whole’ in SWB. [60] illustrates these effects using the utterance `UNIVERSITY+OF+NEBRASKA`, which was pronounced `[y ux n ix v er s ix n d ax bcl b r ae s kcl k ae]` with `OF` deleted and the final syllable `/d ix/` of `UNIVERSITY` delayed until after the initiation of the nasal consonant in `NEBRASKA`[4];

- *changes in phoneme realizations*: occur more frequently in conversational speech. They may occur due to *assimilation* in manner or place of articulation eg. `GRANDPA` `/g r ae n p aa/` → `[g r ae m p aa]` or *monophthongization* of diphthongs eg. `/ay/` → `[ah]`. [60] reports a high frequency of glottal stops `/q/` in SWB acoustics, which replace syllable-final, usually voiceless, stops and occur at the beginning of many syllable-initial vocalic segments. Such changes are also found near phonetic deletions, since adjacent phones are often modified in order to preserve intelligibility [12] eg. `CAN'T` `/k ae n t/` → `[k ae_n t]`;

- *phonetic insertions*: insertions may be caused by asynchronous articulation errors eg. (in certain dialects of British English) `WARMTH` `/w ao m th/` → `[w ao m p th]` or linking *r* in vowel-vowel transitions eg. `DIRECTOR+OF` `/d ay r eh k t ao/` + `/aa v/` → `[d ay r eh k t ao r aa v]`;

- *spectral cues*: only the vaguest hint of “appropriate” spectral cues are present in spectrograms of conversational speech: formant transitions typically associated with specific segment types are missing or differ markedly from the form seen in more formal speech [60];

[4] This transcription differs very slightly from the transcription in [60]; the transcription here was taken from the November 1, 1996 data release.

- *syllable length*: [15, 40] find read syllables longer on average than spontaneous syllables;

- *prosody*: the experiments of [99, 91] show intonation and pause structure of utterances carry information useful for human distinction between spontaneous and planned utterances;

- *misarticulations*: [60] report misarticulation effects such as transposition of specific phonetic segments;

- *disfluencies*: [72] discusses the presence of restarts in conversational Switchboard data [54] eg. *"people will get, I mean, I've, - my brother lives where I work"*. Disfluencies in the spontaneous speech of adult normal speakers of American English are studied extensively in [153];

- *vocabulary and usage of words*: conversational speech contains more monosyllabic words. [44] reports one syllable tokens comprise 9% more of the spontaneous portion of Broadcast News than the planned speech portion. [14, 72] report greater use of pronouns in transcriptions of conversational speech as opposed to written newspaper text of the type commonly in dictated speech tasks. Words are reused more frequently in relaxed speech [44]. [72] also discusses conversational speech markers, known as *discourse markers* such as coordinating conjunctions used at the beginning of segments (`SO`, `WELL`, `AND`), acknowledgements, back-channel cues and *filled pauses* (`UM`, `UH-HUH` etc) and editing phrases (`I MEAN`, `YOU KNOW`), and reports that disfluencies and discourse markers comprise 10% of the Switchboard corpus [54];

- *sentence structure*: the syntactic structure of sentences becomes less hierarchical, following a more linear structure [44]. [72] finds differences in part-of-speech distributions found in the conversational Switchboard [54], dictated Wall Street Journal [128] and the mixed speaking style Broadcast News [57] corpora.

Recall that in the MULTIREG experiment of Section 3.1.1, the word transcriptions for the utterances to be transcribed were deliberately chosen to be *identical* across all speaking styles (excepting noises and special characters, and considering `UM` and `UH` identical). Therefore, differences due to *disfluencies*, *vocabulary and usage of words* and *sentence structure* should not contribute to the results. The remaining differences may be grouped under a general heading of *pronunciation* or *phonological change*: conversational speech differs from read speech in exhibiting greater phonological variability. We believe this specific type of variability may contribute significantly to the results of Section 3.1.1 and more generally to the poor performance of current transcription systems on conversational speech. But whilst this explanation is plausible, for the reasons discussed in the next section, the reader should note that the experiment above does *not* show conclusively that increased phonological variability leads to increased error rates since the MULTIREG data is not controlled for factors such as speech rate. Adopting this as our working hypothesis, however, the next section will discuss aspects of current recognizer design that may lead to an inadequate model of the increased phonological variability in conversational speech.

3.3 Potential Problems In Current Recognizer Design

Recall from Chapter 1 that sentence-level acoustic models are created using the beads-on-a-string procedure: acoustic models of sub-word units are concatenated according to a word-to-model-sequence mapping(s) in the pronunciation lexicon. This procedure is a potential source of weakness when speech becomes conversational, for one or both of the following reasons:

- *inadequate dictionaries*;
- *assumption of segmental nature of speech*.

These problems are discussed in more detail next.

3.3.1 Inadequate Dictionaries

Recognition dictionaries are typically perceived as inadequate for conversational speech recognition due to the limited number of pronunciations per word and because of the source of those pronunciations.

Limited Pronunciations A typical dictionary contains only a few pronunciations for each word. The PronLex-based test dictionary [84] used in the HTK-based recognizer at The Johns Hopkins University Summer Research Workshop on Large Vocabulary Speech Recognition has a single pronunciation for approximately 94% of the words in the test vocabulary, two pronunciations for more than 5% of the words and three or four pronunciations for the remaining (less than 0.5%) words. The LIMSI dictionary-based test dictionary [93, 64] used in the Cambridge University Engineering Department Switchboard recognizer for the 2000 evaluation has a single pronunciation for just under 91% of words in test set vocabulary, two pronunciations for just under 9% of words and three or more pronunciations for the remaining (less than 0.6%) words[5]. Most recognition dictionaries contain even fewer pronunciations per word.

Inappropriate Pronunciations The source of dictionary pronunciations varies, but they are rarely derived from conversational speech. The LIMSI dictionary [93] has been carefully hand-crafted for recognition performance but the tuning was based upon dictated speech. Most dictionaries use pronunciations based on dictated speech, but at an extreme, they may be created using only text-to-speech technology.

Given any pronunciation dictionary, it is assumed that the subword acoustic modelling scheme adequately represents all remaining variability. This is a strong assumption regardless of speaking style. Context-dependent modelling does acknowledge the influence of context on the realization of sounds and mixture of Gaussian output distributions in HMMs can capture variability in segment realizations. However, it can be argued that neither technique is an efficient model of these types of pronunciation change. Furthermore, acoustic models based on phone-level HMMs without skip transitions[6] are unlikely to be an adequate model of phonetic and lexical deletions. Use of limited dictionaries and reliance on the acoustic models to mop up remaining variability proved sufficient for dictated speech, but may not be adequate for capturing the increased range of pronunciations per lexical type that occur in conversational speech. (More discussion and empirical evidence related to this issue may be found in [81, 123, 173]). The problems that might be expected to arise through working with small sets of word pronunciations that are not representative of acoustic realizations of those words would be twofold: direct recognition errors through incorrect pronunciations and broad variance models resulting from a training procedure in which each subword model is potentially trained on data from other subword classes.

[5] Variants with the option of following silences were not counted as distinct pronunciations.

[6] Skip transitions are rarely used in state-of-the-art systems, having been found to degrade performance.

The potential inadequacy of the lexicon motivates *explicit* pronunciation modelling schemes, which attempt to modify the pronunciation dictionary to contain one or more pronunciations per word which are more representative of pronunciations in the speech style to be modelled. A survey of techniques is given in the next chapter and one specific framework extended to conversational speech in Chapter 5.

There is evidence to support research into explicit pronunciation modelling. The papers [148, 149] use *cheating experiments*[7] to investigate the utility of explicit pronunciation modelling schemes. A set of test pronunciations is obtained by aligning a phone-level test set transcription with the reference transcription. These pronunciations are close to optimal for each token in a maximum likelihood (ML) sense, although they may differ from transcriptions by linguists. Adding all pronunciations into a *static* recognition dictionary fixed throughout recognition reduced WER from 47% to 38%; a *dynamic* recognition dictionary adjusted to contain word pronunciations appropriate for each utterance reduced WER to 27%.

3.3.2 Inadequate Modelling of Relative Timing Effects

Explicit pronunciation modelling techniques attempt to improve speech transcription whilst continuing to use the beads-on-a-string scheme for forming sentence-level acoustic models. However, there are reasons to object to the fundamental assumption made by this scheme: namely, the assumption that speech can be rigidly segmented into a linear sequence of (usually phone-like) segments. Speech scientists, linguists and engineers agree that the notion of a phoneme or speech segment is not a realistic one eg. [69, 100, 34, 83]. It takes no account of basic speech production mechanisms. Speech is produced by loosely-coupled articulators, and speech production studies show the amplitude and phase between these gestures varies with changes in speaking rate, manner and style eg. [167]. These changes in relative timing can have extreme effects on the resulting acoustic signal: it often appears that there has been colouring and merging of the underlying 'segments' or even 'segment-like' insertions due to interaction between articulatory gestures both within and across segment boundaries. Examples of these effects include (1) *feature spreading* in coalescence, eg. `CAN'T /k ae n t/` → `[ k ae_n t ]` where nasality from deleted segment `/n/` colours the neighbouring vowel, and (2) asynchronous articulation errors causing stop insertions eg. `WARMTH /w ao m th/` → `[ w ao m p th ]`.

Phoneme-based schemes were adequate for dictated speech recognition, in which the amplitude and phase relations between gestures are fairly consistent. But as speech becomes more conversational, relative timing effects become more significant eg. [167]. Whilst simple to explain at the articulatory or phonological level, this type of variability is difficult to capture within a segment-based acoustic modelling scheme. We believe these effects contribute to the poor performance of current systems when transcribing spectral representations of conversational speech.

Approaches attempting to better model relative timing effects are mostly *implicit* pronunciation modelling schemes operating at the level of the acoustic model and (or) the beads-on-a-string procedure. One approach introduces more flexible state-level parameter sharing schemes, perhaps incorporating more knowledge of phonology or measures of speaking rate and style eg. [62, 63, 123, 41, 148]. A more speculative direction of research has investigated schemes for modelling intermediate articulatory or phonetic representations of speech, which may be a simpler domain in which to model the phonological effects in fluent speech, and several authors report research into extracting such representations automatically from the speech signal eg. [87, 82, 158]. Rather than

[7] *Cheating Experiments* utilize prior knowledge of the test set properties when producing a transcription.

model speech as a linear sequence of segments, authors advocating this type of approach to speech recognition attempt to model speech as a structured arrangement of phonetic or articulatory features between which there may be some degree of variation in the relative timing of phonetic events. Thus, for example, when nasality from nasal phoneme /m/ partially but not completely colours a neighbouring vowel /ae/, this would be modelled by asynchrony in the feature changes between the combination for /m/ and for /ae/. However, whilst there has been considerable work on schemes for extracting appropriate intermediate representations of speech and much discussion of the desirability of this type of approach, there are rather fewer papers considering schemes for incorporating these ideas within the statistical framework for speech recognition. This will be the focus of the later part of this thesis. The problem of modelling articulatory or phonological feature streams whilst allowing some but not unlimited asynchrony between them can be considered as the problem of modelling several, loosely-coupled time series. Chapter 6 reviews conventional speech models that have or could be applied to modelling loosely-coupled times series, particularly with respect to the degree of asynchrony which these models allow between the different times series. Chapters 7 and 8 then present *Loosely-coupled* or *Factorial HMMs*, a more general family of models that are potentially applicable to this type of modelling problem, and develop the theory and algorithms associated with two specific cases.

There is evidence to support research into the acoustic modelling level in addition to improving the lexicon. [105] suggests improved acoustic modelling would be useful, perhaps in combination with improved pronunciation dictionaries. The authors use simulated data to investigate the gains which might be possible through an improved pronunciation dictionary. When simulated test data matches the acoustic modelling assumptions, then performance can be improved by extending a static pronunciation dictionary to cover all linguistic variants occurring in the test set. For speech data, for which our current acoustic modelling assumptions are not necessarily correct, additional variants added to the dictionary (even those in a linguistic transcription of the test set) degraded performance by increasing confusability. This result is explained in terms of the mismatch between model assumptions and speech data, and the broad variance of acoustic models trained using the poor phonemic transcription that results from an inadequate dictionary. In a separate study, [59] performs a diagnostic evaluation of recognizers for the Switchboard task and also concludes that improving the acoustic models used for phonetic classification, as well as the pronunciation dictionaries, would benefit transcription performance.

4

Techniques for Explicit Pronunciation Modelling

The previous chapter discussed the difficulties that pronunciation variability can create for conventional speech transcription systems. This chapter begins by describing the levels at which previous research has addressed the pronunciation modelling problem and classifies schemes at different levels as being either *explicit* or *implicit*. It then surveys techniques proposed for *explicit* pronunciation modelling in more detail. This discussion provides background for the next chapter, which investigates the suitability of an explicit pronunciation modelling scheme for the conversational speech modelling task.

4.1 Formation of Acoustic Probabilities

Figure 4.1 illustrates one view of the formation of acoustic probabilities $p(\mathbf{O}|\mathbf{W})$. The acoustic model is formed using a hierarchical procedure with the following levels:

- word sequence $\mathbf{W} = w_1, \ldots, w_{N_w}$ is mapped to baseform sequence $\mathbf{B} = b_1, \ldots, b_{N_b}$;
- baseform sequence $\mathbf{B}$ is mapped to a logical model index sequence $\mathbf{L} = l_1, \ldots, l_{N_l}$;
- logical model sequence $\mathbf{L}$ is mapped to actual model indices $\mathbf{M} = m_1, \ldots, m_{N_m}$.

and the acoustic probability $p(\mathbf{O}|\mathbf{W})$ is obtained as:

$$\begin{aligned} p(\mathbf{O}|\mathbf{W}) &= \sum_{\mathbf{M},\mathbf{L},\mathbf{B}} p(\mathbf{O},\mathbf{M},\mathbf{L},\mathbf{B}|\mathbf{W}) \\ &\approx \sum_{\mathbf{M}} p(\mathbf{O}|\mathbf{M})\{\sum_{\mathbf{L}} P(\mathbf{M}|\mathbf{L})(\sum_{\mathbf{B}} P(\mathbf{L}|\mathbf{B})P(\mathbf{B}|\mathbf{W}))\} \end{aligned}$$

where the second line follows by assuming each level is conditionally independent of all higher levels except the immediately preceding level. This presentation assumes the model is *static*, meaning none of these distributions varies directly with properties of the speaker or the acoustics. *Dynamic* approaches introduce an additional conditioning variable $\mathbf{X}$ to represent such properties. The decoding criterion for the recognizer becomes $p(\mathbf{W}|\mathbf{O},\mathbf{X})$ and the acoustic component models $p(\mathbf{O}|\mathbf{W},\mathbf{X})$ in some fashion. Schemes for doing so are discussed in [124, 123].

The conventional approach to acoustic model construction fits into this framework as follows. Firstly a *pronunciation dictionary* or *lexicon* maps each word w_i in $\mathbf{W}$ to one or more sequences of baseform units $B(w_i)$ which are concatenated to yield one or more sequences $\mathbf{B}$. Baseform units are typically phoneme-like units. Probability $P(\mathbf{B}|\mathbf{W})$ is typically ignored; where used, it is generally formed by assuming that each sequence

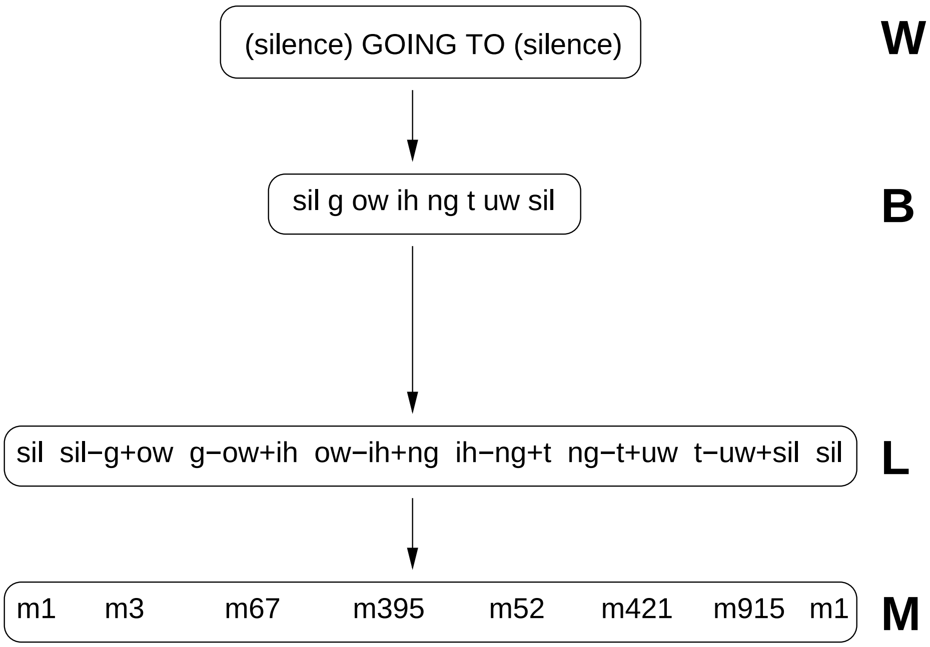

(silence) GOING TO (silence)
W
sil g ow ih ng t uw sil
B
sil sil−g+ow g−ow+ih ow−ih+ng ih−ng+t ng−t+uw t−uw+sil sil
L
m1 m3 m67 m395 m52 m421 m915 m1
M

Figure 4.1 *Formation of the Acoustic Model*

$B(w_i)$ is conditionally independent of all words except the corresponding word w_i. Thus $P(\mathbf{B}|\mathbf{W}) = \prod_{i=1}^{N_w} P(B(w_i)|w_i)$ and the probabilities $P(B(w_i)|w_i)$ are estimated using relative frequency in a forced alignment of the training set eg. [64]. Secondly, a *deterministic* mapping converts $\mathbf{B}$ to a logical model sequence $\mathbf{L}$: typically, each phonemic baseform b_i in some left and right phonemic context is mapped to a logical *triphone* model l_i. There is rarely sufficient data to estimate a distinct HMM for each possible l_i, so a third stage maps $\mathbf{L}$ to a sequence $\mathbf{M}$ specifying the models actually used to calculate acoustic probabilities $p(\mathbf{O}|\mathbf{M})$. This mapping is typically deterministic; it is often performed by a *decision-tree* eg. [122].

The problem of pronunciation variability has been attacked at various levels of this hierarchy. We characterize those modifying the conventional scheme at levels between $\mathbf{B}$ and $\mathbf{L}$ as *explicit* pronunciation modelling techniques and those which affect levels below $\mathbf{L}$ as more *implicit* techniques, where we intend the latter to encompass work investigating alternative families of stochastic models for calculating $p(\mathbf{O}|\mathbf{M})$[1].

The next chapter will investigate an explicit technique, so the remainder of this chapter will present a brief survey of work in this area. Later chapters focus on a more implicit technique. There has been considerably more work of this type, not all of it motivated directly by pronunciation effects, and it is not appropriate to attempt to survey all such techniques within this thesis. The interested reader might start with [36, 62, 63, 148] as recent successful examples of implicit techniques directly motivated by the need to improve pronunciation modelling for conversational speech tasks.

[1] There are a few pronunciation modelling techniques which cannot be classified as explicit or implicit under this definition. For example, some techniques try to construct a mapping between $\mathbf{W}$ and $\mathbf{M}$, as exemplified by [6].

4.2 Explicit Pronunciation Modelling Techniques

This section gives a short overview of approaches to explicit pronunciation modelling. A more comprehensive survey may be found in [160].

4.2.1 Word-Level Techniques

Word-level techniques extend the mapping $\mathbf{W}$ to $\mathbf{B}$ by adding new word pronunciations to the lexicon. New pronunciations can be manually or automatically generated. Automatic schemes typically align phone-level training set transcriptions (produced manually or through phone recognition) with word-level transcriptions; new pronunciations are then extracted from these alignments. Examples include [154].

Word-level schemes have two problems. Firstly, new word-specific pronunciations do not generalize easily to words unseen in the training set. Secondly, word-specific pronunciations may inadequately represent coarticulation effects between words. Solutions have been proposed. The unseen word problem is addressed by [178]. A heuristic solution for better modelling coarticulation between words introduces compound word sequences as new lexical items in the dictionary, with their pronunciations capturing compound-specific coarticulation effects, as in `GOING+TO /g ax n ax/` eg. [42, 120].

4.2.2 Phoneme-Level Techniques

Until recently, most work in pronunciation modelling operated between levels $\mathbf{B}$ and $\mathbf{M}$. There were two reasons for this. The first was based upon linguistic arguments. Phonologists, who study sound structure and patterns within languages, use two levels of representation to distinguish between systematic and more idiosyncratic aspects of pronunciation[2]. The abstract *phonological* representation is usually given in terms of *phonemes*, where each phoneme is the smallest unit of speech that distinguishes one word from another. For example, `/p/` is phonemic, distinguishing `TAP /t ae p/` from `TAG /t ae g/`. Phonemes may have more than one acoustic realization, from amongst the set of *phones*, which do not cause differences in meaning. For example, the realizations of `/p/` in `PAT /p ae t/` and `SPAT /s p ae t/` usually differ phonetically but the contrast never distinguishes words in English. In linear phonology, the *phonetic* or *surface* representation is derived from the phonemic representation through *phonological rules*, which predict the allophonic realizations of phonemes in context. Speech recognition researchers observed that most dictionaries in early recognition systems were defined using phonemes. Since the allophones of phonemes may be acoustically quite different, they hypothesized that the modelling task might be simplified by modelling classes defined in terms of these allophonic *phonetic* or *surface* units. Therefore an extra level of indirection was introduced, $\mathbf{S} = s_1, \ldots, s_{N_s}$ (shown in Figure 4.2), which was intended to represent the allophonic realizations of a phonemic baseform sequence $\mathbf{B}$ as given in the dictionary. Thus:

$$\begin{aligned} p(\mathbf{O}|\mathbf{W}) &= \sum_{\mathbf{M},\mathbf{L},\mathbf{S},\mathbf{B}} p(\mathbf{O},\mathbf{M},\mathbf{L},\mathbf{S},\mathbf{B}|\mathbf{W}) \\ &\approx \sum_{\mathbf{M}} p(\mathbf{O}|\mathbf{M})\{\sum_{\mathbf{L}} P(\mathbf{M}|\mathbf{L})[\sum_{\mathbf{S}} P(\mathbf{L}|\mathbf{S})(\sum_{\mathbf{B}} P(\mathbf{S}|\mathbf{B})P(\mathbf{B}|\mathbf{W}))]\} \end{aligned}$$

[2] This discussion is a simplification and does not refer to recent phonological theories. There are various approaches to phonology, not all of which are linear or rooted in the concept of phonemes. The interested reader is referred to [56, 79, 143].

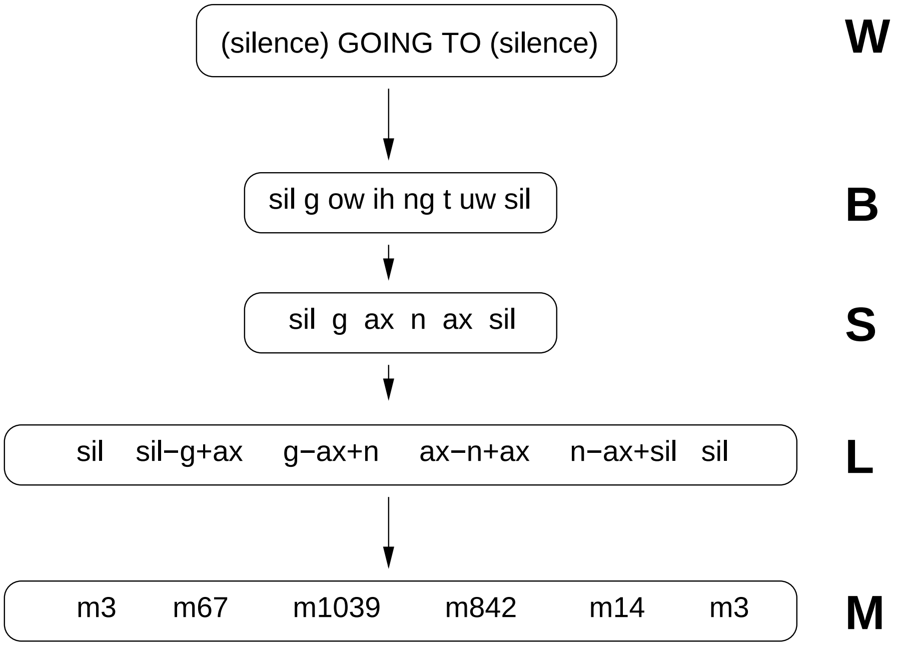


Figure 4.2 *Formation of the Acoustic Model via Surface Phones*

The mapping between $\mathbf{B}$ and $\mathbf{S}$ is analogous to linguistic phonological rules and can be constructed by hand or learnt automatically using a statistical classifier. Typically, these map between baseforms b_n in particular contexts and *surface* realizations of those phonemes s_n. This leads to the second motivation for pronunciation models at the level of phonemic variation: phoneme-level classifiers generalize to allow prediction of new pronunciations for words unseen in the training acoustics. The surface sequence(s) $\mathbf{S}$ predicted by the rules or classifiers are mapped deterministically to a sequence of logical models $\mathbf{L}$ (typically triphones).

Examples of this type of approach abound. A manually-generated rule set is investigated by [161]. Probabilities may be associated with rules to allow calculation of a probability $P(\mathbf{S}|\mathbf{W})$. Hand-crafted rule sets suffer from the limitation that hand-crafted rules may not cover all changes that occur in practice. Approaches using statistical classifiers are investigated by eg. [26, 23, 70, 44, 173]. The approach investigated in the next chapter, proposed by [141], is also in this category. These schemes construct a statistical classifier $T(b_i)$ between baseforms in context and surface realizations s_i (either in the form of manual transcriptions or output from a phone recognizer). The classifiers typically provide probabilities $P(s|T(b_i))$; thus the required probabilities may be formed as $P(\mathbf{S}|\mathbf{B}) \approx \prod_{n=1}^{N_b} P(s_n|T(b_n))$, under appropriate conditional independence assumptions. (Note that these conditional independence assumptions are not necessarily justifiable, as discussed in [20]).

4.3 Use of Explicit Pronunciation Models

Most work on the use of explicit pronunciation modelling techniques reports results for the incorporation of new pronunciations into the test lexicon, where improved pronunciations should reduce misrecognitions. However, an improved dictionary can also be incorporated into acoustic model training. Assuming the new pronunciations are representative of the speaking style, many researchers anticipate that a forced alignment [73, 180] of the training set using the new dictionary should create phonetic transcriptions of the training set more representative of the classes in the acoustics; models trained on these transcriptions should then have lower variances. Despite this anticipated improvement, few authors report results for this use of pronunciation models. Amongst those who do, [142] significantly improves dictated speech recognition by training acoustic models using phonetic transcriptions obtained using a decision-tree-generated dictionary, [154] presents inconclusive results and [42] claims improvements using a combination of techniques, but does not isolate the contribution from the incorporation of new pronunciations in acoustic model training.

5

Explicit Pronunciation Modelling Using Decision Trees

This chapter investigates a phoneme-level, explicit pronunciation modelling technique. It builds upon the framework presented in [140, 141], which was reported to improve performance on the TIMIT [92] dictated speech task. This chapter evaluates the approach using the large-vocabulary, conversational speech Switchboard (SWB) task [54]. Direct application of the original technique to generate new pronunciations for use in recognition does not improve performance, but with appropriate modifications the technique can lead to a small but statistically significant reduction in the error rate for the conversational task. As discussed in the previous chapter, many researchers have also argued that explicit pronunciation models might also be usefully incorporated into acoustic model training. This chapter investigates a variety of schemes for doing so and discusses some of the difficulties that arise.

Some of the tools and experiments reported in this chapter are due to the Pronunciation Modelling group at WS97 (The Johns Hopkins University Summer Research Workshop on Large Vocabulary Speech Recognition) in which the author was a participant, some were performed by Murat Saraclar as part of his doctoral thesis [148] and the remainder were performed by the author whilst funded by a post-WS97 follow-up research grant. Experiments not performed solely by the author will be clearly indicated in the text.

5.1 Basic Pronunciation Modelling Framework

The following framework proposed by [140, 141] makes the fundamental assumption that each realization of a word can be represented as an unambiguous sequence of phones and that linguists can transcribe this sequence accurately. The validity of the assumption is discussed in Section 5.3. A direct application of this approach to the Switchboard corpus involves the following basic steps, which are illustrated in Figure 5.1 and the example of Figure 5.2 (see also [139]):

1. **obtain canonical (phonemic baseform) transcription of training set**: a forced alignment procedure (see eg.[73, 180]) using a standard recognizer pronunciation dictionary will select amongst alternatives if they exist;

2. **obtain surface-form (phonetic) transcription of same training set**: linguists transcribed a four-hour portion of the Switchboard corpus using a broad phonetic symbol set for this purpose [58];

3. **align phonemic and phonetic transcriptions**: a dynamic programming procedure based on the [175] phonetic feature distances can be used for this purpose [139]. The resulting alignment determines the surface (or acoustic) realization of each phoneme in the canonical transcription;

4. **estimate a decision-tree pronunciation model**[1]: a decision tree is constructed to predict the surface or allophonic realization of each phoneme by asking questions about context, including neighbouring phonemes, lexical stress and syllable boundaries;

5. **perform recognition with this pronunciation model**: a language model or word lattice generated by an earlier recognition pass is expanded into a phoneme lattice using the standard pronunciation dictionary. The pronunciation model is then used to transduce the phoneme sequences in this network to yield a network of surface-form realizations weighted by their pronunciation probabilities. Figure 5.3 illustrates this expansion for a two word fragment. Recognition is performed using this surface phone-level network.

The next section outlines the basic experimental setup and then presents experimental results.

5.2 Experimental Results

5.2.1 Corpus

All experiments use the Switchboard corpus, which is a corpus of spontaneous telephone conversations between two individuals about loosely specified topics such as AIDS, gardening or health-care for the elderly [54]. A vocabulary of approximately 20000 words provides adequate coverage for the corpus. Acoustic model training uses 60 hours of speech (about 100000 utterances or a million words) selected from about 2000 conversations. There are 383 different speakers in the training corpus. A speaker-disjoint set of about 1.5 hours of speech (19 entire conversations, 2427 utterances, 18100 words) is set aside for testing ASR systems. The Switchboard Transcription Project [58] has produced manual phonetic transcriptions (fairly broad) for a four hour subset of the corpus: these *ICSI transcriptions* include a 3.5 hour subset (3600 utterances, 100000 phones) of the training set and a 0.5 hour subset (451 utterances, 18000 phones) of the test set.

5.2.2 Baseline Recognizer and Recognition Results

The baseline acoustic models are state clustered cross-word triphone HMMs having 6700 shared states, each with 12 Gaussian densities per state. The PronLex lexicon [84] is used in the baseline system: this has a single pronunciation for approximately 94% of the words in the test vocabulary, two pronunciations for just over 5% of the words and three or four pronunciations for the remaining (less than 0.5%) words. Bigram and trigram models trained on 2.2 million words of transcribed Switchboard conversations are used as language models.

The experiments below are based on word lattices generated using the baseline system with a bigram language model. The lattices are then used as a word graph to constrain a second recognition pass in which a trigram language model is used. Use of the lattice rescoring paradigm reduces time required for experimentation while allowing a search over a large set of likely word sequences. Note that the best possible WER obtainable from hypotheses in these word lattices is less than 10%, which is generally considered adequate for experiments within the lattice rescoring paradigm.

All acoustic model training and lattice rescoring uses HTK [180]; the AT&T Weighted Finite State Transducer tools are used to manipulate word and phone lattices [109].

[1] Other statistical mappings are also applicable, eg. neural networks.

5.2.3 Performance Measures

The effects of the pronunciation modelling technique are assessed using word error rate (WER) eg. [180]. State-of-the-art ASR systems at the time this research was performed (1997-98) achieved 30-35% WER on the Switchboard corpus; the baseline system described above has comparable performance. Phone error rate (PER) is also used as a measure of transcription accuracy. PER figures will be reported later on the 451 test set utterances and the 1800 of the 3600 training set utterances for which the correct phonetic transcription is available.

5.2.4 Comparison: New Pronunciations in Recognition

An experiment performed at WS97 by the pronunciation modelling group examined the effects of introducing pronunciations using the basic scheme of Section 5.1. The results showed that when only a small amount of phonetically labeled data is available in Step 2 (such as the ICSI transcriptions), the resulting WER (after Step 5) is *worse* (1.4% absolute) than the baseline result obtained using the canonical pronunciation dictionary [20, 139]. This is in contrast to the original results of [140, 141], in which a similar procedure applied to a dictated speech task improves performance. The most likely explanations for this effect are the mismatch between the acoustic models and the manually-generated transcriptions and the high degree of lexical confusion.

Researchers at WS97 hypothesized that increasing the amount and type of data available for tree-building in Step 2 [20], extending the original procedure with a new Step 6. Starting with the canonical transcription of the entire acoustic training set (instead of just the hand-labeled portion in Steps 1-2), the pronunciation model of Step 4 is used to create pronunciation networks representing possible phonetic realizations of each training utterance. The most likely phone-sequence through each network is chosen via Viterbi alignment using a set of *existing* acoustic models, giving a “refined” transcription of the entire training set. In the WS97 experiments, this yields around 60 hours of data to be used in pronunciation model building, as opposed to the four hours of data used in the initial experiments above.

Experiments performed at WS97 showed that replacing the small corpus of Step 2 with this larger corpus and then repeating Steps 3-5 gives a small but statistically significant (39.4% → 38.9%, $\sim$ 0.5% absolute) WER reduction [20, 139]. Further iteration of these retranscription-reestimation steps does not yield further improvements [148].

5.2.5 Comparison: New Pronunciations in Acoustic Model Training

The results above show that the additional pronunciations can improve recognition performance. A logical progression incorporates the new pronunciations into acoustic model training: if a training transcription more accurately represents the classes in the acoustics, sharper acoustic models should result from training. With the exception of Section 5.2.5.4, these experiments were performed solely by the author.

Recall that the phonetic training set transcription of the new Step 6 (Section 5.2.4) made use of the new pronunciations. This transcription was compared with the ICSI manual phonetic transcriptions for a subset of 1800 sentences (40000 phones) to determine whether it did better represent the classes in the acoustics. The comparison metric is the string edit distance between the two phonetic transcriptions for each utterance, and does not incorporate information about the time alignment of phonetic segments. The number of errors in the automatic transcriptions is the total number of insertions, deletions and substitutions.

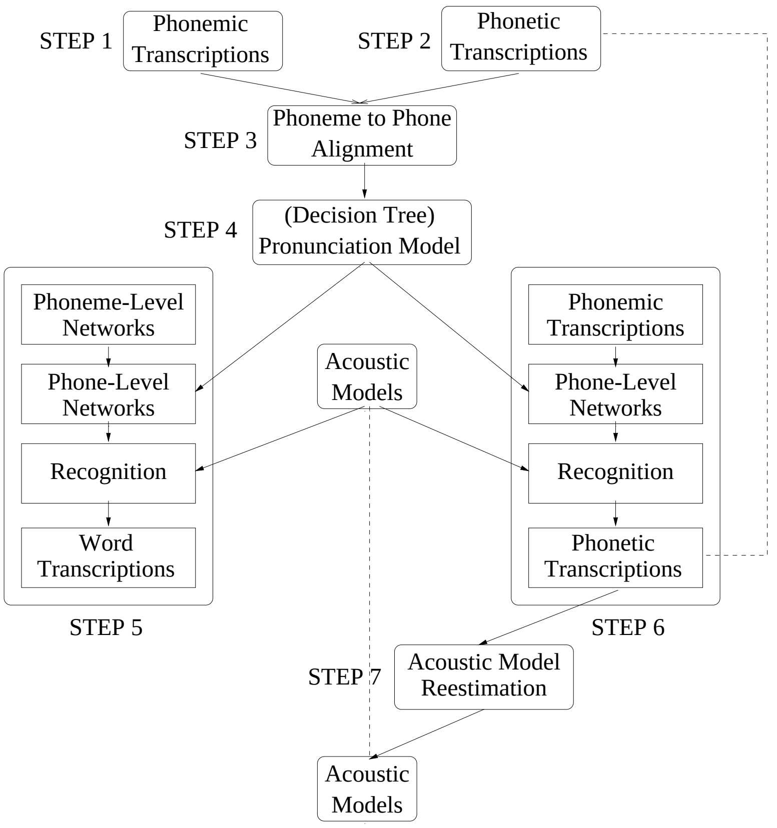


Figure 5.1 *Pronunciation Modeling Framework*

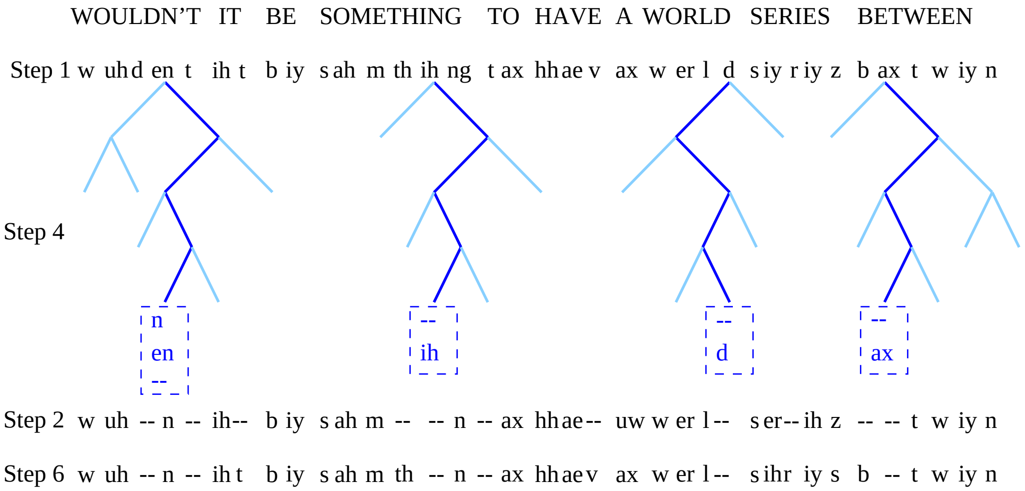
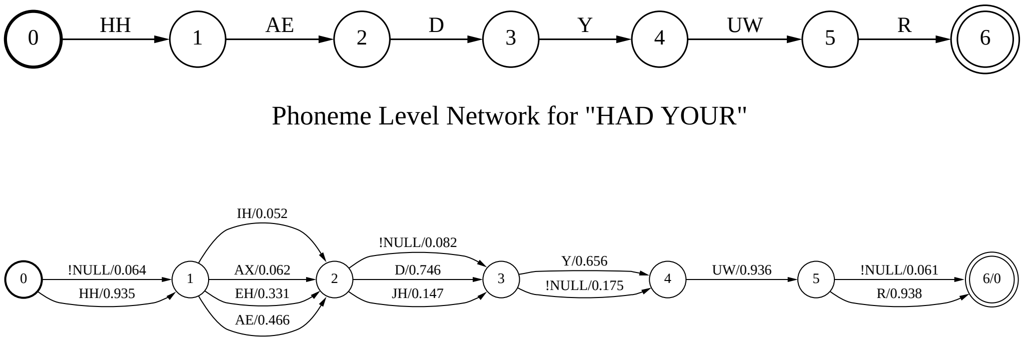

WOULDN'T IT BE SOMETHING TO HAVE A WORLD SERIES BETWEEN
Step 1 w uhd en t ih t b iy s ah m th ih ng t ax hh ae v ax w er l d s iy r iy z b ax t w iy n
Step 4
n
en
--
--
ih
--
d
--
ax
Step 2 w uh -- n -- ih-- b iy s ah m -- -- n -- ax hh ae-- uw w er l-- s er-- ih z -- -- t w iy n
Step 6 w uh -- n -- ih t b iy s ah m th -- n -- ax hh ae v ax w er l-- s ihr iy s b -- t w iy n

Figure 5.2 *Phonemic (Step 1), Phonetic (Step 2) and Automatic (Step 6) Transcriptions and Estimating a Decision Tree Pronunciation Model (Step 4)*

Transcriptions	PER *vs* manual labels	PER *vs* baseforms
Dictionary Baseforms	28.3%	0%
Automatic (Step 6)	26.1%	4.1%

Table 5.1 *Improved Training Transcriptions for Acoustic Model Estimation*



Phoneme Level Network for "HAD YOUR"

Phone Level Network for "HAD YOUR"

Figure 5.3 *Phoneme and Phone Level Networks for "HAD YOUR" (`!NULL` indicates a deletion) (Reproduced from the WS97 pronunciation modeling group final presentation by Michael Riley)*

Adaptation Method	PER *vs* manual labels	PER *vs* baseforms
None	26.1%	4.1%
VTLN	26.0%	4.2%
MLLR	26.0%	4.0%

Table 5.2 *Failed Attempts to Further Improve Training Transcriptions via Adaptation*

The results of this comparison, presented in Table 5.1, show the new (Step 6) phone transcriptions agrees more closely with the manual transcriptions than the baseform transcriptions. The new transcriptions were therefore used to estimate a *new* set of acoustic models, in an additional Step 7. The resulting, *new* acoustic models are then used in a repeat of the Step 6 retranscription, replacing the *original* acoustic models used in Section 5.2.4. The resulting phonetic transcription is then used for a pronunciation model estimation and recognition (repeating Steps 3-5). The repeat of Steps 6, 3-4 after the new Step 7 is done to ensure that the final pronunciation model in Step 4 is matched to the *new* acoustic models before performing the recognition test in Step 5.

An initial experiment showed that training on the Step 6 transcriptions in this fashion gives no improvement in recognition performance (38.9% WER) over the acoustic models trained on canonical baseforms. One explanation might be that the Step 6 transcriptions, whilst more accurate than the baseform transcriptions (Table 5.1), are still not sufficiently high quality to improve recognition accuracy. Four procedures were investigated with the goal of improving the phonetic accuracy of the training transcriptions: speaker adaptation, simpler models, cross transcription and bootstrapping. These are described next.

5.2.5.1 Speaker and Channel Adaptation

Initial experiments by the author investigated whether standard speaker and channel adaptation techniques can be used to adjust the acoustic models used in Step 6 to obtain more accurate phonetic transcriptions. Vocal Tract Length Normalization (VTLN) and Maximum Likelihood Linear Regression (MLLR), both described in Section 2.6, are used to adjust the acoustic models before performing the retranscription in Step 6.

The use of adaptation techniques leads to little change in transcription accuracy relative to the hand-labeled transcriptions (Table 5.2). It also results in little change in transcription content as evidenced by the comparison of the three automatic transcription techniques in Table 5.2. The new transcriptions remain fairly close to the original base-

Acoustic Model Used in Step 6	PER *vs* manual labels	PER *vs* baseforms
12-Gaussian triphone models	26.1%	4.1%
8-Gaussian triphone models	25.7%	5.0%
Single Gaussian triphone models	25.5%	9.3%

Table 5.3 *Simpler Acoustic Models Improve Phonetic Transcription*

Transcription Technique	PER *vs* manual labels	PER *vs* baseforms
Self Transcription	25.7%	5.0%
Cross Transcription	25.3%	8.1%

Table 5.4 *Jack-knifing Improves Phonetic Transcription (8-Gaussian triphone models)*

form transcriptions both before and after adaptation.

The results suggest the original hypothesis – that the phone transcription accuracy in Step 6 can be substantially improved within this framework – is incorrect; it appears instead that the highly-parameterized acoustic models used here are well-tuned to match the acoustics to the PronLex baseforms on which they are trained so that only drastic mispronunciations can be discovered when using these models in the retranscription stage. Adaptation based on the training transcriptions simply reinforces the problem.

5.2.5.2 Simpler Acoustic Models

It seems likely that the accuracy of phonetic transcriptions is being limited by the ability of the highly parameterized acoustic models to match the realized acoustics to the canonical baseforms. Experiments by the author investigated performing Step 6 using simpler acoustic models. In particular, the 12-Gaussian mixtures in the HMM state output densities of the baseline system are replaced during Step 6 by 8-Gaussian mixtures or single Gaussian densities, whilst retaining the same state-tying structure.

The results in Table 5.3 shows that simpler acoustic models do slightly improve the phonetic accuracy of the training transcriptions, but this does not translate into changes in recognition performance.

5.2.5.3 Cross Transcription

The automatic transcription procedure of Step 6 may be hampered by the fact that the acoustic models used for transcription were trained on the *same* acoustics together with the canonical (baseform) transcription. A natural solution is to transcribe the training set using models trained on different data[2]. An experiment by the author partitioned the 60-hour Switchboard training set into two speaker disjoint gender-balanced 30 hour subsets and model sets trained on one half are used to phonetically transcribe the acoustics for the other half of the data (as in Step 6). The resulting transcriptions are then used to train a set of acoustic models (as in Step 7). Steps 6, 3, 4 and 5 are then carried out to estimate and test a pronunciation model.

Phone recognition accuracy relative to the hand-labeled transcriptions improves only slightly when using the cross-transcription method as shown in Table 5.4, but the resulting transcriptions do deviate even more from the baseforms. Despite this, these “refined” transcriptions do not lead to significant changes in recognition performance.

[2] This experiment was suggested by Bill Byrne of the Center for Language and Speech Processing, The Johns Hopkins University.

Transcription Type	Models	PER *vs* manual labels	PER *vs* baseforms
Dictionary Baseforms	—	33.6%	0%
Automatic (Step 6)	Standard	31.4%	3.9%
Automatic (Step 6)	ICSI-models	26.6%	20.7%

Table 5.5 *Using Hand Labeled Data to Train Acoustic Models for Improved Phone Transcription given the Word Transcription (451-utterance subset of the test set)*

5.2.5.4 Bootstrapping

The results above suggest it is difficult to generate more accurate *automatic* phonetic transcriptions using acoustic models originally trained on canonical baseforms. This motivated Saraclar to investigate a more extreme procedure for improving the quality of training set transcriptions [149]. This experiment was not performed by the author. One way to obtain more accurate phonetic transcriptions of the entire acoustic training corpus (Step 6) is to use acoustic models which are trained directly on only the hand-labeled portion of the training corpus (ICSI portion of the corpus).

Only a small portion (3.5 hours) of the acoustic training data has been transcribed at the phone level by human labelers. Due to this limitation, a new set of context-independent phone models (henceforth called *ICSI-models*) were estimated using the hand-labeled portion of the training set. The limited amount of hand-labeled data has two unintended benefits. For one, most of the (60 hours of) speech to be transcribed is not used in model training, yielding some of the benefits of cross-transcription seen in Section 5.2.5.3. For another, the use of monophone models instead of triphones is another step in the direction of simpler acoustic models (for phonetic transcription) described in Section 5.2.5.2.

The automatic transcription of Step 6 is performed next, replacing the *existing* acoustic models with the *ICSI-models* described above. This results in considerably more accurate phonetic training transcription (see Table 5.5). Step 7, training acoustic models on the entire training set, is performed next. The resulting models are named *ICSI-bootstrap models*. This is followed by the usual procedure (Steps 6, 3, 4, and 5) of estimating and testing a new pronunciation model appropriate for these acoustic models.

The following results show that phone transcription accuracy is improved by models trained on hand labels. Since these models are bootstrapped from the phonetically labeled training utterances on which the results of Tables 5.1-5.3 are reported, it is inappropriate to compare transcription accuracy on that set. Instead, results are reported on a 451-utterance subset of our test set, which also has phonetic labels, to compare the transcription accuracy of the *ICSI-models* with models trained on canonical pronunciations. The task is the same as Step 6: choose the best phone-sequence given the word transcription and a pronunciation model. The results of Table 5.5 for the *ICSI-models* indicate that the transcriptions on which the *ICSI-bootstrap* models are trained are much more accurate than the baseforms or the transcriptions used in preceding sections.

A further experiment used the *ICSI-bootstrap* models for recognition. While the standard acoustic models (together with a pronunciation model) have a WER of 38.9%, the WER of the *ICSI-bootstrap* models is 41.3%. The performance of the model on the 451 phonetically labeled utterances in the test data was analyzed to better understand the cause of this degradation. In addition to the WER performance the phone error rate is measured against the hand transcriptions.

Table 5.6 shows the *ICSI-bootstrap* models improve phone accuracy by 4.5% on this subset of the test set, although the WER is worse by 1.4%.

Pronunciation	Acoustic Model			
Model Used	Standard		ICSI-bootstrap	
in Step 5 (Test)	PER	WER	PER	WER
None (Dictionary)	49.1%	49.1%	49.5%	58.9%
Tree Pron. Model	47.7%	48.7%	43.2%	50.1%

Table 5.6 *Comparison of Word and Phone Error Rates of 1-best Recognition Hypothesis Under Different Acoustic and Pronunciation Models*

5.3 Conclusions

It is clear from the results above that there is considerable deviation from canonical, phonemic baseform pronunciations in spontaneous speech. However, the results also demonstrate limitations of explicit pronunciation modelling techniques that operate by learning a mapping from phonemic to phone-level representations of words. There are two main difficulties.

The gains achieved through the recognition-time only introduction of the surface phone-level pronunciations are limited. Saraclar [148] presents a detailed quantitative study which demonstrates why this is so. The study finds most deviation in pronunciation is not large enough to be represented well using a model which operates at the symbolic level. This is because the allophonic realizations of phonemes are not simply one of some restricted set of surface phones. Rather, their acoustic realizations lie on a continuum between the acoustic properties of the original phoneme and those of the surface phone. The pronunciation model used in this chapter does not take account of the residual influence of the original baseform on the surface realization. Although the surface phone-level pronunciations are slightly more accurate than the phonemic pronunciations, failure to model the effects of the underlying baseform on the acoustic realization means there are no great improvements in the acoustic model and therefore little change in recognition performance. The evidence supporting these claims is to be found in Saraclar [148].

There are also fundamental difficulties with introducing new surface phone-level pronunciations into acoustic model training, as exemplified by the previous section. A variety of techniques were investigated for producing quantifiably more accurate training transcriptions, but use of these transcriptions in acoustic model training did not lead to improved recognition performance. Analysis shows that the new transcriptions can lead to improved modelling of the acoustics of surface phones, but this does not translate into improved word recognition. The author and colleagues at The Johns Hopkins University attribute this to increased lexical confusion: many of the word pronunciations predicted by the decision-tree pronunciation model overlap with pronunciations of other words, increasing the difficulty of mapping back to word strings from the more accurate surface phone transcriptions.

The problems associated with use of the explicit pronunciation modelling technique discussed in this chapter arise because it attempts to describe pronunciation change at the level of segments. Pronunciation change is assumed to involve complete changes in segment identity, but it has been shown quantitatively by eg. [148] that this is not the case. The acoustic realizations of phonemes depend upon both the underlying phoneme and its phonemic context, but the changes are often only partial and do not represent substitution or deletion of a different phonetic unit. The remainder of this thesis will focus on an approach that might be more appropriate for modelling the influence of phonemic context and the subsegmental nature of pronunciation change.

6

Techniques for Modelling Asynchrony

The previous chapter showed that only limited gains have been achieved through incorporating explicit pronunciation modelling techniques into conventional speech recognition systems. A framework assuming that words are pronounced as a sequence of concatenated phonemic segments which may be realized as one of a finite set of phonetic segments dependent upon their context is not an adequate model of pronunciation change. Pronunciation change involves changes at levels below the segment as well as at the segment level and the effects of context often lead only to partial and variable degrees of colouring of the realization of the underlying phoneme.

Many researchers and linguists hypothesize that pronunciation variability in conversational speech might be more easily modelled if more knowledge of speech production and linguistic theory was used in the recognizer: specifically, knowledge of relative timing effects as discussed in Chapter 3 eg. [68, 69, 100, 34, 83, 123][1]. Subsegmental variation and differing degrees of colouring of phoneme realizations may be simpler to explain in terms of varying degrees of asynchrony between articulatory gestures or between phonological features. The arguments are similar to those in favour of *non-linear* or *autosegmental* rather than linear phonological models in linguistics [56].

The remainder of this thesis considers a model family that might provide the ability to better model relative timing effects within a statistical speech recognition system. It is hoped such an approach will ultimately lead to better modelling of the partial and subsegmental nature of pronunciation change whilst avoiding the confusability issues associated with segment-level schemes such as that in the previous chapter. Ideally, rather than model speech as a linear sequence of segments, such an approach would model speech as a structured arrangement of features between which there may be some variation in the relative timing of feature changes. To illustrate, one possible implementation of this approach uses a phonological, rather than spectral, feature representation of speech; when nasality from nasal phoneme `/m/` partially but not completely colours a neighbouring vowel `/ae/`, this would be modelled by timing differences in the feature changes between the combination for `/m/` and for `/ae/`. Pronunciation variability is then ascribed to asynchrony between feature changes. However, whilst changes between feature tiers may not be synchronous, there is still some dependence between the points at which they change: the tiers are *loosely-coupled*.

One way to incorporate these ideas into an ASR system is to use a two-stage approach to ASR, in which (i) the acoustic signal is mapped into an intermediate representation comprising a number of potentially asynchronous feature streams, likely to be related to phonologically-motivated distinctive features or articulatory parameters (although alternatives such as cepstra derived from sub-frequency bands or other automatically extracted feature sets are not incompatible), and then (ii) the intermediate representation is modelled in some statistical fashion, using a model somehow incorporating the notion

[1] More extensive discussion related to the potential utility of speech production knowledge in recognition systems can be found in eg. [110, 144, 147].

of loose coupling between feature streams. This broad framework has been advocated by many authors eg. [69, 85, 83, 108], and papers proposing schemes for extracting intermediate representations of speech are ubiquitous. Most are phonological or articulatory representations [151, 83, 156, 39, 69, 2, 158, 86, 30, 119, 45]; an alternative is a frequency subband cepstra representation of speech, since there is reportedly asynchrony between changes in frequency subbands [108]. However, despite the wide variety of schemes for feature extraction, there has been little theoretical or practical work investigating the modelling problem although some recent papers [34, 86, 156, 45, 157, 138] take initial steps in this direction. The latter part of this thesis will focus specifically on this issue. Note that, whereas the work in the previous chapter represented an attempt to enhance a mature technology on cutting-edge tasks, the more implicit approach to pronunciation modelling that might result from the approach investigated in the remainder of the thesis represents new territory. The remainder of this dissertation will therefore focus on basic algorithmic issues, since these must be resolved before a feature-based approach to large vocabulary speech recognition can be investigated in a principled fashion.

The remainder of this chapter reviews conventional speech models that have been used for modelling potentially asynchronous, multiple time series data. The next chapter shows that, with the exception of the State-Coupled Model of Section 6.1.4, all are special cases of the general family of *loosely-coupled* or *factorial* HMMs (FHMMs). It will then go on to consider instances of the FHMM family which are more general than the models described here and that might be usefully applied to the task of modelling loosely-coupled time series data. The reader familiar with directed acyclic graphical models will observe that all models discussed in this and later chapters are also instances of this very general model family[2]. However, restricting our investigations to the subset of factorial HMMs has some advantages. Firstly, the space of directed acyclic graphical models is vast and even with the sub-family of factorial HMMs there are still a large number of possibilities to be explored. Secondly, the close links between FHMMs and conventional HMMs means that many of the issues related to incorporating general graphical models into the statistical framework for speech recognition do not arise. (For a discussion of these issues, see [184]).

6.1 Modelling Multiple Loosely-Coupled Time Series

The data to be modelled comprises K loosely-coupled, discrete-time series, where observations in each time series (or *stream*) k, denoted $o_1^k, o_2^k, \ldots, o_T^k$, are produced on the same time-scale and may be scalars or vectors. For speech recognition, each time series might correspond to a position trace for a particular articulator or to a phonological feature such as voicing or perhaps to cepstra derived from a frequency subband; for audio-visual speech recognition, one time series might correspond to acoustic features and another to features derived from visual features (see, for example, [17, 114]). The vector representing the concatenation of observations from the different streams is denoted $\mathbf{O}_t = (o_t^1, \ldots, o_t^K)$.

This section presents a short survey of models that have been applied to this type of parallel time series data in a speech recognition context. The survey is not intended as a general review of techniques for modelling stochastic processes; the interested reader might start with [162]. It is also not a survey of techniques for incorporating phonological or articulatory information into recognizer design. The discussion further focuses only on model assumptions and not the issues that must be addressed when incorporat-

[2] The paper by [155] also demonstrates that the standard HMM algorithms are special cases of more general algorithms for directed acyclic graphical models.

ing such models into large vocabulary speech recognition systems; more discussion can be found in the original papers.

Each model will be illustrated as a *Directed Acyclic Graphical Model*, for the case $K = 2$, to facilitate comparison of the various conditional independence assumptions made. A brief introduction to this representation is included in Appendix J. Shaded nodes in the illustrations correspond to observed variables; unshaded nodes correspond to hidden or latent variables. The set of latent variables at time t is denoted $\mathbf{S}_t$ and a specific latent variable within that set (henceforth referred to as a hidden *state*) by s_t^k. In all the models discussed here, hidden variables are discrete; observations could be discrete or continuous. Conditional independence assumptions will also be stated for the case $K = 2$: the shorthand $X \perp Y \| Z$ indicates the sets of random variables X and Y are conditionally independent given the set Z.

6.1.1 Hidden Markov Models

The *Hidden Markov Model* (HMM) is a widely-used stochastic model for single discrete time series eg. [10, 131, 130]. It could be used to model data from parallel, loosely-coupled time series if these were first concatenated into a single time series with observations $\mathbf{O}_t$ (defined above), although this may not lead to a parsimonious representation of the data. At each time t there is a single hidden state, denoted s_t. Figure 6.1 shows HMM structure. The conditional independence assumptions are:

- $\mathbf{O}_t \perp \{\mathbf{O}_1, \ldots, \mathbf{O}_{t-1}, \mathbf{O}_{t+1}, \ldots, \mathbf{O}_T, s_1, \ldots, s_{t-1}, s_{t+1}, \ldots, s_T\} \| s_t$
- $s_t \perp \{\mathbf{O}_1, \ldots, \mathbf{O}_{t-1}, s_1, \ldots, s_{t-2}\} \| s_{t-1}$

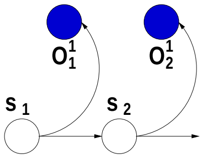


Figure 6.1 *Hidden Markov Model*

6.1.2 HTK Multiple Stream Model

The *HTK multiple stream model* [180] is a generalization of the HMM that models data from parallel time series in a *synchronous* fashion. At each time t there is a single hidden state, denoted s_t. Figure 6.2 shows HTK Multiple Stream Model structure. The conditional independence assumptions for the case of two observation streams are:

- $o_t^1 \perp \{o_1^1, o_1^2, \ldots, o_{t-1}^1, o_{t-1}^2, o_t^2, o_{t+1}^1, o_{t+1}^2, \ldots, o_T^1, o_T^2, s_1, \ldots, s_{t-1}, s_{t+1}, \ldots, s_T\} \| s_t$
- $o_t^2 \perp \{o_1^1, o_1^2, \ldots, o_{t-1}^1, o_{t-1}^2, o_t^1, o_{t+1}^1, o_{t+1}^2, \ldots, o_T^1, o_T^2, s_1, \ldots, s_{t-1}, s_{t+1}, \ldots, s_T\} \| s_t$
- $s_t \perp \{o_1^1, o_1^2, \ldots, o_{t-1}^1, o_{t-1}^2, s_1, \ldots, s_{t-2}\} \| s_{t-1}$

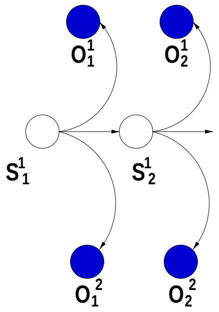


Figure 6.2 *HTK Multiple Streams Model*

6.1.3 Independent Streams and Multiband Models

The *independent streams model* uses a separate HMM to model each time series independently. The model therefore allows complete asynchrony between the different observation streams. It is closely related to the *multiband* model investigated by [16, 65, 108]. At each time t there are K hidden states, denoted s_t^k for $1 \leq k \leq K$. Figure 6.3 shows the independent streams model structure. The conditional independence assumptions for the case of two streams are:

- $o_t^1 \perp \{o_1^1, \ldots, o_{t-1}^1, o_{t+1}^1, \ldots, o_T^1, s_1^1, \ldots, s_{t-1}^1, s_{t+1}^1, \ldots, s_T^1, o_1^2, \ldots, o_T^2, s_1^2, \ldots, s_T^2\} || s_t^1$
- $o_t^2 \perp \{o_1^2, \ldots, o_{t-1}^2, o_{t+1}^2, \ldots, o_T^2, s_1^2, \ldots, s_{t-1}^2, s_{t+1}^2, \ldots, s_T^2, o_1^1, \ldots, o_T^1, s_1^1, \ldots, s_T^1\} || s_t^2$
- $s_t^1 \perp \{o_1^1, \ldots, o_{t-1}^1, s_1^1, \ldots, s_{t-2}^1, o_1^2, \ldots, o_T^2, s_1^2, \ldots, s_T^2\} || s_{t-1}^1$
- $s_t^2 \perp \{o_1^2, \ldots, o_{t-1}^2, s_1^2, \ldots, s_{t-2}^2, o_1^1, \ldots, o_T^1, s_1^1, \ldots, s_T^1\} || s_{t-1}^2$

The model has been investigated in this form only for isolated word recognition tasks, since (at least for the parallel time series representations of speech that have been investigated) it has been argued that allowing asynchrony between observation streams over entire utterances would not be desirable.

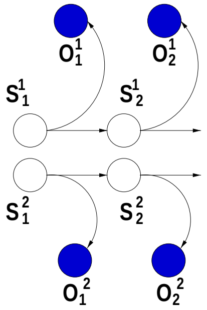


Figure 6.3 *Independent Streams Model*

6.1.4 State-Coupled Model

The *State-Coupled Model* [31] is motivated by the observation that although asynchrony between observation streams may be beneficial, this does not mean that complete asynchrony (as in the independent streams model) is optimal. This motivation is similar to that for the models in the next chapter. At each time t there are K hidden states, denoted s_t^k for $1 \leq k \leq K$. Figure 6.4 shows the *state-coupled model* structure. The conditional independence assumptions for the case of two streams are:

- $o_t^1 \perp \{o_1^1, \ldots, o_{t-1}^1, o_{t+1}^1, \ldots, o_T^1, s_1^1, \ldots, s_{t-1}^1, s_{t+1}^1, \ldots, s_T^1, o_1^2, \ldots, o_T^2, s_1^2, \ldots, s_T^2\} \| s_t^1$
- $o_t^2 \perp \{o_1^2, \ldots, o_{t-1}^2, o_{t+1}^2, \ldots, o_T^2, s_1^2, \ldots, s_{t-1}^2, s_{t+1}^2, \ldots, s_T^2, o_1^1, \ldots, o_T^1, s_1^1, \ldots, s_T^1\} \| s_t^2$
- $s_t^1 \perp \{o_1^1, \ldots, o_{t-1}^1, s_1^1, \ldots, s_{t-2}^1, o_1^2, \ldots, o_{t-1}^2, s_1^2, \ldots, s_{t-1}^2\} \| s_{t-1}^1$
- $s_t^2 \perp \{o_1^1, \ldots, o_T^1, s_1^1, \ldots, s_{t-1}^1, s_{t+1}^1, \ldots, s_T^1, o_1^2, \ldots, o_{t-1}^2, s_1^2, \ldots, s_{t-2}^2\} \| s_t^1, s_{t-1}^2$

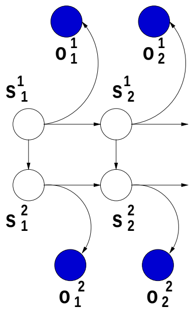


Figure 6.4 *State-Coupled Model*

6.1.5 Extended-PMC Model

The *Extended-PMC Model* approach reflects the same motivation as the State-Coupled Model: asynchrony between observation streams may be beneficial but complete asynchrony (as in the independent streams model) is not necessarily optimal. The approach also retains the separate processing of subbands but introduces soft synchrony constraints. The model is investigated in [104, 163].

Note first that an independent streams model with fixed parameters can be converted into a equivalent HMM using a variant of HMM decomposition or Parallel Model Combination [48, 166]. The state space of this PMC-HMM is the Cartesian product of the state spaces of the individual, per-stream HMMs and the observations produced are $\mathbf{O}_t = (o_t^1, \ldots, o_t^K)$; the transition- and output-probabilities are formed using $P(J|I) = \prod_k P(j^k|I)$ and $p(\mathbf{O}_t|J) = \prod_k p(o_t^k|J)$, where j^k is a state in the kth per-stream HMM and $I = (i^1, \ldots, i^K)$ is a state in the combined HMM. The Extended-PMC approach first trains an independent streams model allowing complete asynchrony between observation streams. The resulting independent streams model is converted into the equivalent PMC-HMM and a second stage then retrains only the *transition* probabilities within this PMC-HMM. This approach is essentially a knowledge-based initialization of an HTK multiple streams model as in Figure 6.2, with a specific choice of topology and in which only a subset of the parameters are reestimated after initialization, so a new diagram is not required.

6.2 Discussion

All of the approaches above attempt to extend existing conventional HMMs to allow modeling of asynchrony, but they do it in an adhoc way. As a consequence none allows the degree of allowable asynchrony to be varied to suit the underlying modelling requirements. The next chapter will discuss a general family of models that allows varying degrees of coupling between the different time series and will show that, with the exception of the State-Coupled Model of Section 6.1.4, the models discussed above may be considered as special cases.

7

Loosely-Coupled Hidden Markov Models

This chapter introduces and develops the theory of loosely-coupled or factorial HMMs (FHMMs), which are potentially appropriate for modelling multiple loosely-coupled time series and were first discussed in [52]. As noted at the end of the previous chapter, this family generalizes many of the approaches to this problem that have been investigated within the speech community. Section 7.1 reviews the general factorial model and then introduces two members of the FHMM family that will be investigated in the empirical study of Chapter 9. The first is drawn from the machine learning literature [150]; the second, introduced here for the first time, makes fewer a-priori assumptions about the nature of the data to be modelled and has properties potentially better suited to speech modelling. Both of the models share some basic assumptions, which will be discussed in Section 7.2; the specifics of each model are developed in Sections 7.3 and 7.4.

7.1 Loosely-Coupled or Factorial HMMs

As mentioned in the previous chapter, one obvious scheme for modelling loosely-coupled time series combines the K observations at each time t into a single observation vector $\mathbf{O}_t = (o_t^1, \ldots, o_t^K)$ and then models these combined observation vectors using a standard HMM. However, the resulting model may not be a parsimonious representation of the data. Alternatively, each observation stream k could be modelled independently using a single N-state HMM per stream. However, this scheme fails to capture any coupling between the different time series. An intermediate approach is to combine the K independent HMMs into a joint model which can capture something of the coupling between different streams. We can form a combined or *factorial* HMM (FHMM) in which (i) the hidden state space is the Cartesian product of the state spaces of the individual HMMs (see Figure 7.1), and (ii) the observations $\mathbf{O}_t$ are formed by concatenating the individual stream observations at time t, ie. $\mathbf{O}_t = (o_t^1, \ldots, o_t^K)$. The Cartesian product hidden state space will be referred to as the *metastate space*, to distinguish it from the state spaces of the original independent HMMs for each stream.

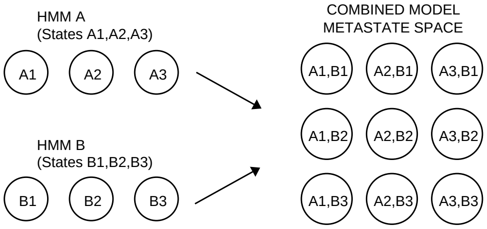


Figure 7.1 *Metastate Space From Combined HMMs A, B*

The FHMM as just described is equivalent to a standard HMM, in which the metastates and observations have internal structure. However, as K and N increase, estimation of the FHMM transition matrix and output distributions will be intractable both computationally and in terms of robust parameter estimation. Recent work in the machine learning literature handles these difficulties through additional assumptions and approximations which exploit the internal, combinatorial structure of the metastates and observations both to reduce the number of parameters and as the basis for efficient, approximate training and decoding algorithms eg. [52, 150].

The next section discusses the set of basic parameter-reducing assumptions that are common to the instances of the factorial HMM family investigated in this thesis. The family of models making these basic assumptions contains several standard speech models as special cases, when K and any additional parameter reduction schemes are chosen appropriately; later sections consider schemes for parameter reduction that result in models more general than these standard speech models.

7.2 Basic Loosely-Coupled Model

Notation P denotes probability mass functions, p denotes densities. Each of the K time series comprises D-dimensional observations[1]. The KD-dimensional combined observations are denoted $\mathbf{O}_t = (o_t^1, \ldots, o_t^K)$. A full, combined observation sequence of length T is denoted $\mathbf{O} = \mathbf{O}_1, \ldots, \mathbf{O}_T$. The presentation of the previous section will be generalized to allow L underlying Markov chains, where it is not necessary that $L = K$; metastates are therefore described by L-tuples $I = (i^1, \ldots, i^L)$ and $J = (j^1, \ldots, j^L)$. The full set of metastates is denoted Θ_{meta}. Each chain has N possible states[2]. The full set of states in the k-th chain is denoted θ_k. Using this notation, the parameters to be estimated in the FHMM are those associated with the metastate priors $P(J)$, the metastate transition matrix $P(J|I)$ and the combined observation output distributions $p(\mathbf{O}_t|J)$.

All parameter reduction schemes considered in this work make the following basic conditional independence assumptions:

- independence of initial metastate components

$$P(J) = \prod_{l=1}^{L} P(j^l) \tag{7.1}$$

- conditional independence of metastate components given previous metastate:

$$P(J|I) = \prod_{l=1}^{L} P(j^l|I) \tag{7.2}$$

- conditional independence of observation components given current metastate:

$$p(\mathbf{O}_t|J) = \prod_{k=1}^{K} p(o_t^k|J) \tag{7.3}$$

These conditional independence assumptions are illustrated in Figures 7.2 and 7.3.

[1] It is straightforward but notationally cumbersome to allow observations of different dimensionality in each stream.

[2] It is straightforward but notationally cumbersome to allow different numbers of states per chain l.

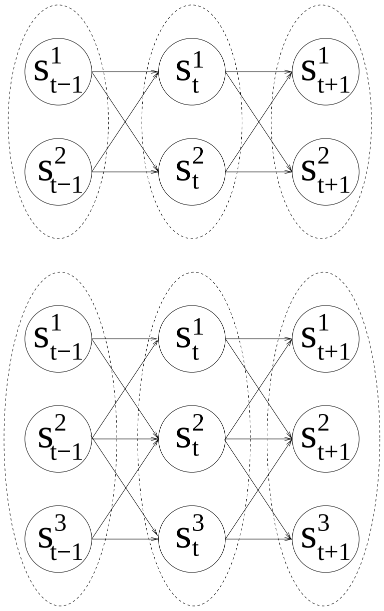


Figure 7.2 *Conditional independence structure for state transitions in the Basic FHMM (for cases L=2 and L=3). Metastates are represented by dashed lines; component states by solid lines.*

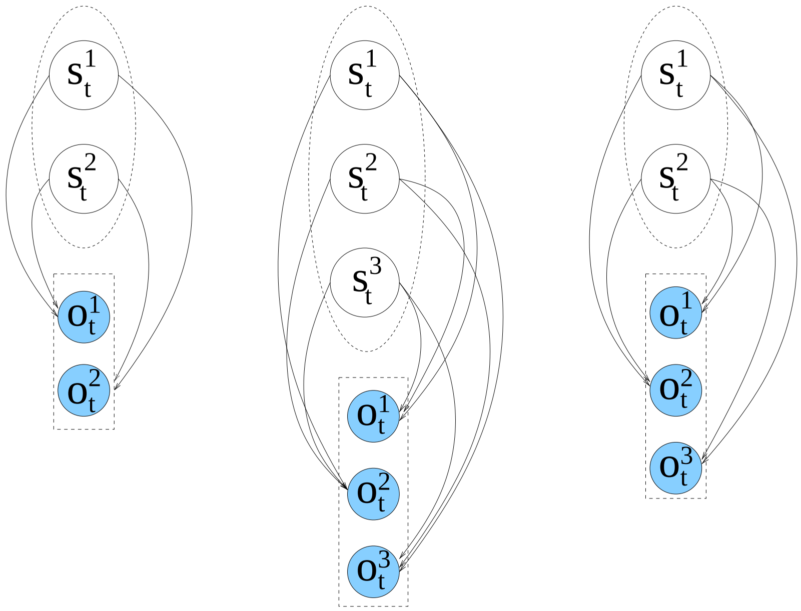


Figure 7.3 *Conditional independence structure for observations in the basic FHMM (for cases (1) K=L=2, (2) K=L=3, and (3) K=3, L=2). Metastates and combined observations are represented by dashed lines; component states and observations by solid lines.*

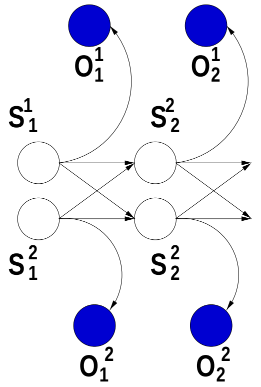


Figure 7.4 *Transition-Only Coupled Basic FHMM*

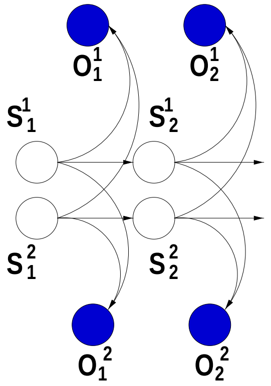


Figure 7.5 *Observation-Only Coupled Basic FHMM*

These basic assumptions will allow the investigation of various schemes for introducing coupling into a combined model. These correspond to the special cases where additional conditional independence assumptions are introduced such that only transition-related or only observation-related probabilities are made dependent upon full metastate identity, as opposed to the case where both are metastate dependent. The case where only transition distributions can be dependent upon metastates will be referred to as *transition-only coupled*, the case where only output distributions can be dependent upon metastates will be referred to as *observation-only coupled* and the general case where both observation and transition distributions may depend upon metastates is *fully coupled*. The conditional independence assumptions made in each case are illustrated in Figures 7.4, 7.5 and 7.6 to clarify the terminology. (Interpretation of conditional independence assumptions from such diagrams is discussed in (Appendix J)).

Several of the conventional speech models discussed in the previous chapter are special cases of the basic model described above. Setting $K = L = 1$ gives the standard *HMM*. Setting $L = 1$ and K to the number of time series gives the *HTK multiple stream model*. Setting $L = K$ and making the additional conditional independence assumptions $p(j^k|I) = p(j^k|i^k)$ and $p(o_t^k|J) = p(o_t^k|j^k)$ for $1 \leq k \leq K$ gives the *independent streams model*. It is also (theoretically) straightforward to incorporate stream exponent weighting parameters into this basic model, as is often done in the independent streams or HTK multiple stream models, where these can be estimated using (for example) the MMI (Maximum Mutual Information) [18], FD (Frame Discrimination) [78] or MCE

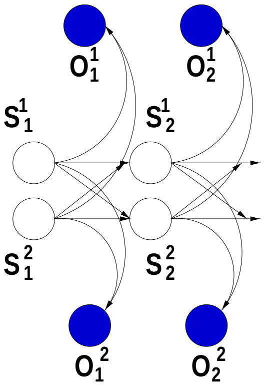


Figure 7.6 *Fully-Coupled Basic FHMM*

(Minimum Classification Error) [77] criterion. Such parameters might be particularly appealing for articulatory or phonological feature modelling, where they could emphasize the subsets of *critical features* important for distinguishing particular sounds.

Our real interest in FHMMs lies in the possibility of using new parameter reduction schemes to obtain more flexible models than these for capturing coupling between the K time series. There are many ways to reduce the number of FHMM parameters to a number that can be robustly estimated: this thesis investigates two possibilities. The first scheme introduces additional assumptions about the nature of the data to be modelled. These *Mixed-Memory Assumptions* were first proposed by [150]. The second scheme uses a data-driven approach to the reduction of parameters and makes fewer a-priori assumptions about the data. These schemes are described in detail in the next two sections. Note that, in both sections, the presentation will assume that $L = K$ for notational simplicity; generalization to $L \neq K$ is straightforward. Finally, we note that a third parameter reduction scheme is proposed in [52]. This was applied to speech modelling in [102]. Since it gave unpromising results, the scheme will not be pursued here.

7.3 Mixed-Memory Assumptions

This section discusses an approach to parameter reduction through the introduction of additional, a-priori assumptions about the nature of the data to be modelled [150]. Models using this parameter reduction scheme will be referred to as Mixed-Memory Factorial HMMs (MM-FHMMs).

7.3.1 Parameter Reduction Scheme

The Mixed-Memory Assumptions are:

- model conditional probability of state components given metastates using a convex combination of *cross-transition* distributions:

$$P(j^k|I) = \sum_{l=1}^{K} \psi^k(l) a^{kl}(j^k|i^l) \tag{7.4}$$

- model the prior probability of state components using a convex combination of *cross-stream* prior distributions[3]:

$$P(j^k) \quad = \quad \sum_{l=1}^{K} \psi^k(l) \pi^{kl}(j^k) \tag{7.5}$$

- model the conditional probability of per-stream observation vectors with a convex combination of *cross-emission* distributions:

$$p(o_t^k|J) = \sum_{l=1}^{K} \phi^k(l) b^{kl}(o_t^k|j^l) \tag{7.6}$$

The MM-FHMM parameter set is thus $\lambda = (\pi, A, B, \phi, \psi)$ where:

- **Cross-Stream Priors** π

  The parameters $\pi^{kl}(j^k)$ are $1 \times N$ matrices, of which there are K^2, giving a total of K^2N parameters. All matrix elements are non-negative. Matrix rows must satisfy sum-to-one constraints: for each k and l, we require $\sum_{j^k \in \Theta_k} \pi^{kl}(j^k) = 1$.

- **Cross-Transition Matrices** A

  The parameters $a^{kl}(j^k|i^l)$ are $N \times N$ cross-transition matrices, of which there are K^2, giving a total of K^2N^2 transition parameters. All matrix elements are non-negative. Matrix rows must satisfy sum-to-one constraints: for each k, l and each state $i^l \in \Theta_l$, we require $\sum_{j^k \in \Theta_k} a^{kl}(j^k|i^l) = 1$.

- **Cross-Emission Distributions** B

  The $b^{kl}(o_t^k|j^l)$ are K^2N cross-emission output distributions, which may be continuous or discrete. Where each time series comprises D-dimensional observations and output densities are full covariance gaussians, this requires a total of $K^2ND\frac{(3+D)}{2}$ observation-related parameters.

[3] Making the same assumptions for the stream-state priors as for the conditional probability of state components given metastates will simplify implementation by allowing the former to be treated as a special case of the latter for estimation and probability calculations.

- **Model-dependent Mixture Weights ϕ, ψ**

  Parameters $\psi^k(l)$, $\phi^k(l)$ are mixture weights. They are fixed for a single model, and give a measure of the dependency between different streams, using a total $2K^2$ parameters. They are constrained to be non-negative and to satisfy sum-to-one constraints ie. for each k we require $\sum_{l=1}^{K} \psi^k(l) = \sum_{l=1}^{K} \phi^k(l) = 1$.

For the case of D-dimensional observations in each time series and multivariate Gaussian output densities, the model would have $K^2N(\frac{3D+D^2}{2} + N)$ free parameters vs. $N^K(\frac{3D+D^2}{2} + N^K)$ for the combined, full metastate space model.

The three types of coupling discussed in Section 7.2 can be investigated as follows. An *observation-only* coupled MM-FHMM is obtained by seting ψ to the $K \times K$ identity matrix. A *transition-only* coupled MM-FHMM is obtained by setting ϕ to the $K \times K$ identity matrix. Finally, a *fully-coupled* MM-FHMM corresponds to the general case of unrestricted ϕ and ψ.

7.3.2 Estimation of Model Parameters

ML parameter estimation is achieved using an EM algorithm [33]. This section illustrates the complete data set and presents parameter update equations without proof. Details of the derivation and of the calculation of the necessary posterior probabilities are given in Appendix D.

Following [150], Eqns (7.4)-(7.6) are viewed as mixture models, introducing two new types of latent variable in addition to those denoting the metastate sequence taken through the model. The new latent variables, denoted by y_t^k and x_t^k below, encode the identity of the cross-emission distribution and cross-transition matrix (ie. the distributions within each mixture model) used in each stream k at each t. Figure 7.7 illustrates the information provided by the x_t^k and y_t^k variables.

The notation for latent variables is as follows:

- s_t^k : state occupied in stream k at time t;
- $\mathbf{S}_t = (s_t^1, \ldots, s_t^K)$: metastate occupied at t;
- $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$: a sequence of metastates;
- $\mathcal{S} = \{\mathbf{S}\}$: the set of possible metastate sequences;
- x_t^k : the hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_{t-1}$ which determines the matrix used for the transition into s_t^k;
- $\mathbf{X}_t = (x_t^1, \ldots, x_t^K)$;
- $\mathbf{X} = \mathbf{X}_1, \ldots, \mathbf{X}_T$: a complete sequence of transition-predicting state component vectors for an utterance of length T;
- $\mathcal{X} = \{\mathbf{X}\}$ the set of possible sequences;

- y_t^k : the hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_t$ which determines the output probability for o_t^k;

- $\mathbf{Y}_t = (y_t^1, \ldots, y_t^K)$;

- $\mathbf{Y} = \mathbf{Y}_1, \ldots, \mathbf{Y}_T$: a complete sequence of observation-predicting state component vectors for an utterance of length T;

- $\mathcal{Y} = \{\mathbf{Y}\}$: the set of possible sequences

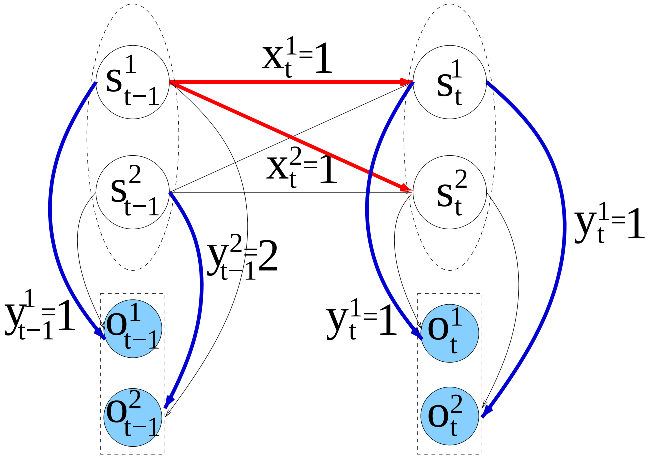


Figure 7.7 *Bold lines show information specified by the hidden (vector) variables Y_{t-1}, X_t, Y_t. Metastates and combined observations are represented by dashed lines; component states and observations by solid lines.*

For the case where each $b^{kl}(o_t^k|j^l)$ is modelled using a single, full covariance, multivariate Gaussian $\mathcal{N}(\mu_j^{kl}, \Sigma_j^{kl})$, the reestimation formulae are as follows:

$$\hat{\pi}^{kl}(j^k) = \frac{P(s_1^k = j^k, x_1^k = l|\mathbf{O})}{P(x_1^k = l|\mathbf{O})} \tag{7.7}$$

$$\hat{a}^{kl}(j^k|i^l) = \frac{\sum_{t=2}^T P(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})}{\sum_{t=2}^T P(s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})} \tag{7.8}$$

$$\hat{\psi}^k(l) = \frac{\sum_{t=1}^T P(x_t^k = l|\mathbf{O})}{T} \tag{7.9}$$

$$\hat{\phi}^k(l) = \frac{\sum_{t=1}^T P(y_t^k = l|\mathbf{O})}{T} \tag{7.10}$$

$$\hat{\mu}_j^{kl} = \frac{\sum_{t=1}^T P(y_t^k = l, s_t^l = i^l|\mathbf{O}) o_t^k}{\sum_{t=1}^T P(y_t^k = l, s_t^l = i^l|\mathbf{O})} \tag{7.11}$$

$$\hat{\Sigma}_j^{kl} = \frac{\sum_{t=1}^T P(y_t^k = l, s_t^l = i^l|\mathbf{O}) o_t^k (o_t^k)^T}{\sum_{t=1}^T P(y_t^k = l, s_t^l = i^l|\mathbf{O})} - \hat{\mu}_j^{kl}(\hat{\mu}_j^{kl})^T \tag{7.12}$$

Generalization to training using multiple observation sequences is straightforward.

7.4 Data-Driven Parameter Reduction

The previous scheme introduced *a priori* assumptions to reduce the number of FHMM parameters. This section proposes a data-driven scheme which will automatically determine dependencies that are usefully distinguished in an ML sense[4]. The approach has two advantages over the previous scheme: (1) fewer assumptions are made about the data to be modelled, and (2) a left-to-right transition topology without skip transitions[5] can be enforced within the metastate space. The reader should note that such topology restrictions have proven particularly important for speech modelling and have been almost universally adopted in that field. However, since such restrictions are of interest only for specific applications, further discussion of the difficulties associated with their enforcement in the MM-FHMM are postponed until the experimental study of Chapter 9. Models using this particular parameter reduction scheme will be referred to as Parameter-Tied Factorial HMMs (PT-FHMMs).

7.4.1 Parameter Reduction Scheme

The primary goal of the parameter reduction scheme is to reduce the number of FHMM parameters to a number which can be robustly estimated using the available data whilst retaining some ability to model coupling between streams. The underlying problem - more model parameters than can be estimated reliably from the available data - is encountered frequently in the speech recognition community and is commonly approached using some variant of *parameter tying*. Parameter tying reduces the number of free parameters to be estimated by putting "similar" constructs (eg. metastates) into equivalence classes and insisting that constructs in a class share the same model parameters (eg. transition or output distributions), whilst "dissimilar" constructs continue to be modelled using different parameters. The precise definitions of the sets of "similar" constructs may be specified a-priori by the modeller, but are more commonly determined using a data-driven approach. The same type of data-driven parameter tying approach can be applied to the problem of parameter reduction within FHMMs.

Our approach to reducing the number of *observation-related* parameters in the basic factored model of Section 7.2 will be, for each stream k, to identify classes of "equivalent" metastates and then to tie the stream k observation distributions $p(o_t^k|J)$ across each such class. The number of *transition-related* parameters will be reduced in a similar fashion: for each chain k, classes of "equivalent" metastates will be identified and then the chain k transition distributions $P(j^k|I)$ will be tied across each such class.

Consider the case of reducing the number of parameters in the $p(o_t^k|J)$ distributions which are used to model the observations in some stream k. (A similar approach will be used for *transition-related* parameters). The basic problem to be solved is as follows. Let Θ_{meta} denote the set of all metastates. Then we seek a partition $C_1^{obs,k}, \ldots, C_{NClasses(obs,k)}^{obs,k}$ of Θ_{meta} to be used in tying the stream k observation distributions $p(o_t^k|J)$. Thus $\bigcup_{n=1}^{NClasses(obs,k)} C_n^{obs,k} = \Theta_{meta}$ and $C_n^{obs,k} \cap C_m^{obs,k} = \emptyset$ for $m \neq n$. We emphasize that there is no requirement that the equivalence classes defined for tying stream k observation-related distributions be the same as the equivalence classes defined for tying the observation-related distributions for any other stream; further, the equivalence classes need not be the same as the equivalence classes defined for the purposes of tying transition distributions $P(j^l|I)$ for any state chain l $(1 \leq l \leq K)$.

Two issues must be addressed to solve this problem: determining an appropriate num-

[4] The idea for this model arose during a discussion with Mark Gales.

[5] A left-to-right topology allows transitions to occur only between metastates $I = (i^1, \ldots, i^K)$ and $J = (j^1, \ldots, j^K)$ where $j^k \in \{i^k, i^k + 1\}$ for each k; all other transitions are initialized to have zero probability.

ber of equivalence classes $NClasses(obs, k)$ and, given $NClasses(obs, k)$, determining a "good" mapping between metastates and the equivalence classes. These issues will be addressed using a greedy, hierarchical partitioning procedure. At the start of the algorithm, all metastates are placed into a single equivalence class. A locally optimal, binary partition (with respect to some objective function) is found for this class, which is split to give two new equivalence classes. The algorithm continues by splitting the existing class for which the associated locally optimal, binary partition leads to the greatest increase in the objective function. This greedy splitting procedure terminates when either (1) gains in objective function fall below a threshold or (2) further splitting of any class would create equivalence classes with insufficient data points. The minimum likelihood-gain threshold and the data-insufficiency thresholds are chosen empirically. This hierarchical approach is similar in flavour to the LBG algorithm [101] but, as we shall see, the objective function and notion of a class "centroid" will be somewhat different.

The next two sections describe the objective function and the methods used for obtaining a locally optimal partition of a set of metastates.

7.4.1.1 Objective Function

Direct use of a maximum likelihood objective function is computationally expensive. However, as observed in eg. [125], use of the EM auxiliary function $\mathcal{Q}(\lambda, \hat{\lambda})$ is tractable and increasing $\mathcal{Q}(\lambda, \hat{\lambda})$ guarantees the likelihood of the data is non-decreasing [33].

Let $\mathcal{C}^{obs,k}$ denote a set of equivalence classes defined amongst metastates for the purposes of tying the stream k observation-related distributions $p(o_t^k|J)$ and $\mathcal{C}^{trans,k}$ denote a set of equivalence classes defined amongst metastates for the purposes of tying the chain k transition-related distributions $P(j^k|I)$. The EM auxiliary function is:

$$\begin{aligned}
\mathcal{Q}(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{j^k \in \theta_k} P(s_1^k = j^k|\mathbf{O}) \log \hat{P}(j^k) \\
&+ \sum_{k=1}^{K} \sum_{C \in \mathcal{C}^{obs,k}} \sum_{J \in C} \sum_{t=1}^{T} P(\mathbf{S}_t = J|\mathbf{O}) \log \hat{p}(o_t^k|C) \\
&+ \sum_{k=1}^{K} \sum_{C \in \mathcal{C}^{trans,k}} \sum_{I \in C} \sum_{j^k \in \theta_k} \sum_{t=2}^{T} P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O}) \log \hat{P}(j^k|C) \\
&= Q_\pi + \sum_{k=1}^{K} Q_{Bk} + \sum_{k=1}^{K} Q_{Ak}
\end{aligned}$$

where

$$\begin{aligned}
Q_\pi &\stackrel{\text{def}}{=} \sum_{k=1}^{K} \sum_{j^k \in \theta_k} P(s_1^k = j^k|\mathbf{O}) \log \hat{P}(j^k) \\
Q_{Bk} &\stackrel{\text{def}}{=} \sum_{C \in \mathcal{C}^{obs,k}} \sum_{J \in C} \sum_{t=1}^{T} P(\mathbf{S}_t = J|\mathbf{O}) \log \hat{p}(o_t^k|C) \\
Q_{Ak} &\stackrel{\text{def}}{=} \sum_{C \in \mathcal{C}^{trans,k}} \sum_{I \in C} \sum_{j^k \in \theta_k} \sum_{t=2}^{T} P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O}) \log \hat{P}(j^k|C)
\end{aligned}$$

The form of $\mathcal{Q}(\lambda, \hat{\lambda})$ will allow separate maximization of the parameters associated with each equivalence class, for each k and each type of distribution (observation or transition) independently.

Calculation of auxiliary function contributions Q_{Bk} and Q_{Ak} can be made more efficient using class sum occupancies and sufficient statistics, see Appendix G.

7.4.1.2 Locally-Optimal Binary Splits: Observation-Related Distributions

Repartitioning seeks a binary partition of an equivalence class $C^{obs,k}$ that leads to an locally-optimal increase in the auxiliary function. We assume single multivariate Gaussian models of classes are adequate for the purpose of finding the partitions. We define the *centroid* of an equivalence class of metastates $C^{obs,k}$ to be $(\mu_{C^{obs,k}}, \Sigma_{C^{obs,k}})$, the ML estimates for a single multivariate Gaussian model of $p(o_t^k|C)$ given the data in the class. We also abbreviate the state occupancy posteriors $P(\mathbf{S}_t = J|\mathbf{O})$ by $\gamma_t(J)$.

The following iterative procedure finds a locally optimal, binary partition of the metastates in $C^{obs,k}$ with respect to $\mathcal{Q}(\lambda, \hat{\lambda})$:

- **Initialization**: Define initial centroids for two new classes $C_1^{obs,k}$ and $C_2^{obs,k}$, denoted by $(\mu_{C_1^{obs,k}}, \Sigma_{C_1^{obs,k}})$ and $(\mu_{C_2^{obs,k}}, \Sigma_{C_2^{obs,k}})$;
- **Iteration**: until convergence:

  a. Find a new binary partition: map each metastate $J \in C^{obs,k}$ to the class that maximizes its contribution to $\mathcal{Q}(\lambda, \hat{\lambda})$. Thus, assign J to class $C_1^{obs,k}$ if

$$\sum_{t=1}^{T} \gamma_t(J) \log \hat{p}(o_t^k|\mu_{C_1^{obs,k}}, \Sigma_{C_1^{obs,k}}) \geq \sum_{t=1}^{T} \gamma_t(J) \log \hat{p}(o_t^k|\mu_{C_2^{obs,k}}, \Sigma_{C_2^{obs,k}}) \quad (7.13)$$

  b. Update centroids: given the current binary partition $C_1^{obs,k}$ and $C_2^{obs,k}$, update the class centroids to maximize $\mathcal{Q}(\lambda, \hat{\lambda})$. For $C \in \{C_1^{obs,k}, C_2^{obs,k}\}$:

$$\hat{\mu}_C = \frac{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J) o_t^k}{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)} \quad (7.14)$$

$$\hat{\Sigma}_C = \frac{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)(o_t^k - \hat{\mu}_C)(o_t^k - \hat{\mu}_C)^T}{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)} \quad (7.15)$$

The procedure will converge to a locally optimal, binary partition of the original equivalence class.

Computation of the distances in Eqn (7.13) and centroid updates in Eqns (7.14)-(E.1) can be made more efficient using class sum occupancies and sufficient statistics, see Appendix G.

7.4.1.3 Locally-Optimal Binary Splits: Transition-Related Distributions

Repartitioning seeks a binary partition of an equivalence class $C^{trans,k}$ that leads to an locally-optimal increase in the auxiliary function. We define the *centroid* of an equivalence class of metastates $C^{trans,k}$ to be the ML estimates for the transition distribution $P(j^k|C)$ given the data for that class. We also abbreviate the state occupancy posteriors $P(\mathbf{S}_t = J|\mathbf{O})$ by $\gamma_t(J)$ and the transition occupancy posteriors $P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O})$ by $\eta_t^k(j^k, I)$.

- **Initialization**: Define initial centroids for two new classes $C_1^{trans,k}$ and $C_2^{trans,k}$, denoted by $P(j^k|C_1^{trans,k})$ and $P(j^k|C_2^{trans,k})$;
- **Iteration**: until convergence:

  a. Find a new binary partition: map each metastate $I \in C^{trans,k}$ to the class that maximizes its contribution to $\mathcal{Q}(\lambda, \hat{\lambda})$. Thus, assign I to class $C_1^{trans,k}$ if

$$\sum_{j^k \in \theta_k} \sum_{t=2}^{T} \eta_t(j^k, I) \log \hat{P}(j^k|C_1^{trans,k}) \geq \sum_{j^k \in \theta_k} \sum_{t=2}^{T} \eta_t(j^k, I) \log \hat{P}(j^k|C_2^{trans,k}) \quad (7.16)$$

  b. Update centroids: given the current binary partition $C_1^{trans,k}$ and $C_2^{trans,k}$, update the class centroids to maximize $\mathcal{Q}(\lambda, \hat{\lambda})$. For $C \in \{C_1^{trans,k}, C_2^{trans,k}\}$ and $j^k \in \theta_k$:

$$\hat{p}(j^k|C) \quad = \quad \frac{\sum_{I \in C} \sum_{t=2}^{T} \eta_t^k(j^k, I)}{\sum_{I \in C} \sum_{t=2}^{T} \gamma_t(I)} \quad (7.17)$$

The procedure will converge to a locally optimal, binary partition of the original equivalence class.

Computation of the distances in Eqn (7.16) and ML updates in Eqn (7.17) can be calculated more efficiently using class sum occupancies and sufficient statistics, see Appendix G.

7.4.2 Estimation of Model Parameters

Once sets of distribution equivalence classes have been defined, ML parameter estimation is again achieved using an EM algorithm [33]. The parameters to be estimated are (for $1 \leq k \leq K$) the prior probabilities $P(j^k)$, the observation-related distributions $p(o_t^k|C)$ for $C \in \mathcal{C}^{obs,k}$ and transition-related distributions $p(j^k|C)$ for $C \in \mathcal{C}^{trans,k}$. In this case, the only latent variables required are those specifying metastate sequences through the model, denoted $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$. Details of the derivation and of the calculation of the necessary posterior probabilities are given in Appendix E. For the case where each $p(o_t^k|J)$ is modelled using a single, full covariance, multivariate Gaussian $\mathcal{N}(\mu_J^k, \Sigma_J^k)$, the reestimation formulae are as follows:

$$\begin{aligned}
\hat{P}(j^k) &= P(s_1^k = j^k|\mathbf{O}) \\
\hat{P}(j^k|C) &= \frac{\sum_{I \in C} \sum_{t=2}^{T} \eta_t^k(j^k, I)}{\sum_{I \in C} \sum_{t=2}^{T} \gamma_t(I)} \\
\hat{\mu}_C &= \frac{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J) o_t^k}{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)} \\
\hat{\Sigma}_C &= \frac{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)(o_t^k - \hat{\mu}_C)(o_t^k - \hat{\mu}_C)^T}{\sum_{J \in C} \sum_{t=1}^{T} \gamma_t(J)}
\end{aligned}$$

Generalization to training using multiple observation sequences is straightforward.

8

Approximate Algorithms For Loosely-Coupled Models

The previous chapter introduced the FHMM family and two special cases, the MM-FHMM and PT-FHMM. Exact likelihood calculations and EM-based estimation for the MM-FHMM and PT-FHMM require calculation of forward and backward probabilities in the metastate space of size N^K. The cost of the forward or backward calculations is therefore $\mathcal{O}(N^{2K}T)$. As K (the number of state chains) or N (the number of states in each chain) increase, this becomes intractable. This chapter considers schemes for more efficient model estimation and decoding, either through approximate algorithms or by identifying special cases of the model for which the calculation of forward and backward probabilities is more efficient. These schemes are presented in terms of the observation-only coupled MM-FHMM, but the approaches could be extended straightforwardly to the PT-FHMM. It is noted in passing that in addition to the algorithms below, sampling-based methods (most obviously the Gibbs sampler) are applicable to this problem but will not be investigated here (see eg. [13]).

8.1 More Efficient Forward-Backward Algorithm

The forward and backward calculations can be made more efficient for the observation-only coupled MM-FHMM[1]. Let $I = (i^1, \ldots, i^K), J = (j^1, \ldots, j^K)$ denote metastates, where $i^k, j^k \in \theta_k$ for each k. Define the following variables:

$$\begin{aligned}\alpha_t(J) &= p(\mathbf{O}_1, \ldots, \mathbf{O}_t, \mathbf{S}_t = J)\\ \alpha_t^0(J) &= p(\mathbf{O}_1, \ldots, \mathbf{O}_{t-1}, \mathbf{S}_{t-1} = J)\end{aligned}$$

and for $1 \leq k \leq K$:

$$\alpha_t^k(J) = p(\mathbf{O}_1, \ldots, \mathbf{O}_{t-1}, s_t^1 = j^1, \ldots, s_t^k = j^k, s_{t-1}^{k+1} = j^{k+1}, \ldots, s_{t-1}^K = j^K)$$

These quantities can be calculated efficiently using the recursions in the *Modified Forward Algorithm* that follows, where 1^k denotes no state has yet been entered in stream k.

[1] A similar speedup is proposed by [52] for factorial HMMs using an alternative parameter reduction scheme. Readers familiar with directed acyclic graphical models [94, 75] will find the existence of such a speedup obvious, observing that: (1) the cost of the general inference algorithm scales as the sum of the sizes of the state spaces of the cliques, and (2) the state spaces of cliques in the observation-only coupled model are smaller than in the fully coupled model.

Modified Forward Algorithm

Step 1: Initialization:

1a. $\alpha_1^0(1^1, \ldots, 1^K) = 1$ and $\alpha_1^0(J) = 0$ for each $J \in \Theta_{meta}$.
1b. For $1 \leq k \leq K$ and each $(j^1, \ldots, j^k, 1^{k+1}, \ldots, 1^K)$:

$$\alpha_1^k(j^1, \ldots, j^k, 1^{k+1}, \ldots, 1^K) = P(j^k)\alpha_1^{k-1}(j^1, \ldots, j^{k-1}, 1^k, \ldots, 1^K)$$

1c. $\alpha_1(J) = \alpha_1^K(J)p(\mathbf{O}_1|J)$ for each $J \in \Theta_{meta}$.

Step 2: Iteration for $t = 2, \ldots, T$:

2a. $\alpha_t^0(J) = \alpha_{t-1}(J)$ for each $J \in \Theta_{meta}$.
2b. For each $1 \leq k \leq K$ and each $(j^1, \ldots, j^k, i^{k+1}, \ldots, i^K) \in \Theta_{meta}$:

$$\alpha_t^k(j^1, \ldots, j^k, i^{k+1}, \ldots, i^K) = \sum_{i^k \in \theta_k} \alpha_t^{k-1}(j^1, \ldots, j^{k-1}, i^k, \ldots, i^K)P(j^k|i^k)$$

2c. $\alpha_t(J) = \alpha_t^K(J)p(\mathbf{O}_t|J)$ for each $J \in \Theta_{meta}$.

Step 3: Termination:

3a. $p(\mathbf{O}|\lambda) = \sum_{J \in \Theta_{meta}} \alpha_T(J)$.

At each $t = 1, \ldots, T$, we must calculate N^K values for each of the K $\alpha_t^k(J)$ variables and also for $\alpha_t(J)$. Each such value requires N additions and multiplications. Thus the complexity of this modified forward-backward algorithm is $\mathcal{O}(KN^{K+1}T)$, rather than $\mathcal{O}(N^{2K}T)$ for the standard forward algorithm in the metastate space.

Similar efficiency improvements are possible in the backward algorithm. Define the following variables:

$$\begin{array}{rcl} \beta_t(J) & = & p(\mathbf{O}_{t+1}, \ldots, \mathbf{O}_T|\mathbf{S}_t = J) \\ \beta_t^{K+1}(J) & = & p(\mathbf{O}_{t+1}, \ldots, \mathbf{O}_T|\mathbf{S}_{t+1} = J) \end{array}$$

and for $1 \leq k \leq K$:

$$\beta_t^k(J) \;\; = \;\; p(\mathbf{O}_{t+1}, \ldots, \mathbf{O}_T|s_{t+1}^1 = j^1, \ldots, s_{t+1}^{k-1} = j^{k-1}, s_t^k = j^k, \ldots, s_t^K = j^K)$$

These quantities can be calculated efficiently using the recursions in the *Modified Backward Algorithm* that follows, where 1^k denotes no state has yet been entered in stream k.

Modified Backward Algorithm

Step 1: (Arbitrary) Initialization:

1a. $\beta_T(J) = 1$ for all $J \in \Theta_{meta}$.

Step 2: Iteration for $t = T-1, \ldots, 1$:

2a. $\beta_t^{K+1}(J) = p(\mathbf{O}_{t+1}|J)\beta_{t+1}(J)$ for each $J \in \Theta_{meta}$.
2b. For each $K \geq k \geq 1$ and each $(i^1, \ldots, i^{k-1}, j^k, \ldots, j^K) \in \Theta_{meta}$:

$$\beta_t^k(i^1, \ldots, i^{k-1}, j^k, \ldots, j^K) \quad = \quad \sum_{i^k \in \Theta_k} \beta_t^{k+1}(i^1, \ldots, i^k, j^{k+1}, \ldots, j^K) p(i^k|j^k)$$

2c. For each $J \in \Theta_{meta}$, $\beta_t(J) = \beta_t^1(J)$.

Step 3: Termination:

3a. $\beta_0^{K+1}(I) = p(\mathbf{O}_1|I)\beta_1(I)$ for each $I \in \Theta_{meta}$.
3b. For $K \geq k \geq 1$ and each $(i^1, \ldots, i^{k-1}, 1^k, \ldots, 1^K)$:

$$\beta_0^k(i^1, \ldots, i^{k-1}, 1^k, \ldots, 1^K) \quad = \quad \sum_{i^k \in \theta_k} P(i^k)\beta_0^{k+1}(i^1, \ldots, i^k, 1^{k+1}, \ldots, 1^K)$$

3c. $p(\mathbf{O}|\lambda) = \beta_0^1(1^1, \ldots, 1^K)$.

8.2 Variational Approximations

8.2.1 Introduction to Variational Approximations

Parameter estimation algorithms based on variational methods are currently popular in the directed acyclic graphical models community, where ML estimation using the EM algorithm is often intractable[2]. This section outlines the basic arguments for the specific case of the observation-coupled MM-FHMM; [76] is a more general presentation.

Variational methods exploit one particular lower bound on the log likelihood of a set of data for likelihood approximation and as the basis of a parameter estimation procedure. In terms of the observation-only coupled MM-FHMM, we may express the bound as:

$$\begin{aligned}
\mathcal{L}(\lambda) = \ln p(\mathbf{O}|\lambda) &= \ln\{\sum_{\mathbf{S},\mathbf{Y}} p(\mathbf{O}, \mathbf{S}, \mathbf{Y}|\lambda)\} \\
&= \ln\{\sum_{\mathbf{S},\mathbf{Y}} Q(\mathbf{S}, \mathbf{Y}|\Psi) \frac{p(\mathbf{O}, \mathbf{S}, \mathbf{Y}|\lambda)}{Q(\mathbf{S}, \mathbf{Y}|\Psi)}\} \\
&\geq \sum_{\mathbf{S},\mathbf{Y}} Q(\mathbf{S}, \mathbf{Y}|\Psi) \ln \frac{p(\mathbf{O}, \mathbf{S}, \mathbf{Y}|\lambda)}{Q(\mathbf{S}, \mathbf{Y}|\Psi)} = \mathcal{L}_Q(\Psi, \lambda)
\end{aligned}$$

[2] Inference in the general case is NP-hard [28].

where $\mathcal{L}(\lambda)$ denotes the likelihood function, $Q(\mathbf{S}, \mathbf{Y}|\Psi)$ is a distribution over the hidden variables with parameters Ψ, and $\mathcal{L}_Q(\Psi, \lambda)$ denotes the lower bound of interest. The inequality in the third line follows by Jensen's inequality [146]. Note that $\mathcal{L}(\lambda)$ exceeds $\mathcal{L}_Q(\Psi, \lambda)$ by exactly $KL[Q(\mathbf{S}, \mathbf{Y}|\Psi)||p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda)]$, the Kullback-Leibler (KL) divergence between the distributions $Q(\mathbf{S}, \mathbf{Y}|\Psi)$ and $p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda)$, which is a non-negative quantity [29]. The lower bound may be tightened for each observation sequence $\mathbf{O}$ by adjusting the *variational parameters* Ψ of the *variational distribution* Q to minimize the KL divergence.

The lower bound above is useful for likelihood approximation. [113] observes that the lower bound $\mathcal{L}_Q(\Psi, \lambda)$ may also be usefully applied to ML parameter estimation problems using the following iterative procedure. The parameters of model p and of variational distribution Q at iteration k are denoted λ^k and Ψ^k respectively.

Variational Approximation-Based Learning

Step 1 Maximize $\mathcal{L}_Q(\Psi, \lambda^k)$ wrt Ψ.

$$\Psi^{k+1} = \arg\max_{\Psi} \mathcal{L}_Q(\Psi, \lambda^k)$$

This is equivalent to minimizing $KL[Q(\mathbf{S}, \mathbf{Y}|\Psi)||p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda^k)]$ wrt Ψ:

$$\Psi^{k+1} = \arg\min_{\Psi} KL[Q(\mathbf{S}, \mathbf{Y}|\Psi)||p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda^k)]$$

Step 2 Maximize $\mathcal{L}_Q(\Psi^{k+1}, \lambda)$ wrt λ.

$$\lambda^{k+1} = \arg\max_{\lambda} \mathcal{L}_Q(\Psi^{k+1}, \lambda)$$

The algorithm is guaranteed to increase the lower bound on the likelihood $\mathcal{L}_Q(\Psi, \lambda)$ at each step, although not necessarily the likelihood $\mathcal{L}(\lambda)$. Convergence of the algorithm may be assessed by monitoring changes in the lower bound.

To examine the operation of the variational learning algorithm, consider first the case where $Q(\mathbf{S}, \mathbf{Y}|\Psi)$ is allowed to range over all possible distributions over the hidden variables. A standard result ([29]) states that the distribution minimizing the KL divergence in Step 1 is $Q(\mathbf{S}, \mathbf{Y}|\Psi) = p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda^k)$, resulting in a KL divergence of zero. Thus, Step 2 seeks

$\lambda^{k+1} = \arg\max_\lambda \sum_{\mathbf{S}, \mathbf{Y}} p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda^k) \ln p(\mathbf{S}, \mathbf{Y}|\mathbf{O}, \lambda)$

which is equivalent to the standard EM algorithm [33].

The lower bound is of more general utility for likelihood approximation or learning in cases where standard likelihood calculations or EM estimation are intractable. In these cases, the family of distributions $Q(\mathbf{S}, \mathbf{Y}|\Psi)$ is assumed to have a form in which inference is more tractable than in the original model. For example, when working with directed acyclic graphical models, a model Q allowing tractable inference is often identified by simplifying the dependencies in the original model p for which inference is intractable.

8.2.2 Mean-Field Variational Approximation

This section investigates a very simple family Q potentially suitable for variational likelihood approximation or observation-related parameter estimation in an observation-only coupled MM-FHMM. The approximation provides a computationally cheap means of integrating K independent HMMs which have been trained on the K streams individually. The scheme as described will not be used to reestimate the transition parameters of the individual HMMs, only the observation-related parameters $\phi^k(l)$ and $b^{kl}(o_t^k|i^l)$.

The following discussion assumes that occupation of the exit metastate N at time $T+1$ is deterministic and guaranteed, and only metastates $\mathbf{S}_1, \ldots, \mathbf{S}_T$ are hidden ie. $\mathbf{S}_{T+1} = (N^1, \ldots, N^K)$. To simplify notation, the superscript k denoting states from a particular stream is dropped in the following discussion ie. $j^k \in \Theta_k$ will be written $j \in \Theta_k$, with the exception of N^k which denotes the exit state for stream k. Set Θ_k represents the set of *emitting* states in the model for stream k, and does not include N^k. The simplest variational approximation Q is a completely factorized approximation, in which all hidden variables are assumed independent given the observations. This approximation, often referred to as a Mean-Field approximation in statistical physics, can be written:

$$Q(\mathbf{S}, \mathbf{Y}|\Psi) \quad = \quad \prod_{t=1}^{T}\{\prod_{k=1}^{K} Q_t^{Sk}(s_t^k|\Psi_t^{Sk}) Q_t^{Yk}(y_t^k|\Psi_t^{Yk})\}$$

where

- $Q_t^{Sk}(s_t^k|\Psi_t^{Sk})$ denotes a pmf with parameters $\Psi_t^{Sk} = \{\Psi_{tj}^{Sk}|j \in \Theta_k\}$ and Ψ_{tj}^{Sk} denotes the probability of a particular outcome j^k;
- $Q_t^{Yk}(y_t^k|\Psi_t^{Yk})$ denotes a pmf with parameter set $\Psi_t^{Yk} = \{\Psi_{tl}^{Yk}|1 \leq l \leq K\}$, and Ψ_{tl}^{Yk} denotes the probability of a particular outcome l.

The lower bound for this case is therefore:

$$\begin{aligned}
\mathcal{L}_Q(\Psi, \lambda) \quad = \quad & \sum_{k=1}^{K} \sum_{j \in \Theta_k} Q_1^{Sk}(j) \ln \pi^{kk}(j) + \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{l=1}^{K} Q_t^{Yk}(l) \ln \phi^k(l) \\
+ \quad & \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{i \in \Theta_l} Q_t^{Yk}(l) Q_t^{Sl}(i) \ln b^{kl}(o_t^k|i) \\
+ \quad & \sum_{t=2}^{T} \sum_{k=1}^{K} \sum_{i \in \Theta_k} \sum_{j \in \Theta_k} Q_t^{Sk}(j) Q_{t-1}^{Sk}(i) \ln a^{kk}(j|i) \\
+ \quad & \sum_{k=1}^{K} \sum_{i \in \Theta_k} Q_T^{Sk}(i) \ln a^{kk}(N^k|i) \\
- \quad & \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{i \in \Theta_k} Q_t^{Sk}(i) \ln Q_t^{Sk}(i) - \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{l=1}^{K} Q_t^{Yk}(l) \ln Q_t^{Yk}(l)
\end{aligned}$$

To simplify maintenance of positivity, ensure appropriate normalization and to guarantee that no hidden event is assigned probability zero during the optimization procedure, a softmax form is used for each variational pmf. (Alternatively these requirements could be included as constraints.)

So for each $j \in \Theta_k$

$$Q_t^{Sk}(s_t^k = j|\Psi_t^{Sk}) \stackrel{\text{def}}{=} \frac{\exp \Psi_{tj}^{Sk}}{\sum_{i \in \Theta_k} \exp \Psi_{ti}^{Sk}}$$

and for each $1 \leq l \leq K$

$$Q_t^{Yk}(y_t^k = l|\Psi_t^{Yk}) \stackrel{\text{def}}{=} \frac{\exp \Psi_{tl}^{Yk}}{\sum_{\nu=1}^{K} \exp \Psi_{t\nu}^{Yk}}$$

A derivation of the following results is included in Appendix I.

Step 1 Minimization of KL Divergence.

Minimization will be implemented using a basic gradient descent procedure. The appropriate derivatives are:

$$
\begin{aligned}
\frac{\partial KL}{\partial \Psi_{\bar{t}\bar{j}}^{Y\bar{k}}} &= Q_{\bar{t}}^{Y\bar{k}}(\bar{l}) \left[(\sum_{l=1}^{K} -Q_{\bar{t}}^{Y\bar{k}}(l) \Psi_{\bar{t}l}^{Y\bar{k}}) + \Psi_{\bar{t}\bar{l}}^{Y\bar{k}} \right] \\
&- Q_{\bar{t}}^{Y\bar{k}}(\bar{l}) \sum_{l \neq \bar{l}} Q_{\bar{t}}^{Y\bar{k}}(l) \left[\ln \phi^{\bar{k}}(l) + \sum_{i \in \Theta_l} Q_{\bar{t}}^{Sl}(i) \ln b^{\bar{k}l}(o_{\bar{t}}^{\bar{k}}|i) \right] \\
&+ Q_{\bar{t}}^{Y\bar{k}}(\bar{l})(1 - Q_{\bar{t}}^{Y\bar{k}}(\bar{l})) \left[\ln \phi^{\bar{k}}(\bar{l}) + \sum_{i \in \Theta_{\bar{l}}} Q_{\bar{t}}^{S\bar{l}}(i) \ln b^{\bar{k}\bar{l}}(o_{\bar{t}}^{\bar{k}}|i) \right]
\end{aligned}
$$

For $1 < t < T$:

$$
\begin{aligned}
&\frac{\partial KL}{\partial \Psi_{\bar{t}\bar{j}}^{S\bar{k}}} = \\
&\quad Q_{\bar{t}}^{S\bar{k}}(\bar{j}) \left[(\sum_{j \in \Theta_{\bar{k}}} -Q_{\bar{t}}^{S\bar{k}}(j) \Psi_{\bar{t}j}^{S\bar{k}}) + \Psi_{\bar{t}\bar{j}}^{S\bar{k}} \right] \\
&- \quad Q_{\bar{t}}^{S\bar{k}}(\bar{j}) \sum_{i \neq \bar{j} \in \Theta_{\bar{k}}} Q_{\bar{t}}^{S\bar{k}}(i) \left[\sum_{k=1}^{K} Q_{\bar{t}}^{Yk}(\bar{k}) \ln b^{k\bar{k}}(o_{\bar{t}}^{k}|i) + \sum_{j \in \Theta_{\bar{k}}} Q_{\bar{t}-1}^{S\bar{k}}(j) \ln a^{\bar{k}\bar{k}}(i|j) + \sum_{j \in \Theta_{\bar{k}}} Q_{\bar{t}+1}^{S\bar{k}}(j) \ln a^{\bar{k}\bar{k}}(j|i) \right] \\
&+ \quad Q_{\bar{t}}^{S\bar{k}}(\bar{j})(1 - Q_{\bar{t}}^{S\bar{k}}(\bar{j})) \left[\sum_{k=1}^{K} Q_{\bar{t}}^{Yk}(\bar{k}) \ln b^{k\bar{k}}(o_{\bar{t}}^{k}|\bar{j}) + \sum_{j \in \Theta_{\bar{k}}} Q_{\bar{t}-1}^{S\bar{k}}(j) \ln a^{\bar{k}\bar{k}}(\bar{j}|j) + \sum_{j \in \Theta_{\bar{k}}} Q_{\bar{t}+1}^{S\bar{k}}(j) \ln a^{\bar{k}\bar{k}}(j|\bar{j}) \right]
\end{aligned}
$$

For $t = 1$:

$$
\begin{aligned}
\frac{\partial KL}{\partial \Psi_{\bar{t}\bar{j}}^{S\bar{k}}} &= Q_{\bar{t}}^{S\bar{k}}(\bar{j}) \left[(\sum_{j \in \Theta_{\bar{k}}} -Q_{\bar{t}}^{S\bar{k}}(j) \Psi_{\bar{t}j}^{S\bar{k}}) + \Psi_{\bar{t}\bar{j}}^{S\bar{k}} \right] \\
&- Q_{\bar{t}}^{Y\bar{k}}(\bar{l}) \sum_{l \neq \bar{l}} Q_{\bar{t}}^{Y\bar{k}}(l) \left[\ln \phi^{\bar{k}}(l) + \sum_{i \in \Theta_l} Q_{\bar{t}}^{Sl}(i) \ln b^{\bar{k}l}(o_{\bar{t}}^{\bar{k}}|i) \right] \\
&+ Q_{\bar{t}}^{Y\bar{k}}(\bar{l})(1 - Q_{\bar{t}}^{Y\bar{k}}(\bar{l})) \left[\ln \phi^{\bar{k}}(\bar{l}) + \sum_{i \in \Theta_{\bar{l}}} Q_{\bar{t}}^{S\bar{l}}(i) \ln b^{\bar{k}\bar{l}}(o_{\bar{t}}^{\bar{k}}|i) \right]
\end{aligned}
$$

And for $t = T$:

$$\begin{aligned}
\frac{\partial KL}{\partial \Psi^{S\bar{k}}_{\bar{t}\bar{j}}} &= Q^{S\bar{k}}_{\bar{t}}(\bar{j}) \left[(\sum_{j \in \Theta_{\bar{k}}} -Q^{S\bar{k}}_{\bar{t}}(j) \Psi^{S\bar{k}}_{\bar{t}j}) + \Psi^{S\bar{k}}_{\bar{t}\bar{j}} \right] \\
&- Q^{Y\bar{k}}_{\bar{t}}(\bar{l}) \sum_{l \neq \bar{l}} Q^{Y\bar{k}}_{\bar{t}}(l) \left[\ln \phi^{\bar{k}}(l) + \sum_{i \in \Theta_l} Q^{Sl}_{\bar{t}}(i) \ln b^{\bar{k}l}(o^{\bar{k}}_{\bar{t}}|i) \right] \\
&- Q^{S\bar{k}}_{T}(\bar{j}) \sum_{i \neq \bar{j} \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{T}(i) \left[\sum_{k=1}^{K} Q^{Yk}_{T}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{T}|i) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{T-1}(j) \ln a^{\bar{k}\bar{k}}(i|j) + \ln a^{\bar{k}\bar{k}}(N^{\bar{k}}|i) \right] \\
&+ Q^{S\bar{k}}_{T}(\bar{j})(1 - Q^{S\bar{k}}_{T}(\bar{j})) \left[\sum_{k=1}^{K} Q^{Yk}_{T}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{T}|\bar{j}) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{T-1}(j) \ln a^{\bar{k}\bar{k}}(\bar{j}|j) + \ln a^{\bar{k}\bar{k}}(N^{\bar{k}}|\bar{j}) \right]
\end{aligned}$$

Step 2 Maximization wrt λ.

Using a derivation similar to that for the standard M-step of the EM algorithm (eg. [121]), then the parameter updates $\hat{\lambda}$ are given by:

$$\begin{aligned}
\hat{\phi}^{k}(l) &= \frac{\sum_t Q^{Yk}_t(l|\Psi^{Yk}_t)}{T} \\
\hat{\mu}^{kl}_j &= \frac{\sum_t Q^{Yk}_t(l|\Psi^{Yk}_t) Q^{Sk}_t(j|\Psi^{Sk}_t) o^k_t}{\sum_t Q^{Yk}_t(l|\Psi^{Yk}_t) Q^{Sk}_t(j|\Psi^{Sk}_t)} \\
\hat{\Sigma}^{kl}_j &= \frac{\sum_t Q^{Yk}_t(l|\Psi^{Yk}_t) Q^{Sk}_t(j|\Psi^{Sk}_t)(o^k_t - \hat{\mu}^{kl}_j)(o^k_t - \hat{\mu}^{kl}_j)^T}{\sum_t Q^{Yk}_t(l|\Psi^{Yk}_t) Q^{Sk}_t(j|\Psi^{Sk}_t)}
\end{aligned}$$

The extension to multiple observation sequences is straightforward.

8.3 Algorithms Using Most-Likely Metastate Sequences

Large vocabulary ASR systems using HMMs generally assume in recognition that a single state sequence $\mathbf{S}^*$ dominates likelihood calculations, ie. $p(\mathbf{O}) \approx p(\mathbf{O}, \mathbf{S}^*)$, where if $\mathcal{S}$ denotes the set of length T state sequences through the model then $\mathbf{S}^* = \arg\max_{S \in \mathcal{S}} p(\mathbf{O}, S)$. The following two subsections describe first the Viterbi algorithm, which finds the optimal $\mathbf{S}^*$ but has the same computational order as the forward-backward calculations; the Chain Viterbi algorithm that follows is a more efficient scheme finding an approximation to $\mathbf{S}^*$. Such a metastate sequence can also be used in a *Viterbi training* (or, when $\mathbf{S}^*$ is approximate, a *Viterbi-training-like*) estimation scheme for MM-FHMMs. The associated parameter update equations are obtained by conditioning the posterior probabilities in Equations (7.7)-(7.12) on $\mathbf{S}^*$ as well as the utterance $\mathbf{O}$. This training algorithm is analogous to the use of standard Viterbi training in HMM systems based on Gaussian mixtures, in which a single state sequence is assumed to dominate but the Gaussian mixtures are trained using the EM algorithm. An alternative, not investigated here, makes the stronger assumption that a single $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Y}^*)$ sequence dominates likelihood calculations.

8.3.1 Viterbi Metastate Sequences

The most-likely metastate sequence $\mathbf{S}^* = \mathbf{S}^*_1, \ldots, \mathbf{S}^*_T$ can be obtained through standard Viterbi decoding in the *metastate* space, at a cost of $\mathcal{O}(N^{2K}T)$. The variable $\delta(t, J)$ is introduced:

$$\delta(t,J) \quad = \quad \max_{\mathbf{S}_1,\mathbf{S}_2,\ldots,\mathbf{S}_{t-1}} p(\mathbf{O}_1,\ldots,\mathbf{O}_t,\mathbf{S}_1,\ldots,\mathbf{S}_{t-1},\mathbf{S}_t=J|\lambda)$$

Values of $\delta(t,J)$ can be efficiently calculated as follows.

Viterbi Algorithm

Step 1: **Initialization** for each $J \in \Theta_{meta}$:

$$\begin{aligned}\delta(1,J) &= P(J)p(\mathbf{O}_1|J)\\ \psi(1,J) &= 0\end{aligned}$$

Step 2: **Iteration** for each $J \in \Theta_{meta}$ at $t = 2,\ldots,T$:

$$\begin{aligned}\delta(t,J) &= \max_{I\in\Theta_{meta}} \left[\delta(t-1,I)P(J|I)\right] p(\mathbf{O}_t|J)\\ \psi(t,J) &= \arg\max_{I\in\Theta_{meta}} \left[\delta(t-1,I)P(J|I)\right]\end{aligned}$$

Step 3: **Termination**:

$$\begin{aligned}P(O,\mathbf{S}^*) &= \max_{J\in\Theta_{meta}} [\delta(T,J)]\\ \mathbf{S}^*_T &= \arg\max_{J\in\Theta_{meta}} [\delta(T,J)]\end{aligned}$$

Step 4: **Backtrace** from $T-1,\ldots,1$ to obtain state sequence $\mathbf{S}^* = \mathbf{S}^*_1,\ldots,\mathbf{S}^*_T$:

$$\mathbf{S}^*_t \quad = \quad \psi(t+1,\mathbf{S}^*_{t+1})$$

Whilst this algorithm is guaranteed to give the most-likely metastate sequence $\mathbf{S}^*$ and requires only half the computation of calculating the forward and backward probabilities, the calculation of Viterbi sequences in the metastate space still scales as $\mathbf{O}(N^{2K}T)$ and thus suffers from the same problems as the EM approach. It is investigated in the empirical study primarily as a basis for evaluating the following approximate algorithm.

8.3.2 Chain Viterbi Metastate Sequences

[150] propose a more efficient scheme for *approximating* $\mathbf{S}^*$ when the K time series are *assumed* weakly coupled[3].

The Chain Viterbi algorithm iterates through each stream k in turn, finding the optimal sequence of hidden states through stream k given fixed values for the hidden states of the other streams. The state space is thus reduced to size N when doing the optimizations for stream k. The algorithm can be initialized by (for example) computing a Viterbi state sequence for each chain individually or by assuming a uniform segmentation of the observations for each stream. Iteration through all K streams continues until convergence, which is not necessarily to the optimal sequence $\mathbf{S}^*$ (see Appendix H for a counter-example).

More formally, letting $\mathbf{S} = \mathbf{S}_1,\ldots,\mathbf{S}_T$ denote the current approximation to the optimal metastate sequence, $\mathbf{S}_t = (\mathbf{S}^1_t,\ldots,\mathbf{S}^K_t)$ denote a metastate, $\mathbf{S}^{[k\rightarrow j]}_t$ denote a metastate in

[3] The approach is similar to the Iterative Conditional Modes algorithm for computing a MAP estimate of Markov Random Field parameters [13].

which the k-th component of $\mathbf{S}_t$ is changed to $j \in \theta_k$, and assignment $\mathbf{S}_t^k = j$ denote an update to $\mathbf{S}$ which sets k-th component of $\mathbf{S}_t$ to $j \in \theta_k$. The variable $\delta^k(t, j)$, defined for $t = 1, \ldots, T$ and $j \in \theta_k$, is introduced. For each k:

$$\delta^k(t, j, \mathbf{S}^{*/k}) \quad = \quad \max_{s_1^k, s_2^k, \ldots, s_{t-1}^k} p(\mathbf{O}_1, \ldots, \mathbf{O}_t, \mathbf{S}^{*/k}(t), s_1^k, s_2^k, \ldots, s_{t-1}^k, s_t^k = j|\lambda)$$

where $\mathbf{S}^{*/k}$ denotes some T-length sequence of states through each chain except the k-th and $\mathbf{S}^{*/k}(t)$ denotes the first t states visited in each chain k in those sequences.

Values of $\delta^k(t, j, \mathbf{S}^{*/k})$ can be efficiently calculated as follows.

Chain Viterbi Algorithm

Step 1: **Initialize** $\mathbf{S}$ to some metastate sequence $\mathbf{S}_1, \ldots, \mathbf{S}_T$.

Step 2: **Iterate**. Let $\mathbf{S}_{prev} = \mathbf{S}$. For each $k = 1, \ldots, K$, perform steps 2a through 2d:

2a. Initialize for each $j \in \theta_k$:

$$\begin{aligned} \delta^k(1, j, \mathbf{S}^{*/k}) &= P(\mathbf{S}_1^{[k \to j]}) \\ \psi^k(1, j, \mathbf{S}^{*/k}) &= 0 \end{aligned}$$

2b. For each $j \in \theta_k$ at $t = 2, \ldots, T$:

$$\begin{aligned} \delta^k(t, j, \mathbf{S}^{*/k}) &= \max_{i \in \theta_k} [\delta(t-1, \mathbf{S}_{t-1}^{[k \to i]}, \mathbf{S}^{*/k}) p(\mathbf{S}_t^{[k \to j]} | \mathbf{S}_{t-1}^{[k \to i]})] p(\mathbf{O}_t | \mathbf{S}_t^{[k \to j]}) \\ \psi^k(t, j, \mathbf{S}^{*/k}) &= \arg\max_{i \in \theta_k} \delta(t-1, \mathbf{S}_{t-1}^{[k \to i]}, \mathbf{S}^{*/k}) p(\mathbf{S}_t^{[k \to j]} | \mathbf{S}_{t-1}^{[k \to i]}) \end{aligned}$$

2c. Termination of kth iteration: calculate likelihood

$$p(\mathbf{O}, \mathbf{S}) \quad = \quad \max_{j \in \theta_k} \delta(T, j, \mathbf{S}^{*/k})$$

2d. Backtrace to update current best metastate sequence:

$$s_T^k \quad = \quad \arg\max_{j \in \theta_k} \delta(T, j, \mathbf{S}^{*/k})$$

and then for $t = T - 1, \ldots, 1$, update $\mathbf{S}$ with s_t^k where

$$s_t^k \quad = \quad \psi(t+1, s_{t+1}^k, \mathbf{S}^{*/k})$$

Step 3. **Termination**: if $\mathbf{S} = \mathbf{S}_{prev}$, then set $\mathbf{S}^* = \mathbf{S}$ and terminate. Else goto **Step 2**.

Softer versions of this iterative scheme are possible, using a subroutine where the state sequences corresponding to some subset of streams are fixed and a Viterbi-like decoding procedure applied to the remainder. This might be applicable in a multiband approach using many frequency subbands, where coupling may be assumed to be weak between bands which are not adjacent.

We note that the chainwise Viterbi method can be viewed as a variational method, in which the approximating distribution Q puts all its probability mass on a single metastate sequence[4]. In this view, the states of the metastate sequence are themselves the

[4] We thank Hagai Attias for this observation.

variational parameters and the chainwise Viterbi method simply optimizes these parameters. Thus the chainwise Viterbi method can also be viewed as optimizing a lower bound on the log-likelihood, namely the log-likelihood of one particular path.

In our experiments, the chainwise Viterbi method is incorporated into model estimation as follows. We first run the chainwise Viterbi algorithm to convergence, yielding a metastate sequence $\mathbf{S}^*$. This metastate sequence is then used in a Viterbi-training-like M-step. An alternative approach might be to update parameters after decoding each chain. No claims about the relative advantages of these approaches or their convergence properties will be made.

9

Experimental Evaluation

This chapter presents an empirical study of the factorial models and algorithms discussed in the previous two chapters. The study uses two small vocabulary speech tasks: an isolated word classification task and a continuous word recognition task. The issues addressed include:

- *comparison of a loosely-coupled model with more conventional speech models on a classification task*;
- *comparison of FHMM parameter reduction schemes on a classification task*;
- *comparison of exact and approximate decoding algorithms for a classification task*;
- *comparison of exact and approximate algorithms in estimation and decoding for a classification task*;
- *comparison of a loosely-coupled model with more conventional speech models on a recognition task*.

The representation of speech used in the classification and recognition tasks will be cepstra derived from frequency subbands (eg. [108, 163, 107]), rather than a more speculative articulatory or phonological representation. There is evidence that asynchrony exists between different frequency bands [108], making this a reasonable task for these initial studies. Whilst there is likely to be more asynchrony in articulatory or phonological representations, the exploration of such representations is beyond the scope of this work.

The following terminology is reviewed from Section 7.2; the illustrations there may also be helpful. The case where only FHMM transition distributions can be dependent upon metastates will be referred to as *transition-only coupled*, the case where only FHMM output distributions can be dependent upon metastates will be referred to as *observation-only coupled* and the general case where both FHMM observation and transition distributions may depend upon metastates is *fully coupled*.

9.1 Evaluation on a Letter Classification Task

9.1.1 Corpus

The classification experiments which follow use the OGI ISOLET database [27], which comprises wideband recordings of isolated utterances of single letters of the alphabet. Whilst far from the conversational speech motivating the research, use of a simple testbed such as ISOLET facilitates an initial feasibility study of novel models and algorithms without the additional complications introduced by continuous speech tasks. We use *Isolet1-4* (6240 utterances, 1 hour of speech) to train and the speaker-disjoint *Isolet5* (1560 utterances, 15 minutes of speech) to test. This training set is twice as large as that used by [27, 78]. All experiments train a single model for each letter, with no parameter tying *across* different models; this differs from the HMM-based system of [78] which utilizes parameter sharing across members of the E-set of letters and also uses an explicit silence model. For reference, performance of our baseline HMMs using a 39-dimensional observation vector of *full-band* cepstra (including 0th) with delta and acceleration coefficients is between 96.2% (3 state HMM) and 96.6% (10 state HMM) for this data set.

9.1.2 Procedure for Subband Cepstra Extraction

The extraction of subband cepstra proceeds as follows. 25ms windows of speech are Fourier-transformed and filtered through a bank of 20 overlapping, equally mel-spaced, filters using the HTK toolkit [180]. Filtering produces a vector of log spectral energies $E = [e_1, \ldots, e_{20}]$. A choice of V frequency subbands subdivides E into V subvectors E_v. A DCT D_v is applied to each E_v to yield a vector of cepstra $C_v = D_v E_v$ for subband v. Decreasing D_v row dimensionality effects cepstral truncation, reducing the dimensionality of C_v from that of E_v: a V-tuple $(\#_1, \ldots, \#_V)$ denotes the truncation scheme, where $\#_v$ indicates retention of cepstra $0, \ldots, \#_v - 1$ in subband v. Finally, observations for the v-th subband stream (o_t^v in our earlier notation) are formed by appending the appropriate delta and acceleration coefficients to C_v.

The experiments below use cepstra from both two and three frequency subbands. Observations for the two-stream experiments comprise cepstra from two subbands 0-2 and 2-8kHz with cepstral truncation (7,6), yielding a 39-dimensional combined observation vector $\mathbf{O}_t$. Observations for three-stream experiments comprise cepstra from three subbands 0-0.9, 0.8-2.7, 2.7-8kHz with cepstral truncation (5,4,4), again yielding a 39-dimensional combined observation vector $\mathbf{O}_t$.

9.1.3 Comparison: Factorial and Conventional Speech Models

The first set of experiments investigates the feasibility of using loosely-coupled models for speech modelling by comparing the classification performance of the MM-FHMM with that of more conventional speech models. Classification uses a maximum likelihood (ML) decision rule ie. utterance $\mathbf{O}$ is allocated to class $\mathbf{W}^*$ where $\mathbf{W}^* = \arg\max_{\mathbf{W}} p(\mathbf{O}|\mathbf{W})$. (This decision rule is equivalent to the Bayes' minimum error decision rule for this task since class priors are equal for the ISOLET test set, by test set definition). Performance is compared against two baselines. The first is a standard HMM-based system trained on the single observation stream that results from merging the observation streams corresponding to the (two- or three-) frequency subbands. However, since HMM and MM-FHMM-based classifiers are quite different in their use of free parameters, additional comparisons will be made against the *HTK multiple stream* and *independent streams* models. These "conventional" models will be configured to be comparable with the loosely-coupled models not only in terms of the total number of parameters but also in their usage of parameters. The MM-FHMM, HTK multiple stream and independent stream models differ in the degree of asynchrony allowed between streams. To reflect this, results will be ordered in terms of increasing potential asynchrony: the synchronous HTK multiple stream model will be followed by the loosely-coupled models and then the completely asynchronous independent streams model. Note that none of the HTK multiple stream, loosely-coupled or independent streams results utilise any form of exponent or

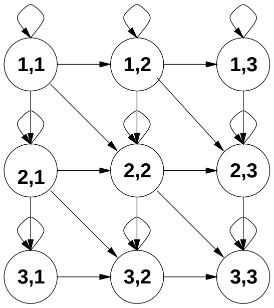


Figure 9.1 *Left-to-right Metastate Space Topology*

other stream weighting.

9.1.3.1 Experimental Setup

The implementations of FHMMs used in the experiments differs slightly from that in Chapter 7 due to the introduction of non-emitting entry and exit states, similar to those in the HTK Toolkit [180]. Appendix F details the minor changes to the reestimation equations.

MM-FHMMs that model K observation streams use K underlying chains. All Gaussians are full covariance, initialized using the global mean and covariance of the training set.

Several experiments will investigate the effect of introducing coupling between chains by making MM-FHMM transition probabilities dependent upon metastates. This introduces a difficulty, since the form of transition coupling dictated by the mixed-memory assumption of Eqn (7.4) introduces a limitation for speech modelling. The HMMs used in phone modelling for speech recognition are typically constrained *a priori* to have a left-to-right transition structure, but this is achievable under Eqn (7.4) only when ψ is the identity matrix I (ie. transitions in stream k depend only on the previous state in stream k). In this study, when we investigate coupling through transition probabilities in MM-FHMMs (achieved by setting $\psi \neq I$), we use a^{kl} matrices which are individually left-to-right. This will limit backwards transitions in the metastate space, but is not as strong as the standard left-to-right constraint. To clarify, Figure 9.1 shows the (arguably) desirable left-to-right metastate space topology. Since it is difficult to draw the full set of connections in the contrasting transition-limited MM-FHMM topology, Figure 9.2 instead shows the connections *removed* from an ergodic metastate space topology when the underlying cross-transition matrices are left-to-right in a two-stream, three state per stream, model. The latter figure illustrates that whilst the left-right constraint is not fully enforced, many of the otherwise possible backwards transitions are nevertheless prevented.

Model sets using cross-emission or cross-transition dependencies are initialized in stages: first, independent HMMs are trained for each stream as in an independent streams system; then, cross-stream dependencies are introduced gradually, with two training iterations between the addition of one cross-dependency per stream. For these model comparison experiments, training is continued until the gain in likelihood falls below a pre-specified threshold.

9.1.3.2 Experimental Results

The third and fourth columns of Table 9.1 give baseline percentage correct (*%C*) performance of standard *HMMs*, which model the single observation stream formed by concatenating observations in each of the individual (two- or three-) frequency subband

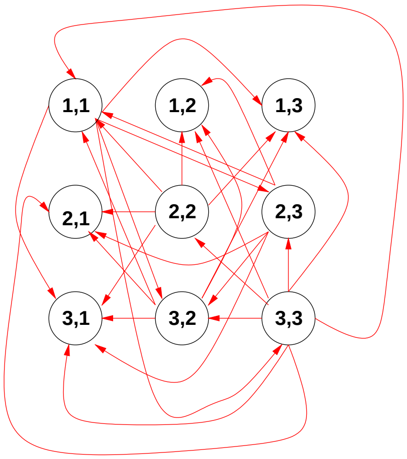


Figure 9.2 *Transitions Removed in Restricted Metastate Space Transition Topology*

Model (# states)	# Parameters	2 Frequency Subbands %C	3 Frequency Subbands %C
HMM (3)	2460	96.3	95.9
HMM (6)	4920	96.1	95.8
HMM (8)	6560	96.4	96.0
HMM (10)	8200	96.7	96.5

Table 9.1 *Results: HMM Baselines (observation vectors formed by concatenating observation streams derived from 2 or 3 frequency subbands)*

Model (states per stream)	# Parameters	%C
HTK multiple stream (3)	1326	94.2
MM-FHMM, transition probability metastate dependence (3)	1337	94.1
independent streams (3)	1329	93.9
HTK multiple stream (6)	2652	94.9
MM-FHMM, transition probability metastate dependence (6)	2672	95.0
independent streams (6)	2658	94.8
HTK multiple stream (8)	3536	95.4
MM-FHMM, transition probability metastate dependence (8)	3562	95.3
independent streams (8)	3544	95.8

Table 9.2 *Results: Transition-Only Coupled Models (2 observation streams, 2 chains)*

Model (states per stream)	# Parameters	%C
HTK multiple stream (3)	948	93.1
MM-FHMM, transition probability metastate dependence (3)	978	93.3
independent streams (3)	954	93.7
HTK multiple stream (6)	1896	94.7
MM-FHMM, transition probability metastate dependence (6)	1950	95.2
independent streams (6)	1908	94.4
HTK multiple stream (8)	2528	95.8
MM-FHMM, transition probability metastate dependence (8)	2598	95.2
independent streams (8)	2544	94.8

Table 9.3 *Results: Transition-Only Coupled Models (3 observation streams, 3 chains)*

Model (states per stream)	# Parameters	%C
HTK multiple stream (3)	2655	94.6
MM-FHMM, output and transition probability metastate-dependence (3)	2662	94.7
MM-FHMM, output probability metastate-dependence (3)	2654	94.9
independent streams (3)	2658	94.0
HTK multiple stream (6)	5310	96.2
MM-FHMM, output and transition probability metastate-dependence (6)	5320	95.8
MM-FHMM, output probability metastate-dependence (6)	5306	96.7
independent streams (6)	5316	95.3
HTK multiple stream (8)	7080	96.2
MM-FHMM, output and transition probability metastate-dependence (8)	7092	96.2
MM-FHMM, output probability metastate-dependence (8)	7074	96.0
independent streams (8)	7088	96.3

Table 9.4 *Results: Observation-Only and Fully Coupled Models (2 observation streams, 2 chains)*

Model (states per stream)	# Parameters	%C
HTK multiple stream (3)	2856	94.4
MM-FHMM, output and transition probability metastate-dependence (3)	2874	96.0
MM-FHMM, output probability metastate-dependence (3)	2850	95.2
independent streams (3)	2862	95.2
HTK multiple stream (6)	5712	96.8
MM-FHMM, output and transition probability metastate-dependence (6)	5736	96.2
MM-FHMM, output probability metastate-dependence (6)	5694	96.7
independent streams (6)	5724	95.8
HTK multiple stream (8)	7616	96.5
MM-FHMM, output and transition probability metastate-dependence (8)	7644	96.4
MM-FHMM, output probability metastate-dependence (8)	7590	96.2
independent streams (8)	7632	96.3

Table 9.5 *Results: Observation-Only and Fully Coupled Models (3 observation streams, 3 chains)*

Number of states N	HTK Multiple Stream with N states	Independent Stream with N states per chain	HMM with 3 states	HMM with 6 states
3	NO	NO	$p = 1.2 \times 10^{-4}$	$p = 1.5 \times 10^{-3}$
6	NO	NO	NO	NO
8	NO	NO	NO	NO

Table 9.6 *Significance Test Results: the results produced by the specified model and those from an N state, 2 stream, 2 chain MM-FHMM with transition probability metastate dependence are tested for differences significant at the $\alpha = 0.01$ level; p-values are specified where results differ significantly.*

cepstra streams. Tables 9.2 and 9.3 show the effects of coupling incorporated by making transition probabilities dependent upon metastates. As mentioned earlier, the models in each block of these tables are ordered in terms of allowable asynchrony between streams: the synchronous *HTK multiple stream* model is followed by a MM-FHMM using transition probabilities dependent upon metastates, which precedes the completely asynchronous *independent streams* model. The significance of differences in performance for models with comparable numbers of parameters can be analysed using the McNemar test [53]. Table 9.6 compares the significance of differences in results produced by the conventional models with those for the 2-stream MM-FHMM with transition probability metastate dependence. (These significance tests analyse the results for modelling cepstra derived from *two* frequency subbands; they correspond to Table 9.2 and to the results for baseline HMMs with similar numbers of parameters, as found in the third column of Table 9.1.) Table 9.7 presents a similar analysis, comparing differences in results produced by the conventional models with those for the 3-stream MM-FHMM with transition probability metastate dependence. (These significance tests analyse the results for modelling cepstra derived from *three* frequency subbands; they correspond to Table 9.3 and to the results for baseline HMMs with similar numbers of parameters, as found in the fourth column of Table 9.1.)

Table 9.4 considers two types of MM-FHMM coupling: first coupling introduced through observation probabilities only, where observation probabilities are made dependent upon metastates, and then systems coupled through both observation and transition probabilities. The table is again ordered using increasing allowable asynchrony. Each state in *HTK multiple stream* and *independent streams* models uses a two Gaussian mixture to model the data from a single stream. The number of observation-related parameters in these systems is thus comparable with the mixed-memory models using metastate-dependent observation probabilities, which use a single Gaussian to model each $b^{kl}(o_t^k|i^l)$ distribution. A similar set of comparisons are made for modelling three subband data in Table 9.5, where states in *HTK stream* and *independent streams* models use a three Gaussian mixture to model the data from a single stream. The significance of differences in performance for fully coupled models with comparable numbers of parameters can be analysed using the McNemar test [53]. Table 9.8 compares results produced by the conventional models with those for the 2-stream MM-FHMM with transition and output probability metastate dependence. (These significance tests analyse the results for modelling cepstra derived from *two* frequency subbands; they correspond to Table 9.4 and to the results for baseline HMMs with similar numbers of parameters, as found in the third column of Table 9.1.) Table 9.9 compares results produced by the conventional models with those for the 3-stream MM-FHMM with transition probability metastate dependence. (These significance tests analyse the results for modelling cepstra derived from *three* frequency subbands; they correspond to Table 9.5 and to the results for baseline HMMs with similar numbers of parameters, as found in the fourth column of Table 9.1.)

Some analysis of MM-FHMM behaviour was performed. The first question investigated

Number of states N	HTK Multiple Stream with N states	Independent Stream with N states per chain	HMM with 3 states
3	NO	NO	$p = 2.0 \times 10^{-5}$
6	NO	NO	NO
8	NO	NO	NO

Table 9.7 *Significance Test Results: the results produced by the specified model and those from an N state, 3 stream, 3 chain MM-FHMM with transition probability metastate dependence are tested for differences significant at the $\alpha = 0.01$ level; p-values are specified where results differ significantly.*

Number of states N	HTK Multiple Stream with N states + 2 Gaussians per state	Independent Stream with N states per chain + 2 Gaussians per state	MM-FHMM with only Observation Probabilities dependent on metastates, N states per chain	HMM with N states
3	NO	NO	NO	$p = 2.6 \times 10^{-3}$
6	NO	NO	NO	NO
8	NO	NO	NO	NO

Table 9.8 *Significance Test Results: the results produced by the specified model and those from an N state, 2 stream, 2 chain MM-FHMM with both output and transition probability metastate dependence are tested for differences significant at the $\alpha = 0.01$ level; p-values are specified where results differ significantly.*

Number of states N	HTK Multiple Stream with N states + 3 Gaussians per state	Independent Stream with N states per chain + 3 Gaussians per state	MM-FHMM with only Observation Probabilities dependent on metastates, N states per chain	HMM with N states
3	$p = 8.0 \times 10^{-4}$	NO	NO	NO
6	NO	NO	NO	NO
8	NO	NO	NO	NO

Table 9.9 *Significance Test Results: the results produced by the specified model and those from an N state, 3 stream, 3 chain MM-FHMM with both output and transition probability metastate dependence are tested for differences significant at the $\alpha = 0.01$ level; p-values are specified where results differ significantly.*

Model (# States)	2-Stream	3-Stream
Transition Metastate Dependence (3)	19% (0%)	34% (3%)
Observation Metastate Dependence (3)	22% (2%)	36% (1%)
Transition and Observation Metastate Dependence (3)	15% (1%)	22% (2%)
Transition Metastate Dependence (6)	33% (1%)	48% (5%)
Observation Metastate Dependence (6)	42% (2%)	51% (4%)
Transition and Observation Metastate Dependence (6)	38% (1%)	38% (3%)
Transition Metastate Dependence (8)	38% (5%)	54% (10%)
Observation Metastate Dependence (8)	46% (4%)	58% (11%)
Transition and Observation Metastate Dependence (8)	46% (3%)	45% (6%)

Table 9.10 *Results: Percentage "Asynchronous" Metastates in Training Set Viterbi Metastate Sequences (Percentage in brackets corresponds to metastates reflecting a higher degree of asynchrony ie. metastates I such that* $\max_{k,l(l\neq k)} |i^k - i^l| > 1$*). Percentages rounded to nearest integer.*

asked whether the allowable asynchrony between chains is used in modelling the ISOLET data. For each class, a Viterbi decoding of each training (or test) set utterance in that class using the *correct* MM-FHMM (ie. the MM-FHMM corresponding to that class) gives the optimal metastate sequence for each utterance in the training (or test) set under the correct model. The metastate sequences produced for the training (or test) set were analysed to determine the percentage of "asynchronous" metastates that occurred. ("Asynchronous" metastates for a two-stream system means metastates (i, j) where $i \neq j$ and for a three-stream system means metastates (i, j, k) where it is not the case that $i = j = k$). A typical set of results, calculated using the training set, is shown in Figure 9.10. The table shows that the potential asynchrony allowed in the models is used in modelling the training set. The proportion of asynchronous metastates increases as the size of the metastate space increases. The equivalent numbers calculated using test set alignments show a small 1-2% increase in the proportion of asynchronous metastates in the Viterbi metastate sequences for unseen data from the correct class. The number of asynchronous metastates used in modelling data from other classes is of course much higher, by between 10-25%. These and other results suggest that the amount of asynchrony used in MM-FHMMs with observation-only metastate dependence is slightly higher than in the MM-FHMMs with other types of metastate dependence, but the differences are small and may not be significant. The numbers in brackets in Figure 9.10 indicate the proportion of highly asynchronous metastates used, where this is defined to be metastates I such that $\max_{k,l(l\neq k)} |i^k - i^l| > 1$. As might be expected for clean dictated speech, metastates corresponding to extreme asynchrony are occupied only a small proportion of the time. The use of asynchronous metastates by phone class was also examined, but there were no immediately obvious differences in use of asynchrony.

Further analysis looked at posterior probabilities for the indicator variables, $p(x_t^k = l|\mathbf{O})$ and $p(y_t^k = l|\mathbf{O})$, and the frequency with which the most-likely value of x_t^k or y_t^k was not k. In two observation stream, fully coupled models, the transition-related indicator variables x_t^k associated with each chain k had a most-likely value $l \neq k$ for only 5-10% of the time; for the observation-related indicator variables, the most-likely value for the first chain (associated with the lowest frequency band) was $l \neq k$ 57-61% of the time and for the second chain 38-48% of the time. These percentages were a little higher, about 2%, for the transition-only and observation-only coupled models. In three stream systems, similar trends were observed. For each chain, the transition-related indicator variable had a most-likely value for self-prediction 90% or more of the time and for either of the remaining two chains of under 5% each, in both transition-only and fully coupled models. The observation-related indicator variables were more balanced in the distribution of most-likely values across chains in the observation-only and fully coupled models, with self-prediction occurring $40 - 60\%$ and the remaining chains $10 - 20\%$ each. The overall trend is for posteriors to indicate self-prediction in the case of transitions and for a more balanced indication of different chains in the case of observations.

Analysis also examined whether the right-to-left (backwards) transitions which are allowed in the restricted left-to-right MM-FHMM topology of Figure 9.2 are used frequently in modelling ISOLET. A Viterbi metastate sequence was found for each training utterance under the model for the corresponding utterance class and the percentage of backwards transitions analysed. The same procedure was repeated for each test utterance. Training set analysis showed that across two- and three- stream transition-only coupled models, with varying numbers of states, the percentage of backwards transitions was always below 1%, and a similar comparison using fully coupled models found the percentage of backwards transitions was below 0.5%. The increase in backwards transitions on test data from the same class was very slight. The small percentages of backwards transitions suggest the MM-FHMMs learn something of the left-to-right nature of the stationary regions in the speech signal. Whilst this is of interest, it does not affect the underlying concern that backwards transitions in competing models could lead to reduced discrimination on tasks more difficult than ISOLET.

9.1.3.3 Conclusions

The overall results show that in most cases performance of the factorial models does not differ significantly from the more conventional speech models on the task of frequency subband modelling. However, the potential advantages of the new models may not be evident on this simple speech modelling task and we interpret attainment of comparable performance as sufficient to conclude that factorial models appear a feasible alternative for speech modelling tasks and merit further investigation.

9.1.4 Comparison: Parameter Reduction Schemes

The previous section addressed the basic question of whether FHMMs, in particular those using the mixed-memory parameter reduction scheme, would scale to a speech modelling task. This section compares the MM-FHMM with the more data-driven PT-FHMM parameter reduction scheme, which has some potential advantages for speech modelling. The task is the same as in the previous section: ISOLET classification.

9.1.4.1 Experimental Setup

The implementations of FHMMs used in the experiments differs slightly from that in Chapter 7 due to the introduction of non-emitting entry and exit states. Appendix F details the minor changes to the reestimation equations.

All MM-FHMMs and PT-FHMMs investigated in this section use two observation streams and two underlying chains. The observation streams are again cepstra derived from two frequency subbands, as discussed in Section 9.1.2. MM-FHMMs are trained as in the previous section and the procedure for training PT-FHMMs is as follows.

Obtaining Sufficient Statistics for PT-FHMM Clustering Sufficient statistics can in principle be obtained using the following general procedure. Independent streams HMMs with single Gaussian observation distributions are trained to model each observation stream. These HMMS are then combined into the equivalent metastate space model (as in the extended-PMC model of Section 6.1.5). The resulting combined model is trained for two iterations. The metastate alignment produced in the second iteration and the resulting updated model parameters provide the sufficient statistics for clustering. Two

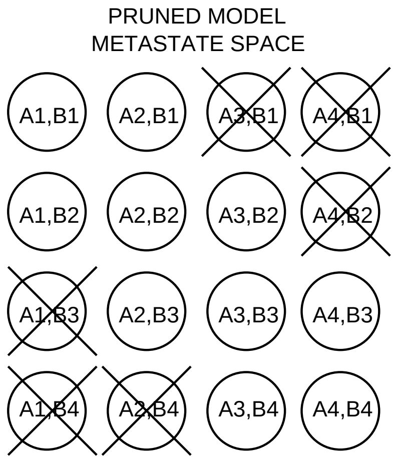


Figure 9.3 *Pruning metastate space to maximum asynchrony of one state per chain*

issues arose in preliminary experiments. Firstly, data sparsity led to numerical issues with full covariance matrix inversion in the untied metastate space model. To resolve this issue, all independent streams HMMs, the untied metastate space model and the clustering procedure were set to use diagonal covariances; the models resulting from clustering are converted to use full covariance matrices before further training is performed. Secondly, sufficient statistics adequate for clustering could not (in most cases) be obtained for each possible metastate within the fully expanded metastate model. The occupancies of metastates corresponding to extreme asynchrony between chains were close to zero: specifically, metastates $I = (i^1, \ldots, i^K)$ for which $max_{k,l(l \neq k)}(i^k - i^l) > 1$. This is not surprising, particularly given the low usage of extreme metastates in the MM-FHMM experiments of the previous section: it seems unlikely that this degree of asynchrony exists between frequency subbands for dictated speech data and this result suggests full metastate space models may not be appropriate for modelling frequency subband representations. To resolve this issue, pruning is applied to remove metastates within the metastate space prior to the two iterations of reestimation that yield statistics for clustering. Allowing a maximum asynchrony of one state, as in Figure 9.3, gave adequate statistics.

Initial Equivalence Classes for PT-FHMM Clustering The equivalence classes used to start the clustering procedure tie distributions such that the initial, pre-clustering PT-FHMM is similar to an independent streams model (but excluding metastates which have been “pruned”). Splitting during clustering therefore adds dependencies into this independent streams model when doing so leads to a “sufficient” gain in auxiliary function.

Initializing PT-FHMM Hierarchical Partitioning Procedure At each partitioning step for observation-related distributions, a single equivalence class with centroid $(\mu_{C^{obs,k}}, \Sigma_{C^{obs,k}})$ is partitioned into two new classes with centroids $(\mu_{C_1^{obs,k}}, \Sigma_{C_1^{obs,k}})$ and $(\mu_{C_2^{obs,k}}, \Sigma_{C_2^{obs,k}})$. These are initialized such that $(\mu_{C_1^{obs,k}}, \Sigma_{C_1^{obs,k}}) = (\mu_{C^{obs,k}}, \Sigma_{C^{obs,k}})$ and $(\mu_{C_2^{obs,k}}, \Sigma_{C_2^{obs,k}})$ is a small shift of $(\mu_{C^{obs,k}}, \Sigma_{C^{obs,k}})$ towards a single element in the parent equivalence class. At each partitioning step for transition-related distributions, a single equivalence class with centroid $P(j^k|C^{trans,k})$ is partitioned into two classes with centroids $P(j^k|C_1^{trans,k})$ and $P(j^k|C_2^{trans,k})$. These are initialized such that $P(j^k|C_1^{trans,k}) = P(j^k|C^{trans,k})$ and $P(j^k|C_2^{trans,k})$ reverses the first two non-zero transition probabilities in the original distribution $P(j^k|C^{trans,k})$.

Training PT-FHMMs The single Gaussian, full covariance PT-FHMMs that result from clustering are trained for four EM iterations before testing.

9.1.4.2 Experimental Results

There are several thresholds in the hierarchical, PT-FHMM clustering procedure. The minimum number of frames per leaf occupancy threshold was set to a fixed value of 150 observations for all experiments. The number of equivalence classes allocated to transition and observation distributions was controlled through varying the thresholds associated with gains in auxiliary function (Section 7.4.1). Note that the number of parameters allocated to the PT-FHMM models may differ by class so where parameter totals are stated, these represent an average taken over all 26 models.

Results are presented in two forms. Firstly, the performance of PT-FHMMs is compared directly with MM-FHMMs in Table 9.11. (The MM-FHMM results are repeated from the previous section to avoid the need for cross-referencing.) Both types of model use six emitting states per chain; metastates are then pruned from the resulting metastate space in the PT-FHMM prior to clustering. Four sets of results are shown. The first is an independent streams model, with no metastate dependence in the observation- or transition-distributions. The difference in results when the independent streams model is created as a special case of the MM-FHMM or of the PT-FHMM is attributed to differences in model initialization. The three remaining boxes compare the PT-FHMM with MM-FHMMs for the three varieties of coupling: observation-only metastate dependencies, transition-only metastate dependencies and both observation- and transition- metastate dependencies. The clustering thresholds for the PT-FHMM were set to give similar allocation of parameters to both observation and transition distributions to enable valid comparisons. The table shows similar results for both types of model and the McNemar test [53] confirms this by finding that differences in performance for PT-FHMM and MM-FHMM systems within the same table cells were not significant at the $\alpha = 0.01$ level. Similar results were observed with three state per chain models.

The effect of varying the number and type of model parameters in the PT-FHMMs is shown in Figure 9.4. The graphs show the effect of increasing the number of observation and/or transition parameters from a structure corresponding to a two chain independent streams model with pruned metastate space (which has 2646 observation parameters and 12 transition parameters).

Table 9.12 compares the usage of asynchronous metastates in Viterbi metastate sequences for each training set utterance under the correct class PT-FHMM or MM-FHMM. (The MM-FHMM figures are repeated from the previous section to avoid the need for cross-referencing.) In both cases, the proportion of asynchronous states used in the observation-only coupled models is higher than in the case of transition-only coupled models, perhaps because coupling through the transition probabilities is relatively weak when applied in combination with high-dimensional observations in each stream. As with the MM-FHMM, the proportion of asynchronous metastate sequences used in PT-FHMMs to model data from the corresponding class does not vary greatly between training and test data. It is interesting that despite the larger, unpruned metastate space of the MM-FHMM, the usage of asynchronous metastates is the same or higher in the PT-FHMM.

The PT-FHMM model sets were analysed to determine the relative number of parameters allocated to transition distributions in the two chains and for observation distributions for the two observation streams, for given values of clustering thresholds. Regardless of clustering threshold, the number of parameters allocated to transition distributions was almost identical. For high observation clustering thresholds, more parameters were allocated to observation distributions for the stream corresponding to the lower frequency band, but at low thresholds the allocation became more balanced.

	Observation Metastate Dependence=NO	Observation Metastate Dependence=YES
Transition Metastate Dependence=NO	MM-FHMM # Paras = 2658 (# trans = 12,# obs = 2646) %C=94.8 vs PT-FHMM # Paras = 2658 (# trans = 12,# obs = 2646) %C=95.3%	MM-FHMM # Paras = 5306 (# trans = 12, # obs = 5294) %C=96.7 vs PT-FHMM # Paras = 5399 (# trans = 12, # obs = 5387) %C=96.4%
Transition Metastate Dependence=YES	MM-FHMM # Paras = 2672 (# trans = 26, # obs = 2646) %C=95.0 vs PT-FHMM # Paras = 2670 (# trans = 24,# obs = 2646) %C=95.4	MM-FHMM # Paras = 5320 (# trans = 26, #obs = 5294) %C=95.8 vs PT-FHMM # Paras = 5411 (# trans = 24, # obs = 5387) %C=96.3

Table 9.11 *Results: Comparison of MM-FHMM and PT-FHMM (2 observation streams, 2 chains, 6 states per chain)*

Model (# States)	MM-FHMM	PT-FHMM
Transition Metastate Dependence (6)	33%	31%
Observation Metastate Dependence (6)	42%	48 %
Transition and Observation Metastate Dependence (6)	38%	48%

Table 9.12 *Percentage "Asynchronous" Metastates in Training Set Viterbi Metastate Sequences for MM-FHMM and PT-FHMM with 2 observation streams, 2 chains, 6 states per chain*

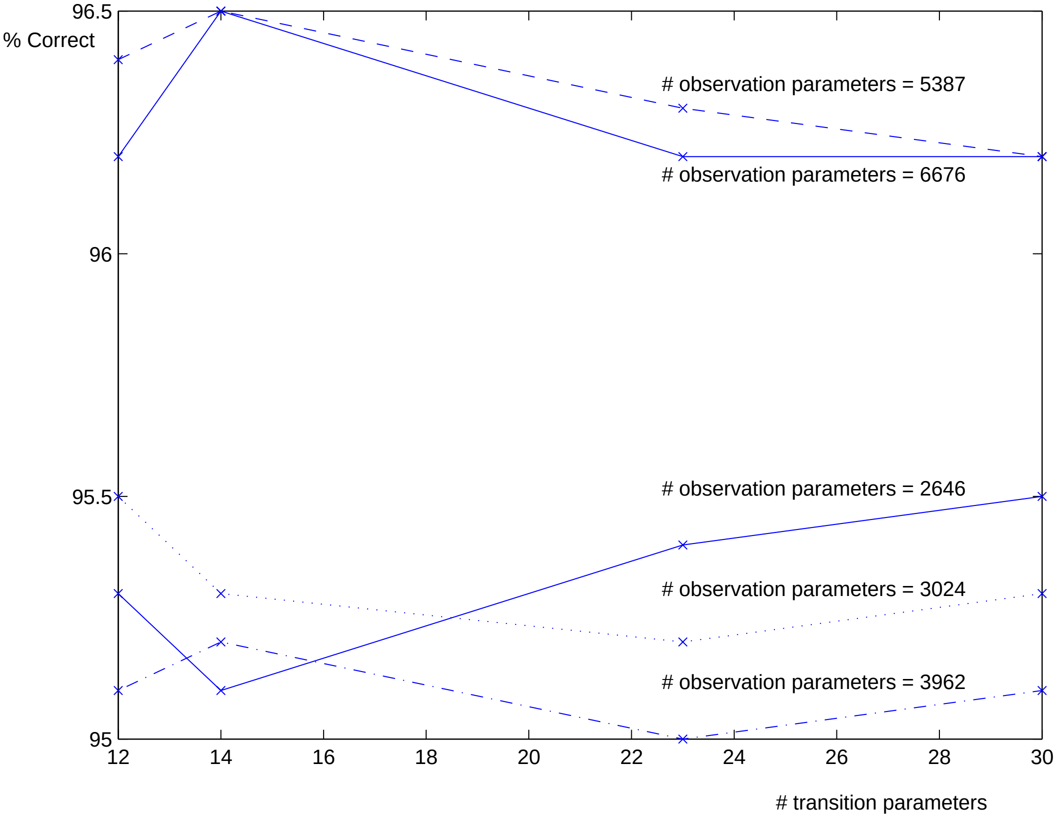


Figure 9.4 *Effect of varying number of observation and transition parameters in PT-FHMM*

9.1.4.3 Conclusions

The classification performance of the PT-FHMM was compared with the MM-FHMM for different coupling structures and differences in performance were not found to be significant. Since the empirical results are comparable and the PT-FHMM has some potential advantages for speech modelling (namely, the ability to enforce a left-to-right metastate space topology, and because the procedure of collecting sufficient statistics for clustering gives an indication of the maximum degree of asynchrony that exists present between streams), the PT-FHMM will be the model of choice for the continuous digit recognition experiments in Section 9.2.

It is interesting to note that the PT-FHMM results in isolation show increased coupling through additional transition-related parameters has little effect on classification performance for the ISOLET task. This is consistent with the standard finding for more conventional (non-factorial) HMM-based systems.

9.1.5 Comparison: Approximate Decoding Schemes

The previous section investigated classification using EM-trained loosely-coupled models of various forms and a ML decision rule. This section considers only models coupled through the *observation* probabilities, and considers the quality of the likelihood approximations and associated classification performance given by Chain Viterbi metastate sequences and by the Mean-Field variational lower bound.

9.1.5.1 Experimental Setup

Each experiment uses the *same*, fixed set of observation-only coupled models, which were trained using the EM algorithm.

Chain Viterbi Initialization: two schemes were investigated for specifying an initial metastate sequence. The first used a uniform segmentation of the observations in stream k against states in chain k; the second used the segmentation obtained by doing a Viterbi decoding of stream k observations using the chain k parameters only (for each k). Preliminary experiments found initialization had little effect on results; results below are for the per-chain Viterbi initialization.

Mean-Field Initialization and Thresholds: two initializations of the $Q_t^{Sk}(j)$ distributions were investigated. Initial per-stream state sequences were either obtained using the *uniform* or the *per-chain Viterbi decoding* schemes as described for the Chain Viterbi initializations; each $Q_t^{Sk}(j)$ distribution was then initialized with a soft version of this distribution, assigning mass 0.8 to the state occupied in the Viterbi or uniform segmentation, and distributing mass equally amongst the remaining states. $Q_t^{Yk}(l)$ distributions were initialized as a uniform distribution. Preliminary experiments found the per-chain Viterbi initialization gave considerably better results and was used to produce the results below. The naive gradient-descent implementation also requires a stepsize: a brute force search over a range of values was used and the results below correspond to the stepsize yielding the highest value for the lower-bound on test set likelihood. (We emphasize we do *not* use the stepsize giving the best *classification performance*, since this would constitute cheating).

9.1.5.2 Experimental Results

Figures 9.5,9.6 and 9.7 compare total test set likelihood under each class A-Z with the values obtained from Viterbi and Chain Viterbi approximations and from the Mean-Field Variational lower bound when modelling two subband cepstral streams. Figures 9.8, 9.9 and 9.10 make the same comparison for the case of modelling three subband cepstral streams. Tables 9.13 and 9.14 compare classification % Correct (%C) performance when using these approximations with a ML decision rule. The tables also give results of tests for significant differences between full-likelihood classification and the approximate algorithms using the McNemar test [53].

states	Full	Viterbi	Chain	Mean
per	Likelihood		Viterbi	Field
stream	%C	%C	%C	%C
3	94.9	94.8	95.0	91.0*($p = 0.0$)
6	95.3	95.4	95.3	95.1
8	96.0	96.0	96.0	96.0

Table 9.13 *Results: Decoding Schemes for 2 observation stream, 2 chain models (*=significantly different from the Full Likelihood case at $\alpha = 0.01$ level, with corresponding p value in brackets)*

9.1.5.3 Conclusions

The graphs show that the Chain Viterbi likelihood approximation is considerably closer to the exact likelihood than the Mean-Field Variational lower bound. In particular, it is close to the Viterbi likelihood, suggesting that the initialization used rarely leads to suboptimal local maxima. As this would suggest, classification performance using the Chain Viterbi algorithm is in almost all cases not significantly different to full-likelihood classification. The Mean-Field Variational Approximation is less successful. This is presumably

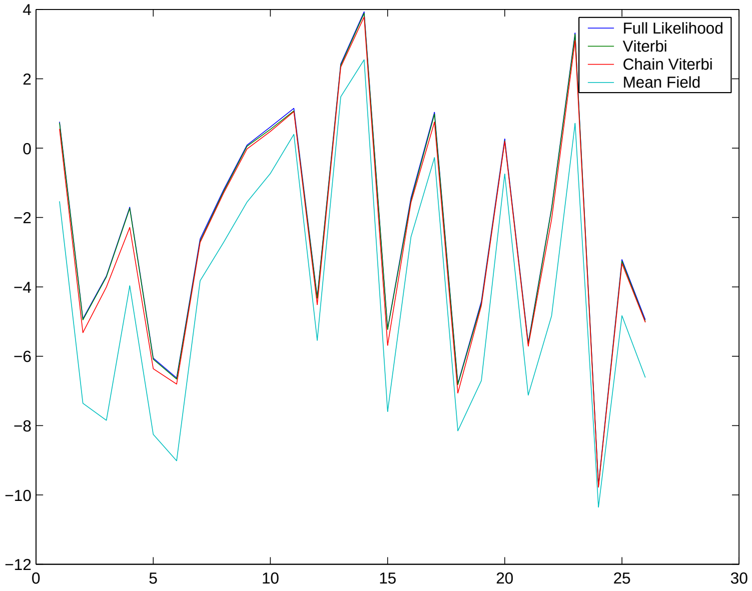


Figure 9.5 *Approximations to Test-Set Likelihoods for 2 streams, 2 chains, 3 state models (X-axis corresponds to classes A-Z)*

states per stream	Full Likelihood %C	Viterbi %C	Chain Viterbi %C	Mean Field %C
3	94.9	95.0	96.2	91.7*($p = 0$)
6	96.4	96.3	95.0	95.2*($p = 9.4 \times 10^{-3}$)
8	96.4	96.3	96.2	95.3*($p = 7.6 \times 10^{-3}$)

Table 9.14 *Results: Decoding Schemes for 3 observation stream, 3 chain models (*=significantly different from the Full Likelihood case at $\alpha = 0.01$ level, with corresponding p value in brackets)*

due to the extreme nature of the mean-field assumption. More structured variational approximations are possible, but will also be more computationally intensive. Since the Chain Viterbi algorithm gives good approximations to likelihood, is relatively insensitive to initialization and does not involve heuristic parameters such as step sizes, it would be our algorithm of choice for future work.

9.1.6 Comparison: Approximate Training Schemes

This subsection compares the classification performance of observation-only coupled MM-FHMMs trained and tested using *matched* (exact or approximate) algorithms ie. EM training is used with full likelihood (FL) classification, Chain Viterbi training is used with a Chain Viterbi approximation in classification etc. Viterbi Training and Decoding results are included as a further baseline for comparison.

9.1.6.1 Experimental Setup

All training algorithms stop one iteration after the gain in data likelihood or the variational lower bound drops below 1%.

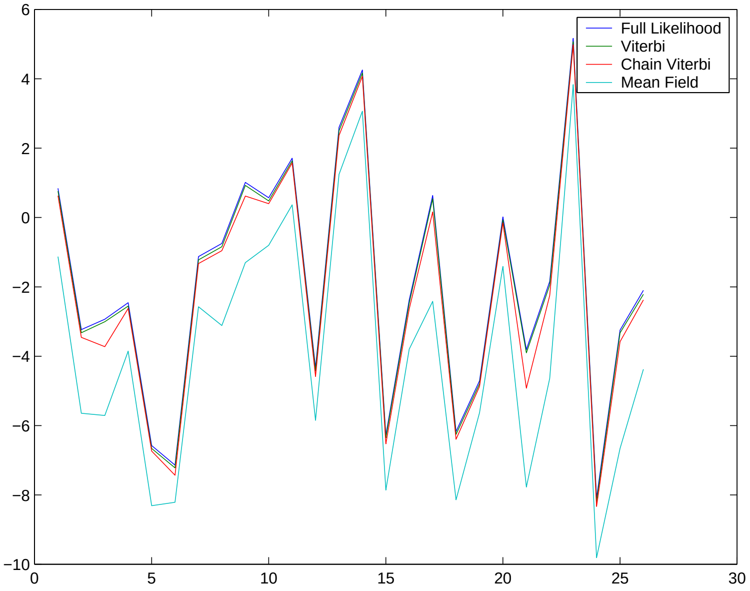


Figure 9.6 *Approximations to Test-Set Likelihoods for 2 streams, 2 chains, 6 state models (X-axis corresponds to classes A-Z)*

Chain Viterbi Initialization: the initial metastate sequence was obtained by doing a Viterbi decoding of stream k observations using the chain k parameters only, since this proved a useful initialization in the decoding-only experiments of the previous section.

Mean-Field Initialization and Thresholds: initial per-stream state sequences were obtained using the *per-chain Viterbi decoding* schemes as in Chain Viterbi initialization; each $Q_t^{Sk}(j)$ distribution was then initialized with a soft version of this distribution, assigning mass 0.8 to the state occupied in the Viterbi or uniform segmentation, and distributing mass equally amongst the remaining states. $Q_t^{Yk}(l)$ distributions were initialized as a uniform distribution. The gradient-descent stepsize used in training and decoding was fixed to the value that was most effective in the decoding-only experiments.

9.1.6.2 Experimental Results

Tables 9.15 and 9.16 compare classification % Correct (%C) performance obtained using the exact and approximate schemes with a ML decision rule. The tables also give results of tests for significant differences between exact likelihood-based training and classification and the approximate algorithms, using the McNemar test [53].

states per stream	Full Likelihood %C	Viterbi %C	Chain Viterbi %C	Mean Field %C
3	94.9	94.9	94.2*($p = 1.9 \times 10^{-2}$)	93.9
6	95.3	95.4	95.2	95.0
8	96.0	95.8	95.9	96.0

Table 9.15 *Results: Matched Training/Decoding Schemes for 2 observation stream, 2 chain models (*=significantly different from the EM-trained, Full Likelihood classification case at $\alpha = 0.01$ level, with corresponding p value in brackets)*

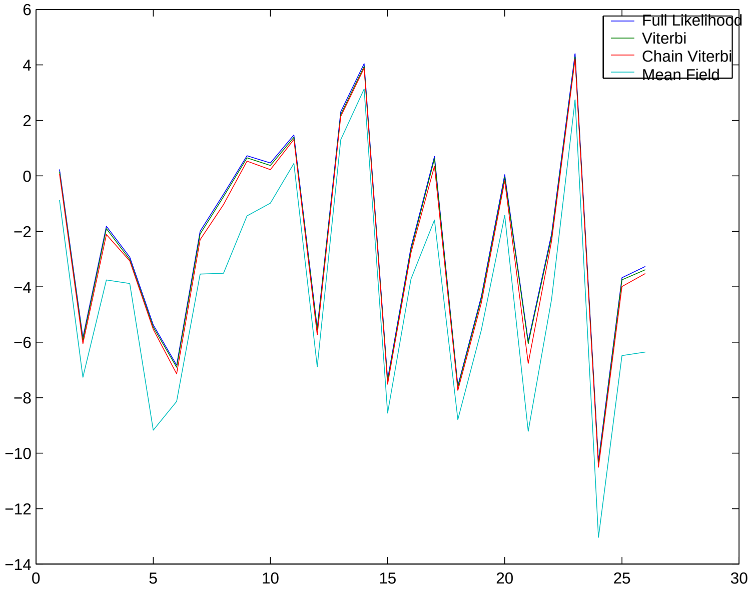


Figure 9.7 *Approximations to Test-Set Likelihoods for 2 streams, 2 chains, 8 state models (X-axis corresponds to classes A-Z)*

states per stream	Full Likelihood %C	Viterbi %C	Chain Viterbi %C	Mean Field %C
3	94.9	95.0	94.9	92.5*$(p = 4.0 \times 10^{-6})$
6	96.4	95.8	95.6	95.3
8	96.4	96.3	96.1	95.7

Table 9.16 *Results: Matched Training/Decoding Schemes for 3 observation stream, 3 chain models (*=significantly different from the EM-trained, Full Likelihood classification case at* $\alpha = 0.01$ *level, with corresponding* p *value in brackets)*

9.1.6.3 Conclusions

Classification performance achieved using the approximate algorithms is not significantly different to that achieved using the exact scheme in most cases. On average, the EM, Viterbi and Chain Viterbi algorithms all take a similar number of iterations to fall below the relative change training termination threshold; the Mean-Field scheme takes fewer. Despite this, our current implementation of the Chain Viterbi scheme has proven more efficient than our naive gradient-descent-based implementation of the mean-field approximation. Mean field equations can often be solved using fixed-point iterative updates that converge more quickly[1] but it seems unlikely that this would have much effect on the final results. Also, whether using gradient descent or fixed-point iteration, the Mean-Field scheme is more sensitive to initialization than the Chain Viterbi approach. Therefore, for reasons much the same as in Section 9.1.5.3, our algorithm of choice for use in future work would again be the Chain Viterbi algorithm.

[1] The author thanks Hagai Attias for this observation.

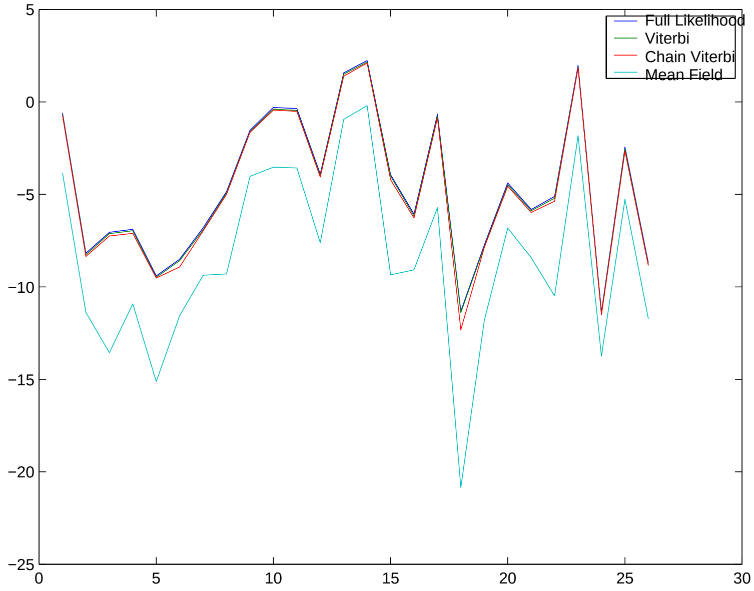


Figure 9.8 *Approximations to Test-Set Likelihoods for 3 streams, 3 chains, 3 state models (X-axis corresponds to classes A-Z)*

9.2 Evaluation on a Continuous Digit Recognition Task

9.2.1 Corpus

TI Digits is a speech corpus collected at Texas Instruments (TI) for the purpose of "designing and evaluating algorithms for speaker-independent recognition of connected digit sequences". The digit sequences are made up of the digits `ZERO`, `OH`, `ONE`, `TWO`, `THREE`, `FOUR`, `FIVE`, `SIX`, `SEVEN`, `EIGHT` and `NINE`. The (approximately[2]) 77 digit sequences spoken by each speaker can be broken down as: 22 isolated digits (2 productions of each of 11 digits), 11 2-digit sequences, 11 3-digit sequences, 11 4-digit sequences, 11 5-digit sequences, 11 6-digit sequences and 11 7-digit sequences. The corpus is described in detail in [98]. Experiments below use only the *men* and *women* subsets within the pre-defined training and test sets; no data from the *boys* or *girls* subsets is used. The training set comprises 8623 utterances (about 4 hours 15 mins of data); the test set 8700 utterances (about 4 hours 15 mins of data).

9.2.2 Procedure for Subband Cepstra Extraction

The extraction of subband cepstra proceeds as follows. 25ms windows of speech are Fourier-transformed and filtered through a bank of 24 overlapping, equally mel-spaced, filters using the HTK toolkit [180]. Filtering produces a vector of log spectral energies $E = [e_1, \ldots, e_{24}]$. A choice of V frequency subbands subdivides E into V subvectors E_v. A DCT D_v is applied to each E_v to yield a vector of cepstra $C_v = D_v E_v$ for subband v. Decreasing D_v row dimensionality effects cepstral truncation, reducing the dimensionality of C_v from that of E_v: a V-tuple $(\#_1, \ldots, \#_V)$ denotes the truncation scheme, where $\#_v$ indicates retention of cepstra $0, \ldots, \#_v - 1$ in subband v. Finally, observations for the v-th subband stream (o_t^v in our earlier notation) are formed by appending the appropriate delta and acceleration coefficients to C_v. The experiments below use observations

[2] Some utterances were removed from the corpus because they "contained egregious speaking errors" [98].

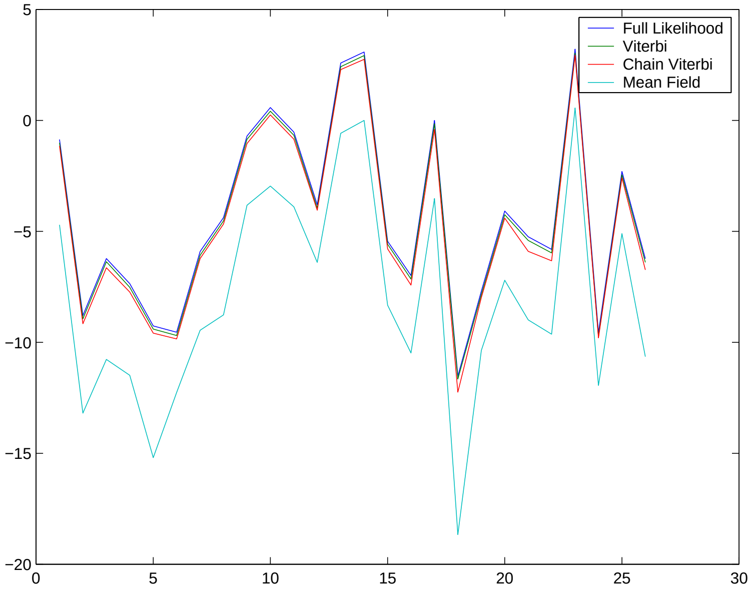


Figure 9.9 *Approximations to Test-Set Likelihoods for 3 streams, 3 chains, 6 state models (X-axis corresponds to classes A-Z)*

comprising cepstra from two subbands 0-2 and 2-10kHz, with cepstral truncation (7,6), yielding a 39-dimensional combined observation vector $\mathbf{O}_t$. (Experiments with alternative frequency subbands and truncation schemes yielded similar results.)

9.2.3 Comparison: Factorial and Conventional Speech Models

This section extends the PT-FHMM to a continuous digit recognition task. Performance will be compared with conventional speech models and also with the extended-PMC model (described in Section 6.1.5).

9.2.3.1 Experimental Setup

Model Topologies A single model is trained for each digit, with no parameter tying across models in different classes. In models using multiple chains (ie. multiband and PT-FHMM models), this means chains are synchronized at the end of each digit. (For a discussion of multiband synchronization issues, the reader is referred to [108].) All digit models have six emitting states per chain, following [31, 104]. Silence is modelled by a single state HMM. Unless otherwise stated, each state in an HMM or each $p(o_t^k|J)$ distribution in a PT-FHMM is modelled by a single Gaussian. All Gaussians use diagonal covariance matrices. Training uses an exact EM algorithm and testing uses a Viterbi approximation, following the standard setup for most conventional continuous speech recognition systems.

Silence Modelling During training, sentence initial and final silence is mandatory. A preliminary set of HMMs was trained on transcriptions with mandatory word final silence; the resulting models were then used to refine the training transcriptions using a forced alignment procedure in which word final silence was optional (as described in [180]). During recognition, sentence initial, sentence final and word final silence are all optional.

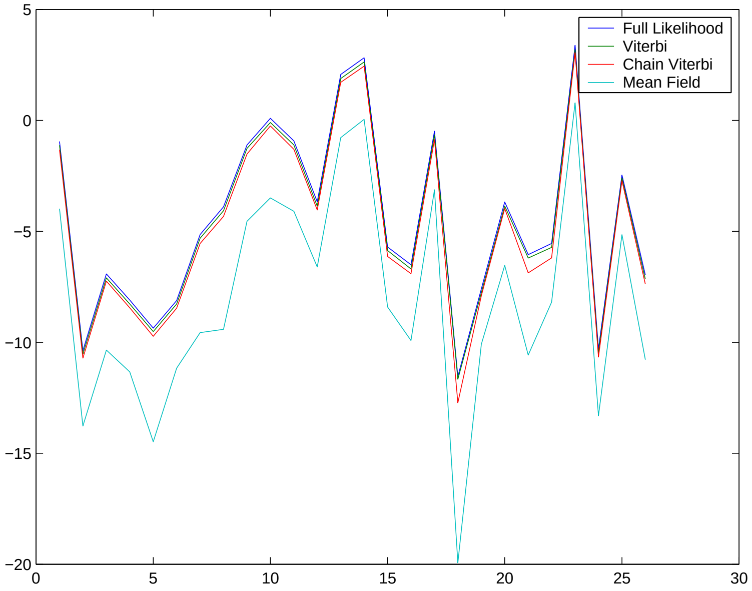


Figure 9.10 *Approximations to Test-Set Likelihoods for 3 streams, 3 chains, 8 state models (X-axis corresponds to classes A-Z)*

Training HMMs HMMs are trained following the standard HTK recipe [180]. Single Gaussian HMMs are initialized using a flat start (ie. models are initialized using the global mean and covariance of the training data) before recognition. These are trained until the relative change in likelihood drops below 10%, at which point one final parameter update step is performed. HMMs using two Gaussian mixtures are booted from the final single Gaussian HMMs using the HTK mixture splitting procedure and then trained using the same relative likelihood change criterion for termination. HMMs using four Gaussian mixtures are booted from the final HMMs with two Gaussian mixtures using the same HTK mixture splitting procedure and trained using the same relative likelihood change criterion for termination.

Training Multiband Models Single Gaussian independent streams models are initialized using the global mean and covariance of the training data and then trained until the relative change in likelihood drops below 10%, at which point one final parameter update step is performed. Synchronization points are *not* used during training: the models for each stream are completely independent[3]. The independent streams models for each digit are then combined into the equivalent metastate space model, as in the extended-PMC approach, before testing; this enforces stream synchronization at digit boundaries.

Training Extended-PMC Models Two variants on the Extended-PMC scheme are investigated. Firstly, an extended-PMC model with a full, unpruned metastate space is initialized using the parameters of the final multiband system trained as above. Secondly, an extended-PMC model with a pruned metastate space model comparable to that of the PT-FHMM is initialized using the multiband system parameters. Transition probabil-

[3] Incorporating synchronization points in training for the case of single Gaussian observation densities led to slightly improved performance in the single Gaussian case, but due to software constraints this training scheme could not be used for systems with Gaussian mixture observation densities. Therefore training without synchronization points was used throughout.

ities in the these extended-PMC models are initialized by appropriately normalizing the following quantities:

$$\tilde{p}(J|I) = \prod_{k=1}^{K} P(j^k|i^k)$$

The transition probabilities of the extended-PMC models are then reestimated; the termination criterion is again based on relative likelihood change.

Obtaining Sufficient Statistics for PT-FHMM Clustering In preliminary experiments, a set of metastate space models was initialized using the parameters of the final set of multiband models. In principle, this model set could then be reestimated for two iterations to obtain the sufficient statistics for clustering. However, as in the ISOLET experiments, it was difficult to obtain sufficient statistics for each possible metastate within a fully expanded metastate model. The same solution, metastate pruning similar to Figure 9.3, was used to restrict the degree of asynchrony to a maximum of one state prior to the two iterations of reestimation that yield statistics for clustering. The asynchrony limit of one state was necessary to obtain adequate statistics for clustering.

Initial Equivalence Classes for PT-FHMM Clustering The equivalence classes used to start the clustering procedure correspond to a tying structure making each PT-FHMM equivalent to a multiband model with stream synchronization at word boundaries (but excluding metastates which have been "pruned".) Splitting during clustering therefore adds dependencies into the multiband model when doing so leads to a "sufficient" gain in auxiliary function.

Training PT-FHMMs The single Gaussian, diagonal covariance models that result from clustering are trained until the relative change in likelihood drops below 10%, at which point one final parameter update step is performed.

Stream weighting None of the models utilise any form of exponent or other stream weighting.

Recognition Network The recognition network is shown in Figure 9.11. An appropriate insertion penalty was determined in preliminary experiments and held fixed throughout the experiments below.

9.2.3.2 Experimental Results

Results for the task of modelling cepstra derived from the full frequency band are presented in the *Fullband* column of Table 9.17. The *2 subbands* column of Table 9.17 presents HMM baseline results for the task of interest here, where observation vectors are formed by concatenating cepstra from two frequency subbands. Comparison of the results in the two columns finds a degradation in performance when moving from the fullband to two frequency subband representation of speech in the single Gaussian case; this also occurred when investigating other (two or three) frequency subband partitions. The differences are much smaller when using mixture of Gaussian output distributions. Although the degradation in the single Gaussian case suggests that use of frequency subband representations of speech is not always desirable, it is not directly relevant to this

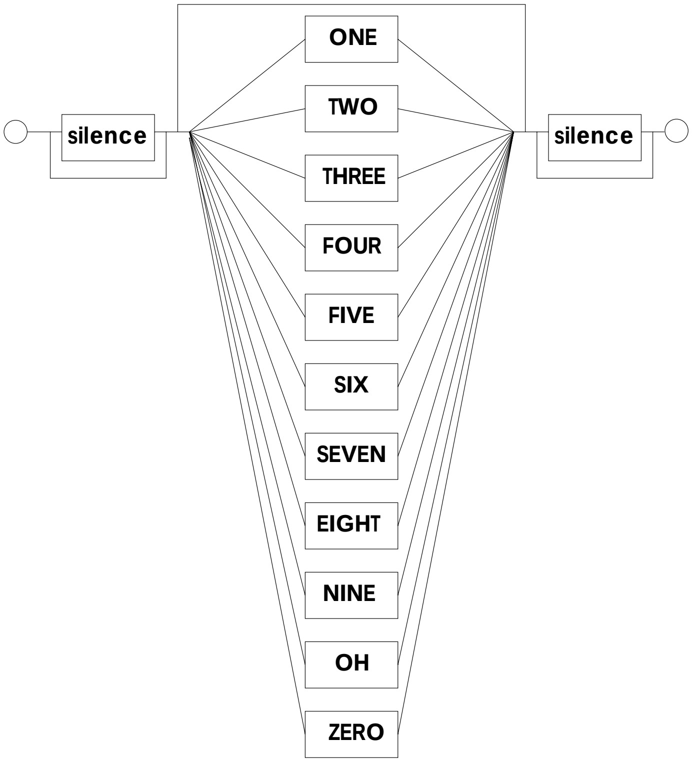


Figure 9.11 *TI Digits Recognition Network*

# Mixture Components	# Parameters Per Model	Fullband % Acc	2 subbands % Acc
1	474	95.2	94.7
2	948	96.9	96.9
3	1422	97.3	97.4

Table 9.17 *Results: HMM Baselines (observation vectors are (1) cepstra derived from the full frequency band (2) concatenation of cepstra derived from 2 frequency subbands)*

investigation since all further comparisons will refer to the task of modelling the same, fixed, two subband representation of speech.

For reference, the performance of an HMM model set trained to model data derived only from the first frequency band (0-2kHz) is 93.0% (single Gaussian), 95.5% (two Gaussian mixtures) and 96.3% (three Gaussian mixtures). An HMM model set trained to model data from only the second frequency band (2-10kHz) achieves 73.4% (single Gaussian), 79.4% (two Gaussian mixtures) and 80.7% (three Gaussian mixtures).

There are several thresholds in the hierarchical, PT-FHMM clustering procedure. The minimum number of frames per leaf occupancy threshold was set to a fixed value of 100 observations for all experiments. The number of equivalence classes allocated to transition and observation distributions was controlled through varying the thresholds associated with gains in auxiliary function (Section 7.4.1). Note that the number of parameters allocated to the PT-FHMM models may differ by class so where parameter totals are stated, these represent an average taken over all 26 models.

Recall that the number and type of coupling parameters in a PT-FHMM model can be

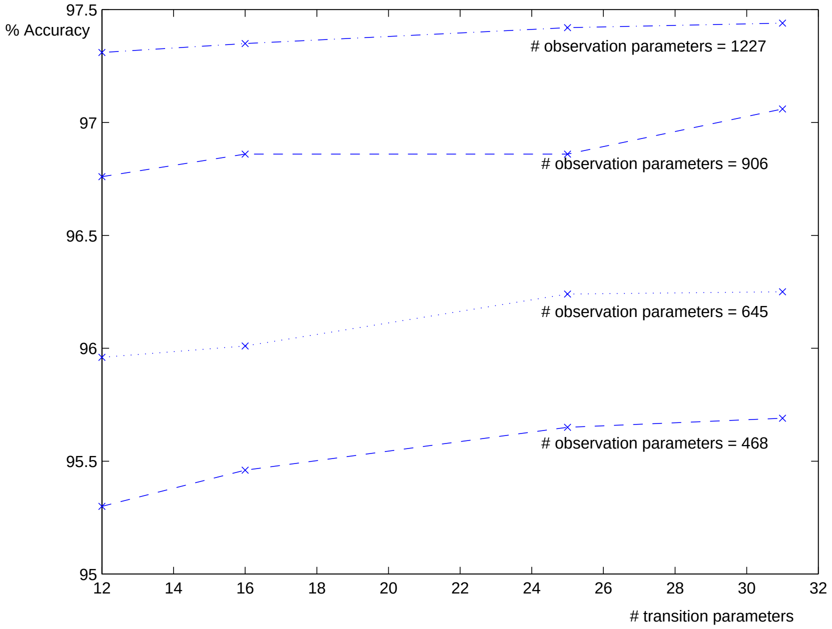


Figure 9.12 *Results: Single Gaussian PT-FHMMs (observation vectors derived from 2 frequency subbands and 2 chains)*

varied through the use of different stopping thresholds in the hierarchical partitioning procedure. The graph in Figure 9.12 shows the performance of the PT-FHMM as the number and type of coupling parameters is varied. The point in the bottom-left corner of the graph (with 468 observation parameters and 12 transition parameters) corresponds to the fully-tied PT-FHMM. The performance of 95.3% for this model initialised and then trained using the PT-FHMM procedure is rather better than the 94.0% achieved by a single Gaussian multiband model using the full metastate space and trained as described above. Some of the difference is attributed to the different initializations and training procedures rather than simply the difference in metastate space topology. (For example, the post-clustering PT-FHMM training uses synchronization points whereas the multiband models used in recognition are formed from independent streams models of each observation stream. This additional freedom during training may lead to poor alignments, particularly in the higher frequency subband.) The graph shows firstly that for a fixed number of observation-related parameters, increasing the number of transition-coupling parameters consistently improves recognition accuracy. The graph further shows that increasing the number of observation-coupling parameters consistently improves recognition accuracy.

The performance of the PT-FHMM is compared with more conventional schemes for modelling frequency subband representations of speech in Tables 9.18, 9.19 and 9.20. Recall that the HTK multiple stream, extended-PMC and multiband models differ in the degree of asynchrony allowed between streams. To reflect this, results for these models are ordered in terms of increasing potential asynchrony: the synchronous HTK multiple stream model will be followed by the extended-PMC model which has loose synchrony constraints and then by the more asynchronous multiband model. The single Gaussian PT-FHMM has a variable number of parameters so the PT-FHMM coupling structure for the result in each of the three tables is chosen to be comparable to the more conventional models in terms of the number and type of parameters. (These PT-FHMM results are cho-

Model	# Parameters Per Model	% Acc
HTK multiple stream	474	94.4
extended-PMC (pruned metastate space)	493	94.2
extended-PMC (full metastate space)	542	94.1
multiband (full metastate space)	480	94.0
PT-FHMM	493	95.7

Table 9.18 *Results: Single Gaussian Systems (observation vectors are cepstra derived from 2 frequency subbands)*

Model	# Parameters Per Model	% Acc
HTK multiple stream	954	97.0
extended-PMC (pruned metastate space)	986	97.0
extended-PMC (full metastate space)	1035	96.9
multiband (full metastate space)	960	96.8
PT-FHMM	937	97.1

Table 9.19 *Results: Single Gaussian PT-FHMM vs Two Gaussian Mixture Systems (observation vectors are cepstra derived from 2 frequency subbands)*

sen from the set shown in Figure 9.12.) However, since there is a maximum number of parameters that can be included in the single Gaussian PT-FHMM (corresponding to the top right-hand point of Figure 9.12), the PT-FHMM result included in Table 9.20 has slightly fewer parameters than the more conventional models. We note in passing that the single Gaussian HTK multiple stream results in Table 9.18 differ from those for the single Gaussian HMMs modelling the concatenated observation streams in Table 9.17. The models are theoretically equivalent and the differences are attributed to differences in implementation.

The significance of all of these results was assessed using the NIST scoring and significance testing package, specifically the MAPSSWE (Matched Pair Sentence Segment (Word Error)) option [118]. Some individual results are statistically significant, but there are only two overall statistically significant trends that hold across the three sets of results discussed above. Firstly, the extended-PMC model with pruned metastate topology has performance above and which is statistically significantly different to the multiband model at the $\alpha = 0.01$ level ($p < 0.001$ in all cases). Secondly, PT-FHMMs matched with multiband models in terms of number and type of parameters have higher performance and are statistically significantly different at the $\alpha = 0.01$ level ($p < 0.002$ in all cases).

Model	# Parameters Per Model	% Acc
HTK multiple stream	1434	97.5
extended-PMC (pruned metastate space)	1465	97.6
extended-PMC (full metastate space)	1524	97.5
multiband (full metastate space)	1464	97.4
PT-FHMM	1258	97.4

Table 9.20 *Results: Single Gaussian PT-FHMM vs Three Gaussian Mixture Systems (observation vectors are cepstra derived from 2 frequency subbands)*

9.2.3.3 Conclusions

Firstly, the results show that the PT-FHMM can be scaled to a continuous speech recognition task larger than ISOLET and still achieve performance competitive with the more conventional HMM, HTK multiple stream, extended-PMC and multiband systems. Secondly, Figure 9.12 shows use of additional observation- or transition-related coupling parameters consistently leads to performance improvements. Thirdly, the incorporation of soft synchrony constraints using the extended-PMC (pruned metastate space) approach or the PT-FHMM approach gives small improvements in performance over the basic multiband approach; these differences are significantly significant at $\alpha = 0.01$ level using the MAPSSWE test.

The results suffice to show that PT-FHMMs scale competitively to continuous speech tasks and merit further investigation. They also show that the incorporation of soft synchrony constraints can improve performance over a multiband approach. These results are promising but a note of caution is required. Firstly, the results suggest that the effects of model initialization and training procedure are particularly important on the TI-DIGITS task and a more extensive study of these effects is necessary. Secondly, these experiments involve only two observation streams. The difficulties associated with collecting sufficient statistics for the two stream case suggest some ingenuity may be required when modelling more observation streams or using longer chains per modelling unit. All of this further experimentation is beyond the scope of the dissertation. Given these caveats, however, the results overall show that the PT-FHMM provides a flexible and scaleable approach to explicitly modelling asynchrony whilst maintaining competitive classification performance.

10

Conclusions

10.1 Summary

This dissertation began by showing there are factors associated with the acoustics of conversational speaking styles that seriously degrade the performance of conventional recognizers. Hypothesizing that many of these factors are associated with the increased pronunciation variability in conversational speech, it then considered schemes for better modelling phonological change within the statistical framework for speech recognition.

First, an explicit pronunciation modelling scheme was investigated. Motivated by work in linear phonology, the scheme attempts to produce an improved dictionary with more accurate pronunciations expressed in terms of surface phones. These are less variable in their acoustic realizations than the more abstract phonemic units often used in recognition dictionaries. The new lexicon can be incorporated into an existing system at recognition time. Collaborative experiments at WS97 (The Johns Hopkins University Summer Research Workshop on Large Vocabulary Speech Recognition) showed this approach gives a very small, but statistically significant, improvement on the Switchboard conversational speech corpus. The new lexicon can also be used to produce training set transcriptions more representative of the classes in the acoustic signal; many researchers anticipated that training on these transcriptions would lead to more accurate, lower variance models of surface phones. However, this thesis has shown that, counter to intuition, use of a variety of quantifiably more accurate phone-level training transcriptions during model estimation does not translate into improved *word* recognition performance despite (as shown by colleagues at The Johns Hopkins University) often yielding the expected improvements in *phone* recognition performance. The explanation is the increase in confusability and homophonous sequences that occurs when representing pronunciations in terms of surface phones. These results suggest that explicit pronunciation modelling schemes which model phonological change at the level of phoneme or phone-like units, whilst appealingly simple, are unlikely to lead to large performance improvements in conversational speech transcription in their current form.

These results motivated a more speculative direction of research. Speech scientists and non-linear phonologists today no longer assume the existence of units such as phonemes and surface phones. Instead, they describe pronunciation change at levels below the phoneme. Complex variability in the acoustic signal is often described very simply in terms of these deeper phonological or articulatory mechanisms. The research in the second part of this dissertation is motivated by a belief that improvements in phonological modelling may require a more implicit approach than used in the initial part of the thesis. The direction taken here seeks models which better characterize underlying speech production mechanisms, perhaps through the use of articulatory or phonological representations of speech. There are two primary issues to be addressed: the intermediate representation of speech and the appropriate statistical model. Since many of the plethora of intermediate representations that have been proposed can be thought of abstractly as multiple, loosely-coupled time series, the two issues can be decoupled to some extent. The latter part of the thesis focussed specifically upon the issue of modelling this type of speech data.

The family of Loosely-coupled or Factorial HMMs was identified as potentially appropriate for modelling multiple, loosely-coupled time series data. It was shown that several conventional speech models are members of the FHMM family. An existing FHMM scheme was discussed, the MM-FHMM. Since the MM-FHMM has some potential drawbacks for the specific task of speech modelling, a new more data-driven FHMM scheme was proposed, the PT-FHMM. The PT-FHMM approach makes fewer a-priori assumptions about the nature of the data to be modelled. Estimation and decoding of these types of model may become computationally expensive and some possible approximate algorithms were proposed.

An empirical study investigated whether these particular FHMMs would scale to speech modelling tasks. A preliminary feasibility study showed the performance of the MM-FHMM is not significantly different to more conventional speech models on the ISOLET classification task. Since the advantages of the new models may not be evident on such a simple task, this result was interpreted as sufficient to conclude the models merit further investigation.

A second study compared the assumption-based MM-FHMM scheme with the new data-driven PT-FHMM scheme on the ISOLET task. The new scheme yielded comparable results. Therefore, since the MM-FHMM has potential drawbacks for speech modelling, the PT-FHMM was adopted in later continuous word recognition experiments.

A third study compared the performance of exact and approximate algorithms in decoding and identified an algorithm suitable for use in much larger vocabulary tasks. The Chain Viterbi algorithm gave performance comparable to the exact algorithms and very similar to the optimal Viterbi algorithm. It is easier to implement, is empirically relatively insensitive to initialization, does not require use of heuristic parameters and gives better approximations to likelihood value than a naive Mean-Field Variational Approximation. The study continued by comparing exact and approximate algorithms when used in both estimation and decoding. Performance of the Chain Viterbi algorithm was again comparable to the exact algorithms and the simplicity-related advantages over the Mean-Field Variational approach continue to hold.

A short final study compared the performance of FHMMs with more conventional models on the TI-DIGITS continuous speech recognition task. The PT-FHMM results were comparable with more conventional models for an equivalent number of parameters. There is considerable scope for more detailed investigation, but the experimental results suggest that the PT-FHMM provides a flexible and scaleable approach to modelling multiple loosely-coupled time series representations of speech whilst maintaining competitive classification performance.

10.2 Future Work

There are obvious extensions of the FHMM-based approach that should yield some performance improvements without the need for extensive research. For example, use of exponent stream weighting was not investigated nor combining the FHMM hypotheses with those of an existing system in a ROVER-style classifier combination framework.

Several interesting issues associated with large vocabulary tasks, such as choice of modelling unit and schemes for parameter tying across those units in the face of data sparsity, have not been addressed. A decision was taken to use a word-level modelling unit in the TI DIGITS experiments and there was sufficient data to allow each word-level FHMM to be estimated robustly. However, for larger vocabulary tasks the choice of word-level units may not be the most appropriate and there is unlikely to be adequate data to train a completely distinct model for each unit.

The potential advantages of the FHMMs are thought to be associated with properties of conversational speech. This remains to be shown. In addition, exploiting those potential advantages may require use of alternative representations of speech. Appropriate representations have not been discussed. Although discussion in this dissertation presented FHMMs as suitable for modelling knowledge-based features extracted directly from the acoustics, many schemes for extracting knowledge-based features require hard decisions to be made in the acoustic preprocessing stage which is not necessarily desirable. There are other possibilities for incorporating FHMMs within a recognition system whilst continuing to incorporate speech production knowledge, such as approaches motivated by work in analysis-by-synthesis eg. [135].

There are a variety of possible estimation schemes based on the Chainwise Viterbi procedure. For example, the metastate sequences resulting after each chainwise decoding can be viewed as a sample from the metastate space and could perhaps be incorporated into an N-Best EM training algorithm[1]. Possible variants and their convergence properties have not been studied in this dissertation.

Finally, FHMMs are potentially suitable for any modelling problem involving multiple, loosely-coupled time series. The thesis has shown that the models and algorithms scale acceptably to larger tasks than the toy examples on which they have previously been tested in the machine learning literature. Tasks where the models might be of interest include noise robustness (which provided the original motivation for multiband models), incorporating prosodic information and audio-visual speech recognition.

[1] The author thanks Asela Gunawardana for this suggestion.

A

Proof of Lemma

Lemma
A function $\sum_{j=1}^{N} w_j \log y_j$ of variables $\{y_j\}_{j=1}^{N}$, subject to constraints $\sum_{j=1}^{N} y_j = 1$ and $\forall j. y_j \geq 0$, attains a global maximum at the single point $y_j = \frac{w_j}{\sum_{i=1}^{N} w_i}$ for $j = 1, \ldots, N$.

Proof
Use Lagrange multipliers, with a single constraint function $g(\mathbf{y}) = \sum_{j=1}^{N} y_j - 1 = 0$. (We shall see the solution automatically satisfies the positivity constraints.) The Lagrangian, incorporating Lagrange multiplier λ, is:

$$L(\mathbf{y}) \quad = \quad f(\mathbf{y}) - \lambda(\sum_{j=1}^{N} y_j - 1) = \sum_{j=1}^{N}(w_j \log y_j - \lambda(y_j - 1))$$

Differentiating and equating to zero:

$$\begin{aligned} \frac{\partial L}{\partial y_i} &= \frac{w_i}{y_i} - \lambda = 0 \\ \frac{\partial L}{\partial \lambda} &= -(\sum_{j=1}^{N} y_j - 1) = 0 \end{aligned}$$

from which we conclude that for all i, $y_i = \frac{w_i}{\lambda}$. Incorporating constraint $g(\mathbf{y})$ we find $\lambda = \sum_{j=1}^{N} w_j$, from which the result follows.

B

Symbolic Differentiation Rules

The following notation is used for symbolic differentiation. Let $f : \Re^D \rightarrow \Re$ be a differentiable function mapping a D-dimensional column vector $\mathbf{x}$ to a real number $f(\mathbf{x})$. Then $\frac{df}{d\mathbf{x}}$ is the $D \times 1$ column vector whose ith element is $\frac{\partial f}{\partial x_i}$. Similarly, let $f : \Re^{D \times D} \rightarrow \Re$ be a differentiable function mapping a square $D \times D$ matrix with elements A_{ij} to a real number $f(A)$. Then $\frac{df}{dA}$ is the $D \times D$ matrix whose ijth element is $\frac{\partial f}{\partial a_{ij}}$.

Rule 1
Let A be a real symmetric $D \times D$ matrix, $\mathbf{x}$ a real $D \times 1$ vector and $f(A) = \mathbf{x}^T A \mathbf{x}$.

$$\begin{aligned}
\frac{\partial f}{\partial x_i} &= \sum_{mn} \frac{\partial}{\partial x_i} A_{mn} x_m x_n = 2 \sum_m A_{im} x_m \\
\frac{df}{d\mathbf{x}} &= 2A\mathbf{x}
\end{aligned}$$

Rule 2
Let A be a real symmetric $D \times D$ matrix, $\mathbf{x}$ and $\mathbf{y}$ are real $D \times 1$ vectors and $f(A) = \mathbf{x}^T A \mathbf{y}$.

$$\begin{aligned}
\frac{\partial f}{\partial A_{ij}} &= \sum_{mn} \frac{\partial}{\partial A_{ij}} A_{mn} x_m y_n = x_i y_j \\
\frac{df}{dA} &= \mathbf{x}\mathbf{y}
\end{aligned}$$

Rule 3
Let A be a real symmetric $D \times D$ matrix and $f(A) = |A|$.

$$\begin{aligned}
|A| &= \sum_{ij} cof_{ij}(A) A_{ij} \\
(A^{-1})_{ij} &= \frac{cof_{ji}(A)}{|A|} \\
\frac{\partial f}{\partial A_{ij}} &= cof_{ij}(A) = |A|(A^{-1})_{ij} \\
\frac{df}{dA} &= |A|(A^{-1})^T = |A|(A^{-1})
\end{aligned}$$

Rule 4

Let A and B be real symmetric $D \times D$ matrices, $\mathbf{x}$ and $\mathbf{y}$ are real $D \times 1$ vectors and $f(A) = (B\mathbf{x} + \mathbf{y})^T A(B\mathbf{x} + \mathbf{y})$.

$$\begin{aligned}
\frac{\partial f}{\partial x_i} &= \sum_{mn} A_{mn} \frac{\partial}{\partial x_i}[(B\mathbf{x} + \mathbf{y})_m (B\mathbf{x} + \mathbf{y})_n] \\
&= \sum_{mn} A_{mn}[B_{mi}(B\mathbf{x} + \mathbf{y})_n + B_{ni}(B\mathbf{x} + \mathbf{y})_m] \\
&= 2\sum_{mn} B_{mi} A_{mn}(B\mathbf{x} + \mathbf{y})_n \\
&= 2\sum_{mn} B^T_{im} A_{mn}(B\mathbf{x} + \mathbf{y})_n \\
\frac{df}{d\mathbf{x}} &= 2B^T A(B\mathbf{x} + \mathbf{y})
\end{aligned}$$

C

ML Estimates for Multivariate Normal Density

Consider a sequence of observations $\mathbf{x}_1^N = \mathbf{x}_1, \ldots, \mathbf{x}_N$, where each $x_n \in \Re^D$ is drawn independently from a Gaussian distribution $N(\mu, \Sigma)$ with unknown mean μ and covariance Σ. Assume that Σ is nonsingular.

The log-likelihood function can be written:

$$\begin{aligned}
\mathcal{L}(\mathbf{x}_1^N; \mu, \Sigma) &= \sum_{n=1}^{N} \log P(\mathbf{x}_n; \mu, \Sigma) \\
&= \sum_{n=1}^{N} \{-\log((2\pi)^{D/2}|\Sigma|^{\frac{1}{2}} - \frac{1}{2}(\mathbf{x}_n - \mu)^T \Sigma^{-1}(\mathbf{x}_n - \mu)\} \\
&= -\frac{ND}{2}\log 2\pi + \frac{N}{2}\log|\Sigma^{-1}| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \mu)^T \Sigma^{-1}(\mathbf{x}_n - \mu)
\end{aligned}$$

where we use the fact $|\Sigma^{-1}| = |\Sigma|^{-1}$. Using the symbolic differentiation rules of Appendix B we obtain:

$$\begin{aligned}
\frac{d\mathcal{L}}{d\mu} &= \sum_{n=1}^{N} \Sigma^{-1}(\mathbf{x}_n - \mu) = 0 \\
\frac{d\mathcal{L}}{d\Sigma^{-1}} &= \frac{N}{2}\frac{1}{|\Sigma^{-1}|}|\Sigma^{-1}|(\Sigma) - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \mu)(\mathbf{x}_n - \mu)^T = 0
\end{aligned}$$

Note now that, because there is a unique ML solution for the inverse covariance matrix and because each covariance matrix has a unique inverse (and vice-versa), then if the ML estimate of Σ^{-1} is given by a matrix M the ML estimate of Σ^{-1} is given by M^{-1}. Thus

$$\begin{aligned}
\hat{\mu} &= \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n \\
\hat{\Sigma} &= \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \mu)(\mathbf{x}_n - \mu)^T
\end{aligned}$$

These results extend straightforwardly to the M-step of EM for the MM-FHMM and PT-FHMM.

D

Parameter Estimation For MM-FHMM

D.1 Deriving ML parameter estimates

ML estimation of MM-FHMM parameters $\lambda = (\pi, A, B, \phi, \psi)$ is achieved using an EM algorithm [33]. Following [150], equations (7.4) and (7.6) are viewed as mixture models, introducing two new types of latent variable in addition to those denoting the metastate sequence taken through the model. The new latent variables, denoted by y_t^k and x_t^k below, encode the identity of the cross-emission distribution and cross-transition matrix (ie. the distributions within each mixture model) used in each stream k at each t. Figure D.1 illustrates the information provided by the x_t^k and y_t^k variables.

The notation for latent variables is as follows:

- s_t^k : state occupied in stream k at time t;
- $\mathbf{S}_t = (s_t^1, \ldots, s_t^K)$: metastate occupied at t;
- $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$: a sequence of metastates;
- $\mathcal{S} = \{\mathbf{S}\}$: the set of possible metastate sequences;
- x_t^k : the hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_{t-1}$ which determines the matrix used for the transition into s_t^k;
- $\mathbf{X}_t = (x_t^1, \ldots, x_t^K)$;
- $\mathbf{X} = \mathbf{X}_1, \ldots, \mathbf{X}_T$: single transition component vector sequence
- $\mathcal{X} = \{\mathbf{X}\}$ the set of possible sequences ;
- y_t^k : the hidden variable $\in \{1, \ldots, K\}$ indicating the component of $\mathbf{S}_t$ which determines the output probability for o_t^k ;
- $\mathbf{Y}_t = (y_t^1, \ldots, y_t^K)$;
- $\mathbf{Y} = \mathbf{Y}_1, \ldots, \mathbf{Y}_T$ a single (observation-predictor) state-component sequence;

- $\mathcal{Y} = \{\mathbf{Y}\}$: the set of possible sequences.

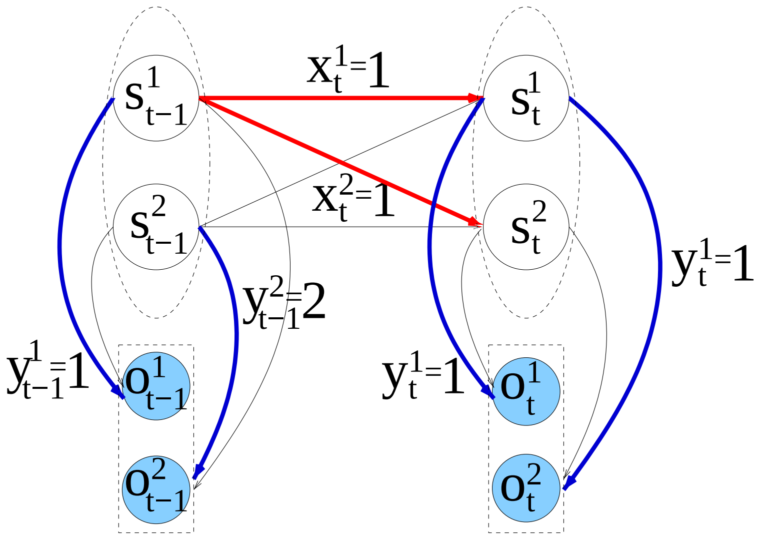


Figure D.1 *Bold lines show information specified by the hidden (vector) variables Y_{t-1}, X_t, Y_t.*

The derivation that follows assumes non-emitting exit states are *not* in use. Probabilities associated with the complete data set are formed as:

$$\begin{aligned}
P(\mathbf{S}_t, \mathbf{X}_t|\mathbf{S}_{t-1}) &= \prod_{k=1}^{K} P(s^k_t, x^k_t|\mathbf{S}_{t-1}) \\
P(s^k_t, x^k_t|\mathbf{S}_{t-1}) &= \psi^k(x^k_t) a^{kx^k_t}(s^k_t|s^{x^k_t}_{t-1}) \\
P(\mathbf{S}_1, \mathbf{X}_1) &= \prod_{k=1}^{K} P(s^k_1, x^k_1) \\
P(s^k_1, x^k_1) &= \psi^k(x^k_1)\pi^{kx^k_1}(s^k_1) \\
p(\mathbf{O}_t, \mathbf{Y}_t|\mathbf{S}_t) &= \prod_{k=1}^{K} p(o^k_t, y^k_t|\mathbf{S}_t) \\
p(o^k_t, y^k_t|\mathbf{S}_t) &= \phi^k(y^k_t) b^{kl}(o^k_t|s^{y^k_t}_t)
\end{aligned}$$

Thus:

$$\begin{aligned}
p(\mathbf{O}, \mathbf{S}, \mathbf{X}, \mathbf{Y}) &= \pi(\mathbf{S}_1, \mathbf{X}_1) p(\mathbf{O}_1, \mathbf{Y}_1|\mathbf{S}_1)\{\prod_{t=2}^{T} P(\mathbf{S}_t, \mathbf{X}_t|\mathbf{S}_{t-1}) p(\mathbf{O}_t, \mathbf{Y}_t|\mathbf{S}_t)\} \\
\log p(\mathbf{O}, \mathbf{S}, \mathbf{X}, \mathbf{Y}) &= \sum_{k=1}^{K}\sum_{t=1}^{T} \log \psi^k(x^k_t) + \sum_{k=1}^{K} \log \pi^{kx^k_1}(s^k_1) + \\
&+ \sum_{k=1}^{K}\sum_{t=1}^{T} \log \phi^k(y^k_t) + \sum_{k=1}^{K}\sum_{t=1}^{T} \log b^{ky^k_t}(o^k_t|s^{y^k_t}_t) \\
&+ \sum_{k=1}^{K}\sum_{t=2}^{T} a^{kx^k_t}(s^k_t|s^{x^k_t}_{t-1})
\end{aligned}$$

Denoting current and updated model parameters by λ and $\hat{\lambda}$, the EM auxiliary function $\mathcal{Q}(\lambda, \hat{\lambda})$ is:

$$\begin{aligned}
\mathcal{Q}(\lambda, \hat{\lambda}) &= \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{\mathbf{Y}\in\mathcal{Y}} P(\mathbf{S},\mathbf{X},\mathbf{Y}|\mathbf{O}) \log \hat{p}(\mathbf{O},\mathbf{S},\mathbf{X},\mathbf{Y}) \\
&= \sum_{k=1}^{K} \sum_{\mathbf{Y}\in\mathcal{Y}} \sum_{t=1}^{T} P(\mathbf{Y}|\mathbf{O}) \log \hat{\phi}^k(y_t^k) + \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{Y}\in\mathcal{Y}} \sum_{t=1}^{T} P(\mathbf{S},\mathbf{Y}|\mathbf{O}) \log \hat{b}^{k y_t^k}(o_t^k | s_t^{y_t^k}) \\
&\quad + \sum_{k=1}^{K} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{t=1}^{T} P(\mathbf{X}|\mathbf{O}) \log \hat{\psi}^k(x_t^k) + \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{t=2}^{T} P(\mathbf{S},\mathbf{X}|\mathbf{O}) \log \hat{a}^{k x_t^k}(s_t^k | s_{t-1}^{x_t^k}) \\
&\quad + \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{X}\in\mathcal{X}} P(\mathbf{S},\mathbf{X}|\mathbf{O}) \log \hat{\pi}^{k x_1^k}(s_1^k) \\
&= \mathcal{Q}_\phi(\lambda, \hat{\lambda}) + \mathcal{Q}_b(\lambda, \hat{\lambda}) + \mathcal{Q}_\psi(\lambda, \hat{\lambda}) + \mathcal{Q}_a(\lambda, \hat{\lambda}) + \mathcal{Q}_\pi(\lambda, \hat{\lambda})
\end{aligned}$$

Thus $\mathcal{Q}(\lambda, \hat{\lambda})$ comprises subfunctions that may be maximized separately. The following Lemma (proved in Appendix A) will be used repeatedly:

Lemma
A function $\sum_{j=1}^{N} w_j \log y_j$ of variables $\{y_j\}_{j=1}^{N}$, subject to constraints $\sum_{j=1}^{N} y_j = 1$ and $\forall j.y_j \geq 0$, attains a global maximum at the single point $y_j = \frac{w_j}{\sum_{i=1}^{N} w_i}$ for $j = 1, \ldots, N$.

1. Optimizing $\mathcal{Q}_\phi(\lambda, \hat{\lambda})$**:** To maximize

$$\mathcal{Q}_\phi(\lambda, \hat{\lambda}) = \sum_{k=1}^{K} \sum_{\mathbf{Y}\in\mathcal{Y}} \sum_{t=1}^{T} P(\mathbf{Y}|\mathbf{O}) \log \hat{\phi}^k(y_t^k)$$

subject to the constraints that $\sum_{\mu=1}^{K} \hat{\phi}^k(\mu) = 1$ and $\forall \mu.\hat{\phi}^k(\mu) \geq 0$, first rearrange as

$$\begin{aligned}
\mathcal{Q}_\phi(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{\mathbf{Y}\in\mathcal{Y}} \sum_{t=1}^{T} \sum_{l=1}^{K} \delta(y_t^k, l) P(\mathbf{Y}|\mathbf{O}) \log \hat{\phi}^k(l) \\
&= \sum_{k=1}^{K} \sum_{l=1}^{K} \{\sum_{t=1}^{T} P(y_t^k = l|\mathbf{O})\} \log \hat{\phi}^k(l)
\end{aligned}$$

where $\delta(x, y)$ is the Dirac δ function, ie. 1 if $x = y$ and 0 otherwise. Then maximize for each k individually and appeal to Lemma to conclude that for each k:

$$\begin{aligned}
\hat{\phi}^k(l) &= \frac{\sum_{t=1}^{T} P(y_t^k = l|\mathbf{O})}{\sum_{v=1}^{K} \sum_{t=1}^{T} P(y_t^k = v|\mathbf{O})} \\
&= \frac{\sum_{t=1}^{T} P(y_t^k = l|\mathbf{O})}{T}
\end{aligned}$$

2. Optimizing $\mathcal{Q}_\psi(\lambda, \hat{\lambda})$**:** To maximize

$$\mathcal{Q}_\psi(\lambda, \hat{\lambda}) = \sum_{k=1}^{K} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{t=1}^{T} P(\mathbf{X}|\mathbf{O}) \log \hat{\psi}^k(x_t^k)$$

subject to the constraints that $\sum_{\mu=1}^{K} \hat{\psi}^k(\mu) = 1$ and $\forall \mu . \hat{\psi}^k(\mu) \geq 0$, first rearrange as

$$\begin{aligned} \mathcal{Q}_\psi(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{\mathbf{X} \in \mathcal{X}} \sum_{t=1}^{T} \sum_{l=1}^{K} \delta(x_t^k, l) P(\mathbf{X}|\mathbf{O}) \log \hat{\psi}^k(l) \\ &= \sum_{k=1}^{K} \sum_{l=1}^{K} \{ \sum_{t=1}^{T} P(x_t^k = l|\mathbf{O}) \} \log \hat{\psi}^k(l) \end{aligned}$$

Then maximize for each k individually and appeal to Lemma to conclude that for each k:

$$\begin{aligned} \hat{\psi}^k(l) &= \frac{\sum_{t=1}^{T} P(x_t^k = l|\mathbf{O})}{\sum_{v=1}^{K} \sum_{t=1}^{T} P(x_t^k = v|\mathbf{O})} \\ &= \frac{\sum_{t=1}^{T} P(x_t^k = l|\mathbf{O})}{T} \end{aligned}$$

3. Optimizing $\mathcal{Q}_a(\lambda, \hat{\lambda})$**:** To maximize

$$\mathcal{Q}_a(\lambda, \hat{\lambda}) = \sum_{k=1}^{K} \sum_{\mathbf{S} \in \mathcal{S}} \sum_{\mathbf{X} \in \mathcal{X}} \sum_{t=2}^{T} P(\mathbf{S}, \mathbf{X}|\mathbf{O}) \log \hat{a}^{k x_t^k}(s_t^k | s_{t-1}^{x_t^k})$$

subject to the constraints that for each k,l and $i^l \in \Theta_l$, $\sum_{j^k \in \Theta_k} \hat{a}^{kl}(j^k|i^l) = 1$ and $\forall j^k \in \Theta_k . \hat{a}^{kl}(j^k|i^l) \geq 0$, first rearrange as:

$$\begin{aligned} \mathcal{Q}_a(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{\mathbf{S} \in \mathcal{S}} \sum_{\mathbf{X} \in \mathcal{X}} \sum_{t=2}^{T} \sum_{l=1}^{K} \sum_{j^k \in \Theta_k} \sum_{i^l \in \Theta_l} \delta(s_t^k, j^k) \delta(s_{t-1}^l, i^l) \delta(x_t^k, l) P(\mathbf{S}, \mathbf{X}|\mathbf{O}) \log \hat{a}^{kl}(j^k|i^l) \\ &= \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{i^l \in \Theta_l} \sum_{j^k \in \Theta_k} \{ \sum_{t=2}^{T} P(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l|\mathbf{O}) \} \log \hat{a}^{kl}(j^k|i^l) \end{aligned}$$

Then optimize the inner expression for each k, l and i individually and appeal to Lemma to conclude that for each k:

$$\hat{a}^{kl}(j^k|i^l) = \frac{\sum_{t=2}^{T} P(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})}{\sum_{t=2}^{T} P(s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})}$$

4. Optimizing $\mathcal{Q}_b(\lambda, \hat{\lambda})$**:** To maximize

$$\mathcal{Q}_b(\lambda, \hat{\lambda}) = \sum_{k=1}^{K} \sum_{\mathbf{S} \in \mathcal{S}} \sum_{\mathbf{Y} \in \mathcal{Y}} \sum_{t=1}^{T} P(\mathbf{S}, \mathbf{Y}|\mathbf{O}) \log \hat{b}^{k y_t^k}(o_t^k | s_t^{y_t^k})$$

first rearrange as

$$\begin{aligned}
\mathcal{Q}_b(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{Y}\in\mathcal{Y}} \sum_{t=1}^{T} \sum_{l=1}^{K} \sum_{i^l\in\Theta_l} \delta(y_t^k, l)\delta(s_t^l, i^l) P(\mathbf{S}, \mathbf{Y}|\mathbf{O}) \log \hat{b}^{kl}(o_t^k|i^l) \\
&= \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{i^l\in\Theta_l} \sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O}) \log \hat{b}^{kl}(o_t^k|i^l)
\end{aligned}$$

Assume each distribution $b^{kl}(o_t^k|i^l)$ is modelled using a multivariate Gaussian density $\mathcal{N}(\mu_i^{kl}, \Sigma_i^{kl})$. Thus, using A^T to denote matrix transpose:

$$b^{kl}(o_t^k|i^l) \overset{\text{def}}{=} \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_i^{kl}|^{\frac{1}{2}}} e^{-\frac{1}{2}(o_t^k - \mu_i^{kl})^T (\Sigma_i^{kl})^{-1} (o_t^k - \mu_i^{kl})}$$

Substituting for the output density expression gives:

$$\sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{i^l\in\Theta_l} \sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O}) \{ -\frac{D_k}{2} \log 2\pi - \frac{1}{2} \log |\hat{\Sigma}_i^{kl}| - \frac{1}{2}(o_t^k - \hat{\mu}_i^{kl})^T (\hat{\Sigma}_i^{kl})^{-1} (o_t^k - \hat{\mu}_i^{kl}) \}$$

The derivation of the maximizing $\hat{\mu}_i^{kl}$ and $\hat{\Sigma}_i^{kl}$ maximize this equation is similar to the derivation of ML parameter estimates for the multivariate Gaussian distribution, included in Appendix C for completeness. The resulting reestimation equations are

$$\begin{aligned}
\hat{\mu}_i^{kl} &= \frac{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O}) o_t^k}{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O})} \\
\hat{\Sigma}_i^{kl} &= \frac{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O}) (o_t^k - \hat{\mu}_i^{kl})(o_t^k - \hat{\mu}_i^{kl})^T}{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O})} \\
&= \frac{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O}) (o_t^k)(o_t^k)^T}{\sum_{t=1}^{T} P(y_t^k = l, s_t^l = i^l|\mathbf{O})} - \hat{\mu}_i^{kl} (\hat{\mu}_i^{kl})^T
\end{aligned}$$

In practice updates and accumulates are not performed for full covariance or inverse covariance matrices. Instead, the positive definite covariance matrices are decomposed using the (numerically stable) Cholesky decomposition $\Sigma_i^{kl} = AA^T$ and the inverted lower Cholesky factor A^{-1} is stored.

5. Optimizing $\mathcal{Q}_\pi(\lambda, \hat{\lambda})$: To maximize

$$\mathcal{Q}_\pi(\lambda, \hat{\lambda}) = \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{X}\in\mathcal{X}} P(\mathbf{S}, \mathbf{X}|\mathbf{O}) \log \hat{\pi}^{kx_1^k}(s_1^k)$$

subject to the constraints that for each k and l, $\sum_{j\in\Theta_k} \hat{\pi}^{kl}(j) = 1$ and $\forall j.\hat{\pi}^{kl}(j) \geq 0$, first rearrange as:

$$
\begin{aligned}
\mathcal{Q}_\pi(\lambda, \hat{\lambda}) &= \sum_{k=1}^{K} \sum_{\mathbf{S}\in\mathcal{S}} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{l=1}^{K} \sum_{j^k\in\Theta_k} \delta(x_1^k, l)\delta(s_1^k, j^k) P(\mathbf{S}, \mathbf{X}|\mathbf{O}) \log \hat{\pi}^{kl}(j^k) \\
&= \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{j^k\in\Theta_k} P(s_1^k = j^k, x_1^k = l|\mathbf{O}) \log \hat{\pi}^{kl}(j^k)
\end{aligned}
$$

Then optimize the inner expression for each k and l and appeal to Lemma to conclude that for each k, l and $j^k \in \Theta_k$:

$$
\hat{\pi}^{kl}(j^k) = \frac{P(s_1^k = j^k, x_1^k = l|\mathbf{O})}{P(x_1^k = l|\mathbf{O})}
$$

D.2 Evaluation of Posterior Probabilities

The required posterior are evaluated exactly probabilities using a generalization of the forward-backward algorithm [10]. The forward and backward probabilities needed here are calculated within the *metastate* space. ie. the forward probabilities are defined as as $\alpha_{\mathbf{S}}(t) = p(\mathbf{o}_1, \ldots, \mathbf{o}_t, \mathbf{S}_t = \mathbf{S})$ and the backward probabilities as $\beta_{\mathbf{S}}(t) = p(\mathbf{o}_{t+1}, \ldots, \mathbf{o}_T|\mathbf{S}_t = \mathbf{S})$. The total likelihood is then given by eg. $p(\mathbf{O}) = \sum_{\mathbf{S}\in\Theta_{meta}} \alpha_{\mathbf{S}}(T)$. Calculation of the posterior probabilities will require summations over sets of metastates that share a common value for one of the component states. Notation "$\mathbf{S}|s^k = j^k$" denotes the set $\{\mathbf{S} \in \Theta_{meta}|s^k = j^k\}$.

1. Cross-Transition Matrices a^{kl}

The required posterior probability $P(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})$, for $k, l \in \{1, \ldots, K\}$, $j^k \in \Theta_k$ and $i^l \in \Theta_l$, is closely related to:

$$
\begin{aligned}
&p(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l, \mathbf{O}) \\
&= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{S}'|s'^l=i^l} \sum_{\mathbf{X}|x^k=l} P(\mathbf{S}_t = \mathbf{S}, \mathbf{S}_{t-1} = \mathbf{S}', \mathbf{X}_t = \mathbf{X}, \mathbf{O}) \\
&= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{S}'|s'^l=i^l} \sum_{\mathbf{X}|x^k=l} \alpha_{\mathbf{S}'}(t-1) P(\mathbf{S}, \mathbf{X}|\mathbf{S}') p(\mathbf{o}_t|\mathbf{S}) \beta_{\mathbf{S}}(t) \\
&= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{S}'|s'^l=i^l} \alpha_{\mathbf{S}'}(t-1) p(\mathbf{o}_t|\mathbf{S}) \beta_{\mathbf{S}}(t) \{ \sum_{\mathbf{X}|x^k=l} p(\mathbf{S}, \mathbf{X}|\mathbf{S}') \} \\
&= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{S}'|s'^l=i^l} \alpha_{\mathbf{S}'}(t-1) p(\mathbf{o}_t|\mathbf{S}) \beta_{\mathbf{S}}(t) \{ \sum_{\mathbf{X}|x^k=l} \prod_{v=1}^{K} \psi^v(x^v) a^{vx^v}(s^v|s'^{x^v}) \} \\
&= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{S}'|s'^l=i^l} \alpha_{\mathbf{S}'}(t-1) p(\mathbf{o}_t|\mathbf{S}) \psi^k(l) a^{kl}(j|i) \beta_{\mathbf{S}}(t) \sum_{\mathbf{X}|x^k=l} \{ \prod_{v=1, v\neq k}^{K} \psi^v(x^v) a^{vx^v}(s^v|s'^{x^v}) \} \\
&= \psi^k(l) a^{kl}(j|i) \{ \sum_{\mathbf{S}|s^k=j^k} \left[\sum_{\mathbf{S}'|s'^l=i^l} \alpha_{\mathbf{S}'}(t-1) \bar{P}^k(\mathbf{S}|\mathbf{S}') \right] p(\mathbf{o}_t|\mathbf{S}) \beta_{\mathbf{S}}(t) \}
\end{aligned}
$$

where

$$\bar{P}^k(\mathbf{S}|\mathbf{S}') \stackrel{\text{def}}{=} \prod_{v=1, v\neq k}^{K} P(s^v|\mathbf{S}')$$

The required posterior is obtained by normalizing with $p(\mathbf{O})$.

2. Transition Mixed-Memory Weights ψ

The posterior $P(x_t^k = l|\mathbf{O})$ for ψ reestimation is closely related to:

$$p(x_t^k = l, \mathbf{O}) = \sum_{j^k \in \Theta_k} \sum_{i^l \in \Theta_l} p(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l, \mathbf{O})$$

where an expression for $p(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l, \mathbf{O})$ was given in the previous section. The required posterior is obtained by normalizing with $p(\mathbf{O})$.

3. Cross-Emission Output Distributions b^{kl}

The required posterior probability $P(y_t^k = l, s_t^l = i^l|\mathbf{O})$ for all k, l and $i^l \in \Theta_l$, is closely related to:

$$\begin{aligned}
&p(y_t^k = l, s_t^l = i^l, \mathbf{O}) \\
&= \sum_{\mathbf{S}|s^l=i^l} \sum_{\mathbf{S}' \in \Theta_{meta}} \sum_{\mathbf{Y}|y^k=l} p(\mathbf{S}_t = \mathbf{S}, \mathbf{Y}_t = \mathbf{Y}, \mathbf{S}_{t-1} = \mathbf{S}', \mathbf{O}) \\
&= \sum_{\mathbf{S}|s^l=i^l} \sum_{\mathbf{S}' \in \Theta_{meta}} \sum_{\mathbf{Y}|y^k=l} \alpha_{\mathbf{S}'}(t-1) P(\mathbf{S}|\mathbf{S}') p(\mathbf{o}_t, \mathbf{Y}|\mathbf{S}) \beta_{\mathbf{S}}(t) \\
&= \sum_{\mathbf{S}|s^l=i^l} \sum_{\mathbf{S}' \in \Theta_{meta}} \alpha_{\mathbf{S}'}(t-1) P(\mathbf{S}|\mathbf{S}') \beta_{\mathbf{S}}(t) \{ \sum_{\mathbf{Y}|y^k=l} \prod_{v=1}^{K} \phi^v(y^v) b^{vy^v}(o_t^v|s^{y^v}) \} \\
&= \sum_{\mathbf{S}|s^l=i^l} \sum_{\mathbf{S}' \in \Theta_{meta}} \alpha_{\mathbf{S}'}(t-1) P(\mathbf{S}|\mathbf{S}') \beta_{\mathbf{S}}(t) \phi^k(l) b^{kl}(o_t^k|j) \{ \sum_{\mathbf{Y}|y^k=l} \prod_{v=1, v\neq k}^{K} \phi^v(y^v) b^{vy^v}(o_t^v|s^{y^v}) \} \\
&= \phi^k(l) b^{kl}(o_t^k|j) \{ \sum_{\mathbf{S}|s^l=i^l} \left[\sum_{\mathbf{S}' \in \Theta_{meta}} \alpha_{\mathbf{S}'}(t-1) P(\mathbf{S}|\mathbf{S}') \right] \bar{p}^k(\mathbf{o}_t|\mathbf{S}) \beta_{\mathbf{S}}(t) \}
\end{aligned}$$

where

$$\bar{p}^k(\mathbf{o}_t|\mathbf{S}) = \prod_{v=1, v\neq k}^{K} p(o_t^v|\mathbf{S})$$

and obtain the required posterior probability by normalizing with $p(\mathbf{O})$.

4. Observation mixed-memory weights ϕ

The required posterior $P(y_t^k = l|\mathbf{O})$ for ϕ matrix reestimation is closely related to:

$$p(y_t^k = l, \mathbf{O}) \quad = \quad \sum_{i^l \in \Theta_l} p(s_t^l = i^l, y_t^k = l, \mathbf{O})$$

where an expression for $p(s_t^l = i^l, y_t^k = l, \mathbf{O})$ was given in the previous section, and then obtain the required posterior by normalizing with $p(\mathbf{O})$.

5. Initial State Distribution π

The required posterior $P(s_1^k = j^k, x_1^k = l|\mathbf{O})$ is closely related to:

$$\begin{aligned} p(s_1^k = j^k, x_1^k = l, \mathbf{O}) &= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{X}|x^k=l} p(\mathbf{S}_1 = \mathbf{S}, \mathbf{X}_1 = \mathbf{X}, \mathbf{O}) \\ &= \sum_{\mathbf{S}|s^k=j^k} \sum_{\mathbf{X}|x^k=l} \pi(\mathbf{S}, \mathbf{X}) p(\mathbf{o}_1|\mathbf{S}) \beta_{\mathbf{S}}(1) \\ &= \sum_{\mathbf{S}|s^k=j^k} \bar{\pi}^k(\mathbf{S}) \psi^k(l) \pi^{kl}(s_1^k) p(\mathbf{o}_1|\mathbf{S}) \beta_{\mathbf{S}}(1) \end{aligned}$$

where

$$\bar{\pi}^k(\mathbf{S}) \stackrel{\text{def}}{=} \prod_{v=1, v \neq k} \pi^v(s^v)$$

The required posterior probability is obtained by normalizing with $p(\mathbf{O})$.

E

Parameter Estimation For PT-FHMM

E.1 Deriving ML Parameter Estimates

ML estimation of MM-FHMM parameters $\lambda = (\pi^1, \ldots, \pi^K, A^1, \ldots, A^K, B^1, \ldots, B^K)$, where π^k denotes parameters of prior distribution $P(j^k)$, A^k denotes parameters of transition distribution $P(j^k|I)$ and B^k those of observation distribution $p(o_t^k|I)$, is achieved using an EM algorithm [33]. Latent variables $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$ are introduced to specify a metastate sequence of length T.

The notation for latent variables is as follows:

- s_t^k : state occupied in stream k at time t;
- $\mathbf{S}_t = (s_t^1, \ldots, s_t^K)$: metastate occupied at t;
- $\mathbf{S} = \mathbf{S}_1, \ldots, \mathbf{S}_T$: a sequence of metastates;
- $\mathcal{S} = \{\mathbf{S}\}$: the set of possible metastate sequences.

The derivation which follows is for an PT-FHMM without tied parameters and assumes that non-emitting exit states are *not* in use. Probabilities associated with the complete data set are formed as:

$$\begin{aligned}
p(\mathbf{O}, \mathbf{S}) &= P(\mathbf{S}_1)p(\mathbf{O}|\mathbf{S}_1)\prod_{t=2}^{T} P(\mathbf{S}_t|\mathbf{S}_{t-1})p(\mathbf{O}_t|\mathbf{S}_t) \\
\log p(\mathbf{O}, \mathbf{S}) &= \sum_{k=1}^{K} \log P(s_1^k) + \sum_{k=1}^{K}\sum_{t=1}^{T} \log p(o_t^k|\mathbf{S}_t) + \sum_{k=1}^{K}\sum_{t=2}^{T} \log P(s_t^k|\mathbf{S}_{t-1})
\end{aligned}$$

Denoting current and updated model parameters by λ and $\hat{\lambda}$, the EM auxiliary function is:

$$\begin{aligned}\mathcal{Q}(\lambda,\hat{\lambda}) &= \sum_{\mathbf{S}\in\mathcal{S}} P(\mathbf{S}|\mathbf{O})\log\hat{p}(\mathbf{O},\mathbf{S}) \\ &= \sum_{k=1}^{K}\sum_{j^k\in\theta_k} P(s_1^k=j^k|\mathbf{O})\log\hat{P}^k(j^k) + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{J\in\Theta_{meta}} P(\mathbf{S}_t=J|\mathbf{O})\log\hat{p}(o_t^k|J) \\ &\quad+\sum_{t=2}^{T}\sum_{k=1}^{K}\sum_{j^k\in\theta_k}\sum_{I\in\Theta_{meta}} P(s_{t-1}=I,s_t^k=j^k|\mathbf{O})\log\hat{P}^k(j^k|I)\end{aligned}$$

Assuming each $p(o_t^k|J)$ is modelled by a Gaussian distribution $\mathcal{N}(\mu_J^k,\Sigma_J^k)$, the parameters maximizing the auxiliary function are derived using steps similar to Appendix D and so the details are not repeated here. The resulting parameter updates are:

$$\begin{aligned}\hat{P}(j^k) &= P(s_1^k=j^k|\mathbf{O}) \\ \hat{P}(j^k|I) &= \frac{\sum_{t=2}^{T}P(s_t^k=j^k,\mathbf{S}_{t-1}=I|\mathbf{O})}{\sum_{t=2}^{T}P(\mathbf{S}_{t-1}=I|\mathbf{O})} \\ \hat{\mu}_J^k &= \frac{\sum_{t=1}^{T}P(\mathbf{S}_t=J|\mathbf{O})o_t^k}{\sum_{t=1}^{T}P(\mathbf{S}_t=J|\mathbf{O})} \\ \hat{\Sigma}_J^k &= \frac{\sum_{t=1}^{T}P(\mathbf{S}_t=J|\mathbf{O})o_t^k(o_t^k)^T-\hat{\mu}_J^k(\hat{\mu}_J^k)^T}{\sum_{t=1}^{T}P(\mathbf{S}_t=J|\mathbf{O})}\end{aligned}$$

The derivation and update equations are similar for the case when sets of distribution equivalence classes have been defined, as is now shown. The parameters to be estimated are (for $1\le k\le K$) the prior probabilities $P(j^k)$, the observation-related distributions $p(o_t^k|C)$ for $C\in\mathcal{C}^{obs,k}$ and transition-related distributions $p(j^k|C)$ for $C\in\mathcal{C}^{trans,k}$. For the case where each $p(o_t^k|J)$ is modelled using a single, full covariance, multivariate Gaussian $\mathcal{N}(\mu_J^k,\Sigma_J^k)$, the reestimation formulae are:

$$\begin{aligned}\hat{P}(j^k) &= P(s_1^k=j^k|\mathbf{O}) \\ \hat{P}(j^k|C) &= \frac{\sum_{I\in C}\sum_{t=2}^{T}\eta_t^k(j^k,I)}{\sum_{I\in C}\sum_{t=2}^{T}\gamma_t(I)} \\ \hat{\mu}_C &= \frac{\sum_{J\in C}\sum_{t=1}^{T}\gamma_t(J)o_t^k}{\sum_{J\in C}\sum_{t=1}^{T}\gamma_t(J)} \\ \hat{\Sigma}_C &= \frac{\sum_{J\in C}\sum_{t=1}^{T}\gamma_t(J)(o_t^k-\hat{\mu}_C)(o_t^k-\hat{\mu}_C)^T}{\sum_{J\in C}\sum_{t=1}^{T}\gamma_t(J)}\end{aligned}$$

Generalization to training using multiple observation sequences is straightforward.

E.2 Evaluation of Posterior Probabilities

State posterior probabilities $\gamma_t(J)\overset{\text{def}}{=}P(\mathbf{S}_t^k=J|\mathbf{O})$ are calculated from standard forward and backward probabilities calculated in the FHMM *metastate* space:

$$P(\mathbf{S}_t = J|\mathbf{O}) \quad = \quad \frac{\alpha_t(J)\beta_t(J)}{p(\mathbf{O})}$$

Calculation of the transition-related posteriors $\eta^k(j^k, I) \overset{\text{def}}{=} P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O})$ requires summations over sets of metastates that share a common value for one of the component states. Notation "$\mathbf{S}|s^k = j^k$" denotes the set $\{\mathbf{S} \in \Theta_{meta}|s^k = j^k, j^k \in \Theta_k\}$. Then:

$$\begin{aligned} P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O}) &= \frac{\sum_{\mathbf{S}|s^k=j^k} P(\mathbf{S}_{t-1} = I, \mathbf{S}_t = \mathbf{S}, \mathbf{O})}{p(\mathbf{O})} \\ &= \frac{\sum_{\mathbf{S}|s^k=j^k} \alpha_{t-1}(I)P(\mathbf{S}|I)p(\mathbf{O}|\mathbf{S})\beta_t(\mathbf{S})}{p(\mathbf{O})} \end{aligned}$$

F

Changes To Equations for Non-Emitting States

The HTK toolkit [180] uses non-emitting model-initial and model-final states to allow prior reestimation to be treated as a special case of the transition reestimation and so that paths included in likelihood calculations pass through a small set of *model final states*. Non-emitting states also simplify the extension from an isolated word to a continuous word recognizer. For similar reasons, our implementations of the MM-FHMM and PT-FHMM use non-emitting initial and final states *within the set of states for each stream k* ie. if the standard model would have emitting states $\{1_k, \ldots, N_k\}$ for the stream k model, then in our implementation it would have states $\{1_k, \ldots, N_k + 2\}$ where 1_k is a stream initial non-emitting state and $N_k + 2$ is a stream final non-emitting state. We then apply a further constraint within the metastate space by allowing entry and exit states from a stream k to occur only in combination with the entry and exit states from the other $K - 1$ streams, ie. $(1^1, \ldots, 1^K)$ and $(N^1, \ldots, N^K)$ are the only states introduced within the metastate space. Initial metastate $\hat{1} = (1^1, \ldots, 1^K)$ may only be occupied before any observations have been emitted; final metastate $\hat{N} = (N^1, \ldots, N^K)$ may only be occupied after all observations have been emitted.

To incorporate this modified model topology, changes are necessary in the forward-backward algorithm and in the transition-related reestimation equations.

F.1 Modified Forward and Backward Algorithms

Using Θ_{meta} to denote the set of *emitting* states only, the forward probability $\alpha_{\mathbf{s}}(t)$ is calculated as:

Forward Algorithm

Step 1: Initialization for each $\mathbf{S} \in \Theta_{meta}$

$$\begin{aligned} \alpha_{\hat{1}}(1) &= 1 \\ \alpha_{\mathbf{S}}(1) &= P(\mathbf{S}|\hat{1})p(\mathbf{O}_1|\mathbf{S}) \end{aligned}$$

Step 2: for each $\mathbf{S} \in \Theta_{meta}$ at $t = 2, \ldots, T$

$$\alpha_{\mathbf{S}}(t) = \left[\sum_{\mathbf{S}' \in \Theta_{meta}} \alpha_{\mathbf{S}'}(t-1)P(\mathbf{S}|\mathbf{S}') \right] p(\mathbf{O}_t|\mathbf{S})$$

Step 3: Termination

$$p(\mathbf{O}|\lambda) = \alpha_{\hat{N}}(T) = \sum_{\mathbf{S} \in \Theta_{meta}} \alpha_{\mathbf{S}}(T)P(\hat{N}|\mathbf{S})$$

The backward probability $\beta_{\mathbf{S}}(t)$ is calculated as:

Backward Algorithm

Step 1: Initialization for each $\mathbf{S} \in \Theta_{meta}$

$$\begin{aligned} \beta_{\hat{N}}(T) &= 1 \\ \beta_{\mathbf{S}}(T) &= P(\hat{N}|\mathbf{S}) \end{aligned}$$

Step 2: for each $\mathbf{S}$ at $t = 2, \ldots, T$

$$\beta_{\mathbf{S}}(t) = \sum_{\mathbf{S}' \in \Theta_{meta}} P(\mathbf{S}'|\mathbf{S}) p(\mathbf{O}_{t+1}|\mathbf{S}') \beta_{\mathbf{S}'}(t+1)$$

Step 3: Termination

$$p(\mathbf{O}|\lambda) = \beta_{\hat{1}}(1) = \sum_{\mathbf{S} \in \Theta_{meta}} P(\mathbf{S}|\hat{1}) p(\mathbf{O}_1|\mathbf{S}) \beta_{\mathbf{S}}(1)$$

F.2 Modified MM-FHMM Transition Reestimation Equations

$$\begin{aligned} \hat{a}^{kl}(j^k|i^l) &= \frac{\sum_{t=2}^{T+1} P(s_t^k = j^k, s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})}{\sum_{t=2}^{T+1} P(s_{t-1}^l = i^l, x_t^k = l|\mathbf{O})} \\ \hat{\psi}^k(l) &= \frac{\sum_{t=1}^{T+1} P(x_t^k = l|\mathbf{O})}{T+1} \end{aligned}$$

F.3 Modified PT-FHMM Transition Reestimation Equations

$$\hat{P}(j^k|I) = \frac{\sum_{t=2}^{T+1} P(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O})}{\sum_{t=2}^{T+1} P(\mathbf{S}_{t-1} = I|\mathbf{O})}$$

G

Efficient Calculations for PT-FHMM Partitioning Procedure

The shortcuts used here are similar to those taken in standard decision tree state clustering schemes eg. [122, 123].

Assume the model to be tied is λ and that posteriors $P(\mathbf{S}_t = J|\mathbf{O})$ have been calculated based on λ in the standard fashion. We write $\gamma_t(J) = p(\mathbf{S}_t = J|\mathbf{O})$ and $\eta_t^k(j^k, I) = p(s_t^k = j^k, \mathbf{S}_{t-1} = I|\mathbf{O})$.

Several calculations that seemingly require the accumulation of likelihoods for each point in the training set will be made more efficient via sufficient statistics, as shown below. These statistics will be the state occupancies $\gamma(J) = \sum_t \gamma_t(J)$ and the parameters of $\hat{\lambda}_{untied}$, ie. the set of parameters $\{\hat{\mu}_J^k, \hat{\Sigma}_J^k, \hat{a}^k(j^k|I)\}$ which result from performing an EM M-Step update on λ, without introducing tying, based on posteriors calculated using λ.

The calculations can also be made more efficient through the use of *sum occupancy* statistics. For clustering transition-related distributions useful sum-occupancy statistics are:

- $\eta^k(j^k, I) \stackrel{\text{def}}{=} \sum_t \eta_t^k(j^k, I) = \hat{a}^k(j^k|I)\gamma(I)$
- $\eta^k(j^k, C^{trans,k}) \stackrel{\text{def}}{=} \sum_{I \in C^{trans,k}} \sum_t P(s_t^k = j^k, \mathbf{S}_t = I|\mathbf{O}) = \sum_{I \in C^{trans,k}} \eta^k(j^k, I)$
- $\gamma(C^{trans,k}) \stackrel{\text{def}}{=} \sum_{I \in C^{trans,k}} \sum_t P(\mathbf{S}_t = I|\mathbf{O}) = \sum_{I \in C^{trans,k}} \gamma(I)$

and for clustering observation-related distributions:

- $\gamma(C^{obs,k}) \stackrel{\text{def}}{=} \sum_{I \in C^{obs,k}} \sum_t P(\mathbf{S}_t = I|\mathbf{O}) = \sum_{I \in C^{obs,k}} \gamma(I)$

G.1 Auxiliary Function Terms

The Q_{Bk} and Q_{Ak} auxiliary function terms appear expensive to evaluate since they involve accumulating likelihoods across all data points in the training set; however, using the sufficient statistics, they simplify to

$$\begin{aligned} Q_{Bk} &= \sum_{C \in C^{obs,k}} \gamma(C) \left[-\frac{D_k}{2} \log 2\pi - \frac{1}{2} \log |\hat{\Sigma}_{C_k}| - \frac{D_k}{2} \right] \\ Q_{Ak} &= \sum_{C \in C^{trans,k}} \sum_{j^k \in \theta_k} \eta^k(j^k, C) \log \hat{a}^k(j^k|C) \end{aligned}$$

G.2 Distance Calculations in Repartitioning

The observation-related distance calculations can be expressed without recourse to the individual training set points. Because, using the sufficient statistics we have:

- $\sum_{t=1}^{T} \gamma_t(J) o_t^k = \gamma(J) \hat{\mu}_J^k$;
- $\sum_{t=1}^{T} \gamma_t(J) o_t^k (o_t^k)^T = \gamma(J)(\hat{\mu}_J^k (\hat{\mu}_J^k)^T + \hat{\Sigma}_J^k)$

and using the standard identity $x^T A x = tr(A x x^T)$ (for vector x and matrix A of appropriate dimensions), this leads to:

$$\begin{aligned}\sum_{t=1}^{T} \gamma_t(J) \log p(o_t^k | \mu_C, \Sigma_C) &= -\frac{\gamma(J) D_k \log 2\pi}{2} - \frac{\gamma(J) \log |\Sigma_C|}{2} \\ &\quad -\frac{1}{2} tr\{\Sigma_C^{-1}([\sum_{t=1}^{T} \gamma_t(J) o_t^k (o_t^k)^T] - 2\gamma(J) \hat{\mu}_J^k \mu_C^k + \mu_C^k (\mu_C^k)^T)\}\end{aligned}$$

The transition-related distance calculations may also be simplified:

$$\sum_{j^k \in \theta_k} \sum_{t=2}^{T} \eta_t(j^k, I) \log \hat{P}(j^k | C_1^{trans,k}) = \sum_{j^k \in \theta_k} \eta(j^k, I) \log \hat{P}(j^k | C_1^{trans,k})$$

G.3 Centroid Reestimation

All centroid reestimation can be implemented without recourse to the training set through the sufficient statistics (bracketed by [.]) and through sum occupancies:

$$\begin{aligned}\hat{\mu}_{C_k^{obs}}^k &= \frac{\sum_{J \in C_k^{obs}} [\sum_{t=1}^{T} \gamma_t(J) o_t^k]}{\sum_{J \in C_k^{obs}} \sum_{t=1}^{T} \gamma_t(J)} \\ &= \frac{\sum_{J \in C_k^{obs}} [\sum_{t=1}^{T} \gamma_t(J) o_t^k]}{\gamma(C_k^{obs})} \\ \hat{\Sigma}_{C_k^{obs}}^k &= \frac{\sum_{J \in C_k^{obs}} \sum_{t=1}^{T} \gamma_t(J)(o_t^k - \hat{\mu}_{C_k^{obs}}^k)(o_t^k - \hat{\mu}_{C_k^{obs}}^k)^T}{\sum_{J \in C_k^{obs}} \sum_{t=1}^{T} \gamma_t(J)} \\ &= \frac{\sum_{J \in C_k^{obs}} [\sum_{t=1}^{T} \gamma_t(J) o_t^k (o_t^k)^T]}{\gamma(C_k^{obs})} - \hat{\mu}_{C_k^{obs}}^k (\hat{\mu}_{C_k^{obs}}^k)^T \\ p(j^k | C_k^{trans}) &= \frac{\sum_{I \in C_k^{trans}} \sum_{t=2}^{T} \eta_t^k(j^k, I)}{\sum_{I \in C_k^{trans}} \sum_{t=2}^{T} \gamma_t(I)} \\ &= \frac{\sum_{I \in C_k^{trans}} \eta^k(j^k, I)}{\gamma(C_k^{trans})}\end{aligned}$$

H

Proof: Suboptimality of Chain Viterbi Algorithm

The following model demonstrates that Chain Viterbi decoding, initialized for each stream k with the Viterbi path for stream k decoded in isolation, is not guaranteed to converge to the Viterbi meta-state sequence.

The model has two observation streams, each comprising 1-dimensional observations. It has two state chains, both with 3 states, indexed by $2^1, 3^1, 4^1$ for chain 1 and $2^2, 3^2, 4^2$ for chain 2 where 4^1 and 4^2 are non-emitting exit states as discussed in Appendix F. The priors are $\pi^{11}(2^1) = 1.0$, $\pi^{12}(2^1) = 1.0$, $\pi^{21}(2^2) = 1.0$ and $\pi^{22}(2^2) = 1.0$; all other states have prior probability zero. The transition matrices are

- a^{11} = [0.4 0.6 0.0; 0.6 0.39 0.01; 0.0 0.0 0.0];
- a^{12} = [0.9 0.1 0.0; 0.1 0.89 0.01; 0.0 0.0 0.0];
- a^{21} = [0.8 0.2 0.0; 0.2 0.79 0.01; 0.0 0.0 0.0];
- a^{22} = [0.49 0.51 0.0; 0.49 0.50 0.01; 0.0 0.0 0.0].

All observation distributions have mean $\mu = 1$ and variance $\sigma = 100$. The mixed memory weights are ϕ=[0.5 0.5; 0.5 0.5] and ψ=[0.01 0.99; 0.99 0.01].

I

Mean-Field Variational Learning Algorithm

I.1 Mean-Field Variational Lower Bound

Under the mean-field approximation, the lower bound can be written:

$$
\begin{aligned}
\mathcal{L}_Q(\Psi,\lambda) &= \sum_{k=1}^{K}\sum_{j\in\Theta_k} Q_1^{Sk}(j)\ln\pi^{kk}(j) + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K} Q_t^{Yk}(l)\ln\phi^{k}(l) \\
&+ \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K}\sum_{i\in\Theta_l} Q_t^{Yk}(l)Q_t^{Sl}(i)\ln b^{kl}(o_t^k|i) \\
&+ \sum_{t=2}^{T}\sum_{k=1}^{K}\sum_{i\in\Theta_k}\sum_{j\in\Theta_k} Q_t^{Sk}(j)Q_{t-1}^{Sk}(i)\ln a^{kk}(j|i) \\
&+ \sum_{k=1}^{K}\sum_{i\in\Theta_k} Q_T^{Sk}(i)\ln a^{kk}(N^k|i) \\
&- \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i\in\Theta_k} Q_t^{Sk}(i)\ln Q_t^{Sk}(i) - \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K} Q_t^{Yk}(l)\ln Q_t^{Yk}(l)
\end{aligned}
$$

I.2 Step 1: Minimizing KL Divergence

I.2.1 Exact Posterior Distribution

The exact posterior distribution over hidden variables for the MM-FHMM can be written in the form:

$$
p(\mathbf{S},\mathbf{Y}|\mathbf{O},\lambda) = \frac{\exp(H(\mathbf{S},\mathbf{Y},\mathbf{O}))}{Z}
$$

where $Z = p(\mathbf{O})$ is a normalization constant and

$$\begin{aligned} H(\mathbf{S},\mathbf{Y},\mathbf{O}) &= \ln p(\mathbf{S},\mathbf{Y},\mathbf{O}) \\ &= \sum_{k=1}^{K}\sum_{j\in\Theta_k}\delta(s_1^k,j)\ln\pi^{kk}(j) + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K}\delta(y_t^k,l)\ln\phi^k(l) \\ &+ \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K}\sum_{i\in\Theta_l}\delta(y_t^k,l)\delta(s_t^l,i)\ln b^{kl}(o_t^k|i) \\ &+ \sum_{t=2}^{T}\sum_{k=1}^{K}\sum_{i\in\Theta_k}\sum_{j\in\Theta_k}\delta(s_t^k,j)\delta(s_{t-1}^k,i)\ln a^{kk}(j|i) \\ &+ \sum_{k=1}^{K}\sum_{i\in\Theta_k}\delta(s_T^k,i)\ln a^{kk}(N^k|i) \end{aligned}$$

I.2.2 Variational Approximation

The variational approximation can be written in the form:

$$Q(\mathbf{S},\mathbf{Y}|\Psi) = \frac{\exp H_Q(\mathbf{S},\mathbf{Y}|\Psi)}{Z_Q}$$

where

$$Z_Q = \prod_{t=1}^{T}\prod_{k=1}^{K}(\sum_{i\in\Theta_k}\exp\Psi_{ti}^{Sk})(\sum_{l=1}^{K}\exp\Psi_{tl}^{Yk})$$

is a normalization constant and

$$H_Q = \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j\in\Theta_k}\delta(s_t^k,j)\Psi_{tj}^{Sk} + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K}\delta(y_t^k,l)\Psi_{tl}^{Yk}$$

I.2.3 KL Divergence

The KL divergence can now be written as:

$$KL[Q(\mathbf{S},\mathbf{Y}|\Psi)||p(\mathbf{S},\mathbf{Y}|\mathbf{O},\lambda)] = \mathcal{E}_Q H_Q(\mathbf{S},\mathbf{Y}) - \ln Z_Q - \mathcal{E}_Q H + \ln Z$$

This expression can be minimized wrt Ψ using basic gradient descent; the next section provides derivatives which are useful for finding the gradient of the KL Divergence in Subsection I.2.5.

I.2.4 Useful Derivatives

The following derivatives will be useful in finding the gradient of the KL Divergence:

$$\frac{\partial Q_t^{Sk}(j)}{\partial\Psi_{\bar{t}\bar{j}}^{S\bar{k}}} = \frac{\partial}{\partial\Psi_{\bar{t}\bar{j}}^{S\bar{k}}}\left[\frac{\exp\Psi_{tj}^{Sk}}{\sum_{i\in\Theta_k}\exp\Psi_{ti}^{Sk}}\right] = \begin{cases} -Q_{\bar{t}}^{S\bar{k}}(j)Q_{\bar{t}}^{S\bar{k}}(\bar{j}) & j\neq\bar{j}, t=\bar{t}, k=\bar{k} \\ Q_{\bar{t}}^{S\bar{k}}(\bar{j})[1-Q_{\bar{t}}^{S\bar{k}}(\bar{j})] & j=\bar{j}, t=\bar{t}, k=\bar{k} \\ 0 & otherwise \end{cases}$$

$$\frac{\partial Q^{Yk}_t(l)}{\partial \Psi^{Y\bar{k}}_{t\bar{l}}} = \frac{\partial}{\partial \Psi^{Y\bar{k}}_{t\bar{l}}}\left[\frac{\exp \Psi^{Yk}_{tl}}{\sum_\nu \exp \Psi^{Yk}_{t\nu}}\right] = \begin{cases} -Q^{Y\bar{k}}_{\bar{t}}(l)Q^{Y\bar{k}}_{\bar{t}}(\bar{l}) & l \neq \bar{l}, t = \bar{t}, k = \bar{k} \\ Q^{Y\bar{k}}_{\bar{t}}(\bar{l})[1 - Q^{Y\bar{k}}_{\bar{t}}(\bar{l})] & l = \bar{l}, t = \bar{t}, k = \bar{k} \\ 0 & otherwise \end{cases}$$

$$\frac{\partial (Q^{Sk}_t(j)\Psi^{Sk}_{tj})}{\partial \Psi^{S\bar{k}}_{\bar{t}\bar{j}}} = \begin{cases} -\Psi^{S\bar{k}}_{\bar{t}j}Q^{S\bar{k}}_{\bar{t}}(j)Q^{S\bar{k}}_{\bar{t}}(\bar{j}) & j \neq \bar{j}, t = \bar{t}, k = \bar{k} \\ Q^{S\bar{k}}_{\bar{t}}(\bar{j})(\Psi^{S\bar{k}}_{\bar{t}\bar{j}} + 1) - (Q^{S\bar{k}}_{\bar{t}}(\bar{j}))^2\Psi^{S\bar{k}}_{\bar{t}\bar{j}} & j = \bar{j}, t = \bar{t}, k = \bar{k} \\ 0 & otherwise \end{cases}$$

$$\frac{\partial (Q^{Yk}_t(j)\Psi^{Yk}_t(l))}{\partial \Psi^{Y\bar{k}}_{\bar{t}\bar{l}}} = \begin{cases} -\Psi^{Y\bar{k}}_{\bar{t}l}Q^{Y\bar{k}}_{\bar{t}}(l)Q^{Y\bar{k}}_{\bar{t}}(\bar{l}) & l \neq \bar{l}, t = \bar{t}, k = \bar{k} \\ Q^{Y\bar{k}}_{\bar{t}}(\bar{l})(\Psi^{Y\bar{k}}_{\bar{t}\bar{l}} + 1) - (Q^{Y\bar{k}}_{\bar{t}}(\bar{l}))^2\Psi^{Y\bar{k}}_{\bar{t}\bar{l}} & l = \bar{l}, t = \bar{t}, k = \bar{k} \\ 0 & otherwise \end{cases}$$

I.2.5 Derivatives of KL Divergence

The derivatives of expressions comprising the KL Divergence are:

$$\mathcal{E}_Q H_Q - \ln Z_Q =$$
$$\sum_{t=1}^{T}\sum_{k=1}^{K}\left[\sum_{j\in\Theta_k} Q^{Sk}_t(j)\Psi^{Sk}_{tj} + \sum_{l=1}^{K} Q^{Yk}_t(l)\Psi^{Yk}_{tl} - \ln(\sum_{j\in\Theta_k} \exp \Psi^{Sk}_{tj}) - \ln(\sum_{l=1}^{K} \exp \Psi^{Yk}_{tl})\right]$$

and:

$$\begin{aligned} \frac{\partial(\mathcal{E}_Q H_Q - \ln Z_Q)}{\partial \Psi^{Y\bar{k}}_{\bar{t}\bar{j}}} &= Q^{Y\bar{k}}_{\bar{t}}(\bar{l})\left[(\sum_{l=1}^{K} -Q^{Y\bar{k}}_{\bar{t}}(l)\Psi^{Y\bar{k}}_{\bar{t}l}) + \Psi^{Y\bar{k}}_{\bar{t}\bar{l}}\right] \\ \frac{\partial(\mathcal{E}_Q H_Q - \ln Z_Q)}{\partial \Psi^{S\bar{k}}_{\bar{t}\bar{j}}} &= Q^{S\bar{k}}_{\bar{t}}(\bar{j})\left[(\sum_{j\in\Theta_{\bar{k}}} -Q^{S\bar{k}}_{\bar{t}}(j)\Psi^{S\bar{k}}_{\bar{t}j}) + \Psi^{S\bar{k}}_{\bar{t}\bar{j}}\right] \end{aligned}$$

$$\begin{aligned} \mathcal{E}_Q H &= \sum_{k=1}^{K}\sum_{j\in\Theta_k} Q^{Sk}_1(j)\ln \pi^{kk}(j) + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K} Q^{Yk}_t(l)\ln \phi^k(l) \\ &+ \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{K}\sum_{i\in\Theta_l} Q^{Yk}_t(l)Q^{Sl}_t(i)\ln b^{kl}(o^k_t|i) \\ &+ \sum_{t=2}^{T}\sum_{k=1}^{K}\sum_{i\in\Theta_k}\sum_{j\in\Theta_k} Q^{Sk}_t(j)Q^{Sk}_{t-1}(i)\ln a^{kk}(j|i) \\ &+ \sum_{k=1}^{K}\sum_{i\in\Theta_k} Q^{Sk}_T(i)\ln a^{kk}(N^k|i) \end{aligned}$$

$$\begin{aligned}
\frac{\partial(\mathcal{E}_Q H)}{\partial \Psi^{Y\bar{k}}_{\bar{t}\bar{l}}} &= Q^{Y\bar{k}}_{\bar{t}}(\bar{l}) \sum_{l \neq \bar{l}} -Q^{Y\bar{k}}_{\bar{t}}(l) \left[\ln \phi^{\bar{k}}(l) + \sum_{i \in \Theta_l} Q^{Sl}_{\bar{t}}(i) \ln b^{\bar{k}l}(o^{\bar{k}}_{\bar{t}}|i) \right] \\
&+ Q^{Y\bar{k}}_{\bar{t}}(\bar{l})(1 - Q^{Y\bar{k}}_{\bar{t}}(\bar{l})) \left[\ln \phi^{\bar{k}}(\bar{l}) + \sum_{i \in \Theta_{\bar{l}}} Q^{S\bar{l}}_{\bar{t}}(i) \ln b^{\bar{k}\bar{l}}(o^{\bar{k}}_{\bar{t}}|i) \right]
\end{aligned}$$

For $1 < t < T$:

$$\begin{aligned}
&\frac{\partial(\mathcal{E}_Q H)}{\partial \Psi^{S\bar{k}}_{\bar{t}\bar{j}}} = \\
&\quad Q^{S\bar{k}}_{\bar{t}}(\bar{j}) \sum_{i \neq \bar{j} \in \Theta_{\bar{k}}} -Q^{S\bar{k}}_{\bar{t}}(i) \left[\sum_{k=1}^{K} Q^{Yk}_{\bar{t}}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{\bar{t}}|i) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{\bar{t}-1}(j) \ln a^{\bar{k}\bar{k}}(i|j) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{\bar{t}+1}(j) \ln a^{\bar{k}\bar{k}}(j|i) \right] \\
&+ Q^{S\bar{k}}_{\bar{t}}(\bar{j})(1 - Q^{S\bar{k}}_{\bar{t}}(\bar{j})) \left[\sum_{k=1}^{K} Q^{Yk}_{\bar{t}}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{\bar{t}}|\bar{j}) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{\bar{t}-1}(j) \ln a^{\bar{k}\bar{k}}(\bar{j}|j) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{\bar{t}+1}(j) \ln a^{\bar{k}\bar{k}}(j|\bar{j}) \right]
\end{aligned}$$

And finally

$$\begin{aligned}
\frac{\partial(\mathcal{E}_Q H)}{\partial \Psi^{S\bar{k}}_{1\bar{j}}} &= Q^{S\bar{k}}_{1}(\bar{j}) \sum_{j \neq \bar{j} \in \Theta_{\bar{k}}} -Q^{S\bar{k}}_{1}(j) \left[\ln \pi^{\bar{k}\bar{k}}(j) + \sum_{k=1}^{K} Q^{Yk}_{1}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{1}|j) + \sum_{i \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{2}(i) \ln a^{\bar{k}\bar{k}}(i|j) \right] \\
&+ Q^{S\bar{k}}_{1}(\bar{j})(1 - Q^{S\bar{k}}_{1}(\bar{j})) \left[\ln \pi^{\bar{k}\bar{k}}(\bar{j}) + \sum_{k=1}^{K} Q^{Yk}_{1}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{1}|\bar{j}) + \sum_{i \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{2}(i) \ln a^{\bar{k}\bar{k}}(i|\bar{j}) \right]
\end{aligned}$$

$$\begin{aligned}
&\frac{\partial(\mathcal{E}_Q H)}{\partial \Psi^{S\bar{k}}_{T\bar{j}}} = \\
&\quad Q^{S\bar{k}}_{T}(\bar{j}) \sum_{i \neq \bar{j} \in \Theta_{\bar{k}}} -Q^{S\bar{k}}_{T}(i) \left[\sum_{k=1}^{K} Q^{Yk}_{T}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{T}|i) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{T-1}(j) \ln a^{\bar{k}\bar{k}}(i|j) + \ln a^{\bar{k}\bar{k}}(N^{\bar{k}}|i) \right] \\
&+ Q^{S\bar{k}}_{T}(\bar{j})(1 - Q^{S\bar{k}}_{T}(\bar{j})) \left[\sum_{k=1}^{K} Q^{Yk}_{T}(\bar{k}) \ln b^{k\bar{k}}(o^{k}_{T}|\bar{j}) + \sum_{j \in \Theta_{\bar{k}}} Q^{S\bar{k}}_{T-1}(j) \ln a^{\bar{k}\bar{k}}(\bar{j}|j) + \ln a^{\bar{k}\bar{k}}(N^{\bar{k}}|\bar{j}) \right]
\end{aligned}$$

I.3 Step 2: Maximization Step

Using a derivation similar to that in Appendix C, the maximizing λ is:

$$\begin{aligned}
\hat{\phi}^{k}(l) &= \frac{\sum_t Q^{Yk}_{t}(l|\Psi^{kK}_{t})}{T} \\
\hat{\mu}^{kl}_{j} &= \frac{\sum_t Q^{Yk}_{t}(l|\Psi^{Yk}_{t}) Q^{Sk}_{t}(j|\Psi^{Sk}_{t}) o^{k}_{t}}{\sum_t Q^{Yk}_{t}(l|\Psi^{Yk}_{t}) Q^{Sk}_{t}(j|\Psi^{Sk}_{t})} \\
\hat{\Sigma}^{kl}_{j} &= \frac{\sum_t Q^{Yk}_{t}(l|\Psi^{Yk}_{t}) Q^{Sk}_{t}(j|\Psi^{Sk}_{t}) (o^{k}_{t} - \hat{\mu}^{kl}_{j})(o^{k}_{t} - \hat{\mu}^{kl}_{j})^{T}}{\sum_t Q^{Yk}_{t}(l|\Psi^{Yk}_{t}) Q^{Sk}_{t}(j|\Psi^{Sk}_{t})}
\end{aligned}$$

J

Directed Acyclic Graphical Models (Bayesian Networks)

This section reviews background necessary for interpreting the graphical representation of models used in this dissertation and does so without proof. More rigorous introductions to the theory and practice of graphical models are found in [75, 94, 174].

A *Directed Acyclic Graphical Model* (DAGM) or *Bayesian Network* (BN) is a graphical statement of conditional independence relations amongst a set of random variables $\mathcal{X} = \{X_1, \dots, X_N\}$. It comprises structure and implementation. The structure is a directed, acyclic graph $\mathcal{G} = (V, E)$. There is a one-to-one mapping between nodes in V and the set $\mathcal{X}$; E denotes the set of directed edges in $\mathcal{G}$. The structure of a particular graph $\mathcal{G}$ implies a set of conditional independence relationships or *Markov properties* amongst the variables in $\mathcal{X}$. These Markov properties can be read directly from $\mathcal{G}$, using the property of *d-separation*. Two sets of variables A and B are *d-separated* by a set of variables S if and only if all paths that connect any node in A to any node in B have the following property: there is a node v in the path satisfying

- $v \in S$ and the arrows along the path do not converge at v;
- $v \notin S$, any descendant of v is not in S and the arrows along the path converge at v.

The d-separation of A and B by S implies that the variables in A are conditionally independent of the variables in B given the variables in S. The resulting set of conditional independence statements are often referred to as the *global* Markov properties[1]. Equivalently, the same set of conditional independence properties can be read from $\mathcal{G}$ in terms of the *local* Markov properties: each variable in $\mathcal{G}$ is conditionally independent of its non-descendants given its parents.

A particular graph $\mathcal{G}$ is compatible with a set of probability models, each obeying the corresponding local or global Markov properties. Thus any probability model $p(X_1, \dots, X_N)$ compatible with $\mathcal{G}$ may be factored as follows. Letting $pa(X_i)$ denote the set of *parents*[2] of node X_i, then

$$p(X_1, \dots, X_N) = \prod_{n=1}^{N} p(X_n | pa(X_n))$$

The implementation of a DAGM $\mathcal{G}$ is determined by the forms assumed for the conditional probability distributions $p(X_n|pa(X_n))$.

A *Dynamic Bayesian Network* (DBN) is a directed graphical model of a temporal process, where the set of random variables to be modelled is $\mathcal{X} = \{X_t^k | 1 \leq k \leq K, 1 \leq$

[1] A more precise presentation would first define the global Markov properties and then show these are equivalent to those implied by d-separation.

[2] When an edge is directed from a node i to a node j, then node i is said to be a *parent* of node j.

$t \leq T\}$. As before, conditional independence relationships can be read directly from the graphical representation and the joint distribution for these variables is given by $\prod_{t=1}^{T} \prod_{k=1}^{K} p(X_t^k | pa(X_t^k))$. All DBNs discussed in this dissertation satisfy the first order Markov property ie. the parents of a variable in timeslice t occur in timeslice t or $t-1$. Therefore, the corresponding graph can be completely specified by the nodes at $t = 1$, $t = 2$, and the links between them: the network structure for each $t > 2$ is formed by replicating the edges between the set of variables at $t = 1$, $t = 2$ between the new set of variables at $t - 1$, t.

For speech modelling, some variables in the set $\mathbf{X}$ are *observed* and some are *hidden*. Nodes corresponding to observed variables are shaded.

K

Switchboard Transcription Project Phone Set

The phone set used by the Switchboard Transcription Project is reproduced below. The information was released with the transcriptions (`http://www.icsi.berkeley.edu/real/stp`).

Vowels (17)	
iy	’beat’
ih	’bit’
ey	’bait’
eh	’bet’
ae	’bat’
ux	high, front, rounded allophone of /uw/ as in ’suit’
ix	high, central vowel (unstressed), as in ’roses’
ax	mid, central vowel (unstressed), as in ’the’
ah	mid, central vowel (stressed), as in ’butt’
uw	’boot’
uh	’book’
ao	’bought’
aa	’cot’
ay	’bite’
oy	’boy’
aw	’bough’
ow	’boat’
Liquids (4)	
l	’led’
_dl	velarized l e.g., ’all’ ao l_dl
r	’red’
Glides (2)	
y	’yet’
w	’wet’
hh w	’what’ - not a separate symbol but a concatenation of two separate transcription symbols to distinquish from the ”plain vanilla” [w]
Syllabic resonants (6)	
er	’bird’
axr	unstressed allophone of /er/, as in ’diner’
el	syllabic allophone of /l/, as in ’bottle’
em	syllabic allophone of /m/, as in ’yes ’em’ (’yes ma’am’)
en	syllabic allophone of /n/, as in ’button’
eng	syllabic allophone of /ng/, as in ’Washington’ (uncommon)
Nasals (3)	
m	’mom’ (nasal stop)
n	’non’ (nasal stop)
ng	’sing’ (nasal stop - only occurs in syllable-final position in English)
Affricates (2)	
ch	’church’
jh	’judge’

Stops (13)	
p	’pop’
b	’bob’
t	’tot’
d	’dad’
k	’kick’
g	’gag’
pcl	closure associated with stop
bcl	”
tcl	”
dcl	”
kcl	”
gcl	”
q	glottal stop - allophone of /t/, as in ’Atlanta’ where the first /t/ can be realized as [q]. Also may occur between words in continuous speech, especially at vowel-vowel boundaries, and at the beginning of vowel-initial utterances.
* _vd	voicing, either partial or complete, of a normally voiceless segment
Fricatives (9)	
f	’fief’
v	’verv’
th	’thief’
dh	’they’
s	’sis’
z	’zoo’
sh	’shoe’
zh	’measure’
hh	’hay’
_vd	voicing, either partial or complete, of a normally voiceless segment
Flaps and trills (3)	
* dx_d	alveolar flap (allophone of [d])
* dx_t	alveolar flap (allophone of [t])
nx	nasal flap (allophone of [n])
Non-speech (2)	
pau	silence within an utterance that does not correspond to the closure for a stop or affricate; usually audible at sentence level
h#	non-speech event(s)
Other Diacritics (3)	
_cl	assigned to the glottal stop (q) when a full closure
_cr	creaky voice
* _epi	closure resulting from coarticulation of fricative and nasal or lateral. eg. t_epi, p_epi. Distinguished from the Arpabet symbolization by attachment to a specific phone segment
* ?	unknown speech sound
* _!	unusual speech pattern - deviates significantly from normal e.g., weird stress, pronunciation, etc.
* _#	truncated segment (as when it has been prematurely cut by the the computer segmenter)
Feature diacritics (8)	
_fr	fricated of a usually non-fricated segment
_h	aspirated of a usually non-aspirated segment
_n	nasalized of a usually non-nasalized segment
_vd	voicing, either partial or complete, of a normally voiceless segment
_vl	devoicing, either partial or complete, of a normally voiced segment
_gl	glide portion of a dipthong (used when adjacent vowel nucleus)
_tr	vowel transition between preceeding and following vowel

Bibliography

[1] SM Ahadi and PC Woodland. Rapid speaker adaptation using model prediction. In *Proceedings of ICASSP*, pages 684–687, 1995.

[2] AMA Ali, J Van der Spiegel, P Mueller, G Haentjens, and J Berman. An acoustic-phonetic feature-based system for automatic phoneme recognition in continuous speech. In *Proceedings of ISCAS*, pages 118–121, 1999.

[3] A Andreou, T Kamm, and J Cohen. Experiments in vocal tract normalization. In *Proceedings of CAIP Workshop: Frontiers in Speech Recognition II*, 1994.

[4] T Applebaum and B Hanson. Regression features for recognition of speech in noise. In *Proceedings of ICASSP*, page S14.26, 1991.

[5] GM Ayers. Discourse functions of pitch range in spontaneous and read speech. In *Working Papers In Linguistics No 44*, pages 1–49. Ohio State University, 1994.

[6] M Bacchiani. *Speech Recognition System Design Based On Automatically Derived Units*. PhD thesis, Boston University, MA, USA, 1999.

[7] LR Bahl, PF Brown, PV de Souza, RL Mercer, and MA Picheny. Acoustic markov models used in the tangora speech recognition system. In *Proceedings of ICASSP*, pages 497–501, 1988.

[8] LR Bahl, PF Brown, de Souza PV, and RL Mercer. A tree-based statistical language model for natural language speech recognition. *IEEE Transactions ASSP*, 37(7):1001–1008, 1989.

[9] LR Bahl, F Jelinek, and RL Mercer. A maximum-likelihood approach to continuous speech recognition. *IEEE Transactions PAMI*, 5(2):179–90, 1983.

[10] LE Baum, T Petrie, G Soules, and N Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[11] J Bellegarda. Exploiting latent semantic information in statistical language modelling. *Proceedings of IEEE*, 88(8):1279–1296, 2000.

[12] J Bernstein, G Baldwin, M Cohen, H Murveit, and M Weintraub. Phonological studies for speech recognition. In *Proceedings of the DARPA Speech Recognition Workshop*, 1986.

[13] J Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, 48(3):259–302, 1986.

[14] D Biber. *Variations Across Speech and Writing*. Cambridge University Press, 1988.

[15] E Blaauw. Phonetic differences between read and spontaneous speech. In *Proceedings of ICSLP*, pages 751–754, 1992.

[16] H Bourlard, S Dupont, and C Ris. Multi-stream speech recognition. Technical Report IDIAP-RR 96-07, IDIAP, 1996.

[17] M Brand, N Oliver, and A Pentland. Coupled hidden markov models for complex action recognition. In *Proceedings of IEEE CVPR*, pages 994–999, 1997.

[18] PF Brown. *The Acoustic-Modeling Problem In Automatic Speech Recognition*. PhD thesis, IBM TJ Watson Research Center, NY, USA, 1987.

[19] J Butzberger, H Murveit, E Shriberg, and P Price. Spontaneous speech effects in large vocabulary speech recognition applications. In *Proceedings of DARPA Speech and Natural Language Workshop*, pages 339–343, 1992.

[20] W Byrne, M Finke, S Khudanpur, J McDonough, H Nock, M Riley, M Saraclar, C Wooters, and G Zavaliagkos. Pronunciation modelling using a hand-labelled corpus for conversational speech recognition. In *Proceedings of ICASSP*, pages 313–316, 1998.

[21] Lin Lawrance Chase. *Error-Responsive Feedback Mechanisms for Speech Recognizers*. PhD thesis, Carnegie Mellon University, PA, USA, 1997. Also available as a technical report, CMU-RI-TR-97-18.

[22] C Chelba. *Exploiting Syntactic Structure for Natural Language Modeling*. PhD thesis, The Johns Hopkins University, MD, USA, 2000.

[23] Francine R. Chen. Identification of contextual factors for pronunciation networks. In *Proceedings of ICASSP*, pages 753–756, 1990.

[24] S Chen and J Goodman. An empirical study of smoothing techniques for language modelling. *Computer Speech and Language*, 13(4):359–394, 1999.

[25] PR Clarkson. *Adaptation of Statistical Language Models for Automatic Speech Recognition*. PhD thesis, Cambridge University Engineering Dept, Cambridge, UK, 1999.

[26] Michael Harris Cohen. *Phonological Structures for Speech Recognition*. PhD thesis, Computer Science Division, Department of Electrical Engineering and Computer Science, University of California, CA, USA, 1989.

[27] R Cole, Y Muthusamy, and M Fanty. The isolet spoken letter database. Technical Report CSE 90-004, OGI, 1990.

[28] G Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 42(2-3):393–405, 1990.

[29] T Cover and J Thomas. *Elements Of Information Theory*. John Wiley and Sons, 1991.

[30] P Dalsgaard. Phoneme label alignment using acoustic-phonetic features and gaussian probability density functions. *Computer Speech And Language*, 6:303–329, 1992.

[31] K Daoudi, D Fohr, and C Antoine. A new approach for multi-band speech recognition based on probabilistic graphical models. In *Proceedings of ICSLP*, pages I:329–332, 2000.

[32] SB Davis and P Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions ASSP*, 28(4):357–366, 1980.

[33] AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[34] L Deng and K Erler. Structural design of a hidden markov model based speech recognizer using multi-valued phonetic features: Comparison with segmental speech units. *Journal of the Acoustical Society of America*, 92(92):3058–3067, 1992.

[35] VV Digalakis, D Rtischev, and LG Neumeyer. Speaker adaptation using constrained estimation of gaussian mixtures. *IEEE Transactions SAP*, 3(5):357–366, 1995.

[36] E Eide. Automatic modeling of pronunciation variations. In *Proceedings of Eurospeech*, pages 451–454, 1999.

[37] E Eide and H Gish. A parametric approach to vocal tract length normalization. In *Proceedings of ICASSP*, pages 346–348, 1996.

[38] E Eide, H Gish, P Jeanrenaud, and A Mielke. Understanding and improving speech recognition performance through the use of diagnostic tools. In *Proceedings of ICASSP*, 1995.

[39] E Eide, JR Robin Rohlicek, H Gish, and S Mitter. A linguistic feature representation of the speech waveform. In *Proceedings of ICASSP*, pages 483–486, 1993.

[40] M Eskenazi. Changing speech styles: Strategies in read speech and casual and careful spontaneous speech. In *Proceedings of ICSLP*, pages 755–758, 1992.

[41] M Finke, J Fritsch, D Koll, and A Waibel. Modeling and efficient decoding of large vocabulary conversational speech. In *Proceedings of Eurospeech*, pages 467–470, 1999.

[42] Michael Finke and Alex Waibel. Speaking mode dependent pronunciation modeling in large vocabulary conversational speech recognition. In *Proceedings of Eurospeech*, pages 2379–2382, 1997.

[43] JG Fiscus. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover). In *Proceedings of IEEE Workshop ASRU*, 1997.

[44] JE Fosler-Lussier. *Dynamic Pronunciation Models For Automatic Speech Recognition*. PhD thesis, ICSI, UC Berkeley, CA, USA, 1999.

[45] J Frankel, K Richmond, S King, and P Taylor. An automatic speech recognition system using neural networks and linear dynamic models to recover and model articulatory traces. In *Proceedings of ICSLP*, pages IV:254–257, 2000.

[46] S Furui. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions ASSP*, 34(1):52–59, 1986.

[47] M Gales, D Pye, and P Woodland. Variance compensation within the MLLR framework for robust speech recognition and speaker adaptation. In *Proceedings of ICSLP*, pages 1832–1835, 1996.

[48] MJF Gales. *Model-Based Techniques For Noise Robust Speech Recognition*. PhD thesis, Cambridge University Engineering Dept, Cambridge, UK, 1995.

[49] MJF Gales. The generation and use of regression class trees for mllr adaptation. Technical Report CUED/F-INFENG/TR263, Cambridge University Engineering Department, 1996.

[50] A Ganapathiraju, V Goel, J Picone, A Corrada, G Doddington, K Kirchhoff, M Ordowski, and B Wheatley. Syllable - a promising recognition unit for lvcsr. In *Proceedings of IEEE Workshop ASRU*, 1997.

[51] J-L Gauvain and C-H Lee. Map estimation of continuous density hmm: Theory and applications. In *Proceedings of DARPA Speech And Natural Language Workshop*, 1992.

[52] Z Ghahramani and MI Jordan. Factorial hidden markov models. *Machine Learning*, 29:245–273, 1997.

[53] L Gillick and SJ Cox. Some statistical issues in the comparison of speech recognition algorithms. In *Proceedings of ICASSP*, pages 532–535, 1989.

[54] John J Godfrey, Edward C Holliman, and Jane McDaniel. Switchboard: Telephone speech corpus for research and development. In *Proceedings of ICASSP*, pages 517–520, 1992.

[55] V Goel, Byrne WJ, and Khudanpur S. Lvcsr rescoring with modified loss functions: A decision theoretic perspective. In *Proceedings of ICASSP*, pages 425–428, 1998.

[56] JA Goldsmith, editor. *Phonological Theory: The Essential Readings*. Blackwell, 1999.

[57] D Graff. The 1996 broadcast news speech and language-model corpus. In *Proceedings of DARPA Speech Recognition Workshop*, 1997.

[58] S Greenberg. The switchboard transcription project. Technical report, The Johns Hopkins University (Center for Language and Speech Processing) Summer Research Workshop, 1995. `http://www.icsi.berkeley.edu/real/stp`.

[59] S Greenberg, S Chang, and J Hollenback. An introduction to the diagnostic evaluation of switchboard-corpus automatic speech recognition systems. In *Proceedings of Speech Transcription Workshop*, 2000.

[60] Steven Greenberg, Joy Hollenback, and Dan Ellis. Insights into spoken language gleaned from phonetic transcription of the switchboard corpus. In *Proceedings of ICSLP*, pages S24–27, 1996.

[61] A Gunawardana and W Byrne. Discounted likelihood linear regression for rapid speaker adaptation. *Computer Speech and Language*, 15(1):15–38, 2001.

[62] T Hain and PC Woodland. Dynamic hmm selection for continuous speech recognition. In *Proceedings of Eurospeech*, pages 532–535, 1999.

[63] T Hain and PC Woodland. Modelling sub-phone insertions and deletions in continuous speech recognition. In *Proceedings of ICSLP*, pages IV:172–175, 2000.

[64] T Hain, PC Woodland, G Evermann, and D Povey. The cu-htk march 2000 hub5e transcription system. In *Proceedings of Speech Transcription Workshop*, 2000.

[65] H Hermansky, S Tibrewala, and M Pavel. Towards asr on partially corrupted speech. In *Proceedings of ICSLP*, pages 462–465, 1996.

[66] Nist english broadcast news transcription (hub4) benchmark test results. ftp://jaguar.ncsl.nist.gov/csr99/bn98en_official_scores_990112/readme.htm, 1999.

[67] Conversational speech transcription (hub5) benchmark test results. ftp://jaguar.ncsl.nist.gov/lvcsr/mar2001/hub5_scores_2001_03_26/readme.htm, 2001.

[68] MA Huckvale. A comparison of neural-network and hidden-markov model approaches to the tiered segmentation of speech. In *Proceedings of Institute of Acoustics Conference on Speech and Hearing*, 1992.

[69] MA Huckvale. Word recognition from tiered phonological models. In *Proceedings of Institute of Acoustics Conference on Speech and Hearing*, volume 16(5), pages 163–170, 1994.

[70] JJ Humphries and PC Woodland. Using accent-specific pronunciation modelling for improved large vocabulary continuous speech recognition. In *Proceedings of Eurospeech*, pages 2367–2370, 1997.

[71] R Iyer. Language modelling with sentence-level mixtures. Master's thesis, Boston University, MA, USA, 1994.

[72] R Iyer. *Improving And Predicting Performance of Statistical Language Models In Sparse Domains*. PhD thesis, Boston University, MA, USA, 1998.

[73] F Jelinek. *Statistical Methods For Speech Recognition*. The MIT Press, 1997.

[74] F Jelinek and R Mercer. Interpolated estimation of markov source parameters from sparse data. In ES Gelsema and LN Kanal, editors, *Pattern Recognition In Practice*, pages 381–397. Elsevier Science BV, 1980.

[75] FV Jensen. *An Introduction To Bayesian Networks*. Springer, 1996.

[76] M Jordan, Z Ghahramani, T Jaakkola, and L Saul. An introduction to variational methods for graphical models. In Michael I. Jordan, editor, *Learning in Graphical Models*, pages 105–161. Kluwer Academic Press Press, 1998.

[77] B-H Juang, W Chou, and C-H Lee. Minimum classification error rate methods for speech recognition. *IEEE Transactions SAP*, 5(3):257–265, 1997.

[78] S Kapadia. *Discriminative Training of Hidden Markov Models*. PhD thesis, Cambridge University Engineering Dept, Cambridge, UK, 1998.

[79] P Katamba. *An Introduction to Phonology*. Addison-Wesley, 1989.

[80] S Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. In *Proceedings of ICASSP*, pages 400–401, 1987.

[81] P Keating. Word-level phonetic variation in large speech corpora. In Berndt Pompino-Marschal, editor, *ZAS Working Papers In Linguistics*. 1997. Available at `http://www.humnet.ucla.edu/humnet/linguistics/people/keating/berlin1.pdf`.

[82] S King, T Stephenson, S Isard, P Taylor, and A Strachan. Speech recognition via phonetically featured syllables. In *Proceedings of ICSLP*, 1998.

[83] S King and P Taylor. Detection of phonological features in continuous speech using neural networks. *Computer Speech And Language*, 14(4):333–353, 2000.

[84] P Kingsbury, S Strassel, and C McLemore. Comlex pronouncing lexicon (renamed in 1997 release as callhome american english lexicon). Available from Linguistic Data Consortium `http://www.ldc.upenn.edu`, 1997.

[85] K Kirchhoff. Syllable-level desynchronization of phonetic features for speech recognition. In *Proceedings of ICSLP*, pages 2274–2276, 1996.

[86] K Kirchhoff. Robust speech recognition using articulatory information. Technical Report 98-036, ICSI, UC Berkeley, 1998.

[87] K Kirchhoff. *Robust Speech Recognition Using Articulatory Information*. PhD thesis, University of Bielefeld, Germany, 1999.

[88] L Knohl and A Rinscheid. Speaker normalization and adaptation using feature map selection. In *Proceedings of Eurospeech*, pages 367–370, 1993.

[89] R Kuhn and R De Mori. A cache-based natural language model for speech recognition. *IEEE Transactions PAMI*, 12(6):570–583, 1990.

[90] R Kuhn and R De Mori. Corrections to 'a cache-based natural language model for speech recognition'. *IEEE Transactions PAMI*, 14(3):691–92, 1992.

[91] G Laan. The contribution of intonation, segmental durations and spectral features. *Speech Communication*, 22:43–65, 1997.

[92] LF Lamel, RH Kassel, and S Seneff. Speech database development: Design and analysis of the acoustic-phonetic corpus. In *Proceedings of DARPA Speech Recognition Workshop*, pages 100–109, 1986.

[93] Lori Lamel and Gilles Adda. On designing pronunciation lexicons for large vocabulary, continuous speech recogntion. In *Proceedings of ICSLP*, pages 6–9, 1996.

[94] SL Lauritzen. *Graphical Models*. Oxford: Clarendon, 1996.

[95] K-F Lee. *Automatic Speech Recognition - The Development of the SPHINX System*. Kluwer Academic Press, 1989.

[96] L Lee and R Rose. Speaker normalization using efficient frequency warping procedures. In *Proceedings of ICASSP*, pages 353–356, 1996.

[97] CJ Leggetter and PC Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer Speech and Language*, 9:171–186, 1995.

[98] RG Leonard. A database for speaker-independent digit recognition. In *Proceedings of ICASSP*, pages 42.11–14, 1984.

[99] H Levin, C Schaffer, and C Snow. The prosodic and paralinguistic features of reading and telling stories. *Language And Speech*, 25(1):43–54, 1982.

[100] M Liberman. Proposal for research topic atthe johns hopkins university (center for language and speech processing) summer research workshop: Acoustic-phonetic feature detectors. Unpublished, 1998.

[101] Y Linde, A Buzo, and RM Gray. An algorithm for vector quantizer design. *IEEE Transactions Communications*, 28:84–95, 1980.

[102] B Logan and PJ Moreno. Factorial hidden markov models for speech recognition: Preliminary experiments. Technical Report 97/7, Cambridge Research Laboratory, 1997.

[103] X Luo. *Balancing Model Resolution And Generalizability In Large Vocabulary Continuous Speech Recognition*. PhD thesis, The Johns Hopkins University, MD, USA, 1999.

[104] B Mak and Y-C Tam. Asynchrony with trained transition probabilities improves performance in multi-band speech recognition. In *Proceedings of ICSLP*, pages IV:149–152, 2000.

[105] D McAllaster, L Gillick, F Scattone, and M Newman. Fabricating conversational speech data with acoustic models: A program to examine model-data mismatch. In *Proceedings of DARPA Broadcast News Workshop*, 1998.

[106] J McDonough and W Byrne. Speaker adaptation with all-pass transforms. In *Proceedings of ICASSP*, pages 757–760, 1999.

[107] P McMahon, P McCourt, and S Vaseghi. Discriminative weighting of multi-resolution sub-band cepstral features for speech recognition. In *Proceedings of ICSLP*, pages 1055–1058, 1998.

[108] N Mirghafori. *A Multi-Band Approach to Automatic Speech Recognition*. PhD thesis, ICSI, UC Berkeley, CA, USA, 1999.

[109] M Mohri, FCN Pereira, and M Riley. The design principles of a weighted finite state transducer library. *Theoretical Computer Science*, 231:17–32, 2000.

[110] Roger K Moore. Critique: The potential role of speech production models in automatic speech recognition. *Journal of the Acoustical Society of America*, 99(3):1710–1712, 1996.

[111] N Morgan, D Baron, J Edwards, D Ellis, D Gelbart, A Janin, T Pfau, S Shriberg, and A Stolcke. The meeting project at icsi. In *Notebook Proceedings of HLT (Human Language Technology) Conference*, 2001.

[112] H Murveit, J Butzberger, V Digalakis, and M Weintraub. Large vocabulary dictation using sri's decipher speech recognition system: Progressive search techniques. In *Proceedings of ICASSP*, pages 319–322, 1993.

[113] RM Neal and GE Hinton. A view of the em algorithm that justifies incremental and other variants. In Michael I. Jordan, editor, *Learning in Graphical Models*, pages 355–370. Kluwer Academic Press, 1998.

[114] C Neti, G Potamianos, J Luettin, I Matthews, Herve Glotin, D Vergyri, J Sison, J Mashari, and J Zhou. Audio-visual speech recognition. Technical report, The Johns Hopkins University (Center for Language and Speech Processing) Summer Research Workshop, 2000.

[115] H Ney and X Aubert. Dynamic programming search: From digit strings to large vocabulary word graphs. In C-H Lee, FK Soong, and KK Paliwal, editors, *Automatic Speech and Speaker Recognition*, pages 385–413. Kluwer Academic Press, 1995.

[116] H Ney, U Essen, and R Kneser. On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language*, 8(1):1–38, 1994.

[117] TR Niesler, EWD Whittaker, and PC Woodland. Comparison of part-of-speech and automatically derived category-based language models for speech recognition. In *Proceedings of ICASSP*, pages 177–180, 1998.

[118] NIST. Score package. Available from `http://www.nist.gov/speech/tools/index.htm`.

[119] P Niyogi, C Burges, and P Ramesh. Distinctive feature detection using support vector machines. In *Proceedings of ICASSP*, pages 425–428, 1999.

[120] HJ Nock and SJ Young. Detecting and correcting poor pronunciations for multiword units. In *ESCA Workshop on Modelling Pronunciation Variation for Automatic Speech Recognition*, pages 89–95, 1998.

[121] HJ Nock and SJ Young. Loosely coupled hmms for asr: A preliminary study. Technical Report CUED/F-INFENG/TR386, CUED, 2000.

[122] J Odell. *The Use of Context in Large Vocabulary Speech Recognition*. PhD thesis, Cambridge University Engineering Dept, Cambridge, UK, 1995.

[123] M Ostendorf. Incorporating linguistic theories of pronunciation variation into speech recognition models. In *Philosophical Transactions of Royal Society*, volume 358, pages 1325–1338. London, UK, 2000.

[124] M Ostendorf, B Byrne, M Bacchiani, M Finke, A Gunawardana, K Ross, S Roweis, E Shriberg, D Talkin, A Waibel, B Wheatley, and T Zeppenfeld. Modeling systematic variations in pronunciation via a language-dependent hidden speaking mode. Technical Report ECE-97-0002, Boston University, 1997.

[125] M Ostendorf and H Singer. Hmm topology design using maximum likelihood successive state splitting. *Computer Speech And Language*, 11:17–41, 1997.

[126] D Pallett, J Fiscus, G Garofolo, A Martin, and M Przybocki. Broadcast news benchmark test results. In *Proceedings of DARPA Broadcast News Workshop*, 1999.

[127] DB Paul, RP Lippmann, Y Chen, and CJ Weinstein. Robust hmm-based techniques for recognition of speech produced under stress and in noise. In *Proceedings of DARPA Speech Recognition Workshop*, 1986.

[128] Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of ICSLP*, pages 899–902, 1992.

[129] D Pye and PC Woodland. Experiments in speaker normalisation and adaptation for large vocabulary speech recognition. In *Proceedings of ICASSP*, pages 1047–1051, 1997.

[130] L Rabiner and B-H Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1993.

[131] LR Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77(2):257–285, 1989.

[132] LR Rabiner and RW Schafer. *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

[133] PK Rajasekaran and G Doddington. Robust speech recognition: Initial results and progress. In *Proceedings of DARPA Speech Recognition Workshop*, 1986.

[134] K Reinhard and M Niranjan. Diphone multi-trajectory subspace models. In *Proceedings of ICASSP*, pages 1001–1004, 1999.

[135] HB Richards, JS Bridle, MJ Hunt, and JS Mason. Vocal tract shape trajectory estimation using mlp analysis-by-synthesis. In *Proceedings of ICASSP*, pages 1287–1290, 1997.

[136] F Richardson, M Ostendorf, and JR Rohlicek. Lattice-based search strategies for large vocabulary recognition. In *Proceedings of ICASSP*, pages 576–579, 1995.

[137] M Richardson, J Bilmes, and C Diorio. Hidden-articulator markov models for speech recognition. In *Proceedings of IEEE Workshop ASRU*, 1999.

[138] M Richardson, J Bilmes, and C Diorio. Hidden-articulator markov models: Performance improvements and robustness to noise. In *Proceedings of ICSLP*, pages III:131–134, 2000.

[139] M Riley, W Byrne, M Finke, S Khudanpur, A Ljolje, J McDonough, H Nock, M Saraclar, C Wooters, and G Zavaliagkos. Stochastic pronunciation modelling from hand-labelled phonetic corpora. *Speech Communication*, 29:209–224, 1999.

[140] MD Riley. Statistically-derived pronunciation networks. Technical report, AT&T Bell Labs, 1990.

[141] Michael D Riley. A statistical model for generating pronunciation networks. In *Proceedings of ICASSP*, pages 737–740, 1991.

[142] Michael D Riley and Andrej Ljolje. Automatic generation of detailed pronunciation lexicons. In Chin-Hui Lee, Frank K Soong, and K Paliwal, Kuldip, editors, *Automatic Speech and Speaker Recognition*, chapter 1, pages 1–17. Kluwer Academic Press, 1996.

[143] I Roca and W Johnson. *A Course In Phonology*. Blackwell, 1999.

[144] RC Rose, J Schroeter, and MM Sondhi. The potential role of speech production models in automatic speech recognition. *Journal of the Acoustical Society of America*, 99(3):1699–1709, 1996.

[145] R Rosenfeld. *Adaptive Statistical Language Modeling: A Maximum-Entropy Approach*. PhD thesis, Carnegie Mellon University, PA, USA, 1994.

[146] S Ross. *A First Course In Probability*. Prentice Hall, 1997.

[147] M Russell. Progress towards speech models that model speech. In *Proceedings of IEEE Workshop ASRU*, 1997.

[148] M Saraclar. *Pronunciation Modeling For Conversational Speech Recognition*. PhD thesis, The Johns Hopkins University, MD, USA, 2000.

[149] M Saraclar, H Nock, and S Khudanpur. Pronunciation modeling by sharing gaussian densities across phonetic models. *Computer Speech And Language*, 14(2):137–160, 2000.

[150] LK Saul and MI Jordan. Mixed memory markov models. *Machine Learning*, 37:75–87, 1999.

[151] LK Saul, MG Rahim, and JB Allen. A statistical model for robust integration of narrowband cues in speech. *Computer Speech And Language*, 15(2):175–194, 2001.

[152] R Schwartz, L Nguyen, and J Makhoul. Multiple-pass search strategies. In C-H Lee, FK Soong, and KK Paliwal, editors, *Automatic Speech and Speaker Recognition*, pages 429–456. Kluwer Academic Press, 1996.

[153] EE Shriberg. *Preliminaries to a Theory of Speech Disfluencies*. PhD thesis, University of California at Berkeley, 1994.

[154] T Sloboda and A Waibel. Dictionary learning for spontaneous speech recognition. In *Proceedings of ICSLP*, pages 2328–2331, 1996.

[155] P Smyth, D Heckerman, and M Jordan. Probabilistic independence networks for hidden markov probability models. Technical Report 1565, MIT, 1996.

[156] T Stephenson. Speech recognition using phonetically-featured syllables. Master's thesis, Centre For Cognitive Science, University of Edinburgh, 1998.

[157] T Stephenson, H Bourlard, S Bengio, and AC Morris. Automatic speech recognition using dynamic bayesian networks with both acoustic and articulatory variables. In *Proceedings of ICSLP*, pages II:951–954, 2000.

[158] KN Stevens. From acoustic cues to segments, features and words. Plenary Lecture, Proceedings of ICSLP, 2000.

[159] SS Stevens, J Volkmann, and EB Newmann. A scale for the measurement of the psychological magnitude of pitch. *Journal of the Acoustical Society of America*, 8(185), 1937.

[160] H Strik and C Cucchiarini. Modeling pronunciation variation for asr: A survey of the literature. *Speech Communication*, 29:225–246, 1999.

[161] G Tajchman, E Fosler, and D Jurafsky. Building multiple pronunciation models for novel words using exploratory computational phonology. In *Proceedings of Eurospeech*, pages 2247–2250, 1995.

[162] Howard M Taylor and Samuel Karlin. *An Introduction to Stochastic Modeling*. Harcourt/Acadmic Press, 1998.

[163] MJ Tomlinson, MJ Russell, RK Moore, AP Buckland, and MA Fawley. Modelling asynchrony in speech using elementary single-signal decomposition. In *Proceedings of ICASSP*, pages 1247–1250, 1997.

[164] C Tuerk and T Robinson. A new frequency shift function for reducing inter-speaker variance. In *Proceedings of Eurospeech*, pages 351–354, 1993.

[165] LF Uebel and PC Woodland. An investigation into vocal tract length normalisation. In *Proceedings of Eurospeech*, pages 2527–2530, 1999.

[166] AP Varga and RK Moore. Hidden markov model decomposition of speech and noise. In *Proceedings of ICASSP*, pages 845–848, 1990.

[167] B Vaxelaire, R Sock, and P Perrier. Gestural overlap, place of articulation and speech rate: An x-ray investigation. In *Proceedings of ICSLP*, pages II:166–169, 2000.

[168] AJ Viterbi. Error bounds for convolutional codes and an asymmetrically optimum decoding algorithm. *IEEE Transactions Information Theory*, IT-13:260–267, 1967.

[169] E Wade, E Shriberg, and P Price. User behaviours affecting speech recognition. In *Proceedings of ICSLP*, pages 995–998, 1992.

[170] D Wegmann, S et McAllaster, J Orloff, and B Peskin. Speaker normalization on conversational telephone speech. In *Proceedings of ICASSP*, pages 339–341, 1996.

[171] M Weintraub, A Stolcke, and A Sankar. Sri switchboard progress and experiments. In *Proceedings of DARPA LVCSR Workshop*, 1995.

[172] M Weintraub, K Taussig, K Hunicke-Smith, and A Snodgrass. Effect of speaking style on lvcsr performance. In *Proceedings of ICSLP*, pages S16–S19 (addendum), 1996.

[173] Mitch Weintraub, Steven Wegmann, Yu-Hung Kao, Sanjeev Khudanpur, Charles Galles, Eric Fosler, and Murat Saraclar. Automatic learning of word pronunciation from data. Technical report, The Johns Hopkins University (Center for Language and Speech Processing) Summer Research Workshop, 1996.

[174] J Whittaker. *Graphical Models In Applied Multivariate Statistics*. John Wiley and Sons, 1990.

[175] M Margaret Withgott and Francine R Chen. *Computational Models of American Speech*. CLSI, 1993.

[176] PC Woodland and D Povey. Large scale discriminative training for speech recognition. In *Proceedings of ASR*, 2000.

[177] PC Woodland, D Pye, and MJF Gales. Iterative unsupervised adaptation using maximum likelihood linear regression. In *Proceedings of ICSLP*, pages 1133–1136, 1996.

[178] C Wooters and A Stolcke. Multiple-pronunciation lexical modeling in a speaker independent speech understanding system. In *Proceedings of ICSLP*, pages 1363–1366, 1994.

[179] J Wu and S Khudanpur. Combining nonlocal, syntactic and n-gram dependencies in language modeling. In *Proceedings of Eurospeech*, pages 2179–2182, 1999.

[180] S Young, J Jansen, J Odell, D Ollason, and P Woodland. *The HTK Book (Version 2.0)*. ECRL, 1995.

[181] SJ Young, NH Russell, and JHS Thornton. Token passing: A simple conceptual model for connected speech recognition. Technical Report CUED/F-INFENG/TR38, CUED, 1989.

[182] P Zhan, M Westphal, M Finke, and A Waibel. Speaker normalization and speaker adaptation: A combination for conversational speech recognition. In *Proceedings of Eurospeech*, pages 2087–2090, 1997.

[183] G Zweig and M Padmanabhan. Exact alpha-beta computation in logarithmic space with application to map word graph construction. In *Proceedings of ICSLP*, pages II:855–858, 2000.

[184] GG Zweig. *Speech Recognition With Dynamic Bayesian Networks*. PhD thesis, UC Berkeley, CA, USA, 1998.

[185] A Zwicky. On casual speech. In P Peranteau, J Levi, and G Phares, editors, *Papers from the Eighth Regional Meeting of the Chicago Linguistics Society*, pages 607–615. 1972.