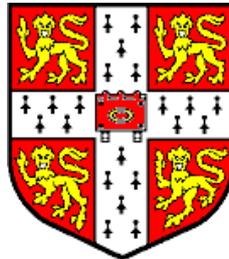


---

# Hidden Model Sequence Models for Automatic Speech Recognition

**Thomas Hain**

Darwin College  
University of Cambridge



November 2001

Dissertation submitted to the University of Cambridge  
for the degree of Doctor of Philosophy

---

# Abstract

Most modern automatic speech recognition systems make use of acoustic models based on hidden Markov models. To obtain reasonable recognition performance within a large vocabulary framework, the acoustic models usually include a pronunciation model, together with complex parameter tying schemes. In many cases the pronunciation model operates on a phoneme level and is derived independently of the underlying models. In contrast, this work is aimed at improving pronunciation modelling on a sub-phone level in a combined framework. The modelling of pronunciation variation is assumed to be of special importance for recognition of spontaneous speech.

The novel framework of hidden model sequence modelling using the basis of hidden Markov models is presented. The framework allows arbitrary mappings between model and phoneme sequences. A stochastic model for this mapping, the model sequence model, is introduced. The parameters of this new model can be estimated jointly with the parameters of the underlying hidden Markov model set by the use of a maximum likelihood criterion. The set of potential mappings is divided into the cases of fixed and variable alignment between the model and phoneme sequences. Whereas the former only allows the modelling of substitutions, the latter can be used to describe additional sub-phone insertion and deletion effects.

A range of different approaches for potential model sequence models is proposed. The natural form of modelling the fixed alignment case is an N-gram type model using a constrained phoneme context. Issues such as model structure, data sparsity and appropriate initialisation of the model parameters are discussed in detail and a set of solutions is tested. For the case of variable alignment a multigram based model sequence model is presented and two different implementations are investigated. Special attention is given to the modelling of sub-phone deletions.

Experimental evidence is presented on the basis of results obtained on two transcription tasks: Resource Management as an example for read speech; and Switchboard as an example for a complex spontaneous speech task. On both tasks statistically significant improvements are obtained over a standard HMM baseline with a relative reduction in word error rate of more than 25% on Resource Management and 5% on Switchboard.

## Keywords

Speech recognition, pronunciation modelling, hidden model sequences, model sequence models, context, sub-phone variation, insertion and deletion effects, spontaneous speech, hierarchical systems, hidden Markov models.

# Declaration

This dissertation is the result of my own work carried out at the Cambridge University Engineering Department; it includes nothing which is the outcome of work done in collaboration. Reference to the work of others is specifically indicated in the text where appropriate. Some of the material has been presented at international conferences and workshops over recent years: (Hain and Woodland, 1999b), (Hain and Woodland, 1999a) and (Hain and Woodland, 2000).

The length of this thesis including footnotes and appendices is approximately 42000 words.

---

# Acknowledgements

Firstly, I would like to thank my supervisor Phil Woodland for his help and support throughout the work on this thesis. His experienced opinion, well targeted questions, and careful analysis often helped me to focus on the relevant matters. I owe much of my knowledge about building speech recognition systems to Phil and the days and nights we spent in preparation for Hub4 or Hub5 evaluations.

Special thanks go to Steve Young for his support and the great privilege to work in a group under his guidance.

Many people have helped me during the course of my studies in a variety of ways. I would like to thank Mark Gales for the many long, fruitful discussions, Thomas Niesler for his support and friendship and Gunnar Evermann for helping whenever he could and his dedication to HTK. I am grateful for the presence of Dan Povey, Sue Johnson, Andreas Tuerk, Harriet Nock, Patrick Gosling, Gary Cook, Ed Whittaker, and Matt Stuttle and all the others in the labs, and thank them for their help and for providing a good working atmosphere.

I am indebted to Steve Young, Phil Woodland, Mark Gales, Gunnar Evermann, and all other contributors to CU-HTK for providing and maintaining this great research toolkit and other related software available at the Speech, Vision and Robotics group.

The biggest thank you must go to my wife Doris for moving to Cambridge, for her unfailing love and support and for enduring many lost evenings, weekends, and holidays. I am grateful to my daughter Helene for seconding her favourite soft toys to assist in “writing Papa’s book” although even these eager helpers could not ensure the regular visit to the playground. Final thanks go to my parents for their support and encouragement through all these years.

# Notation

## Commonly used symbols and notation

$\mathbf{x} = (x_1, x_2, \dots, x_d)^T$	an $d$ -dimensional column vector
$\mathbf{A}$	a matrix with arbitrary dimension
$\mathbf{A}^T$	the transpose of the matrix $\mathbf{A}$
$ \mathbf{A} $	the determinant of the square matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	the inverse of the square matrix $\mathbf{A}$
$\mathbf{X}$	an arbitrary length sequence of scalar or vector-valued elements
$x_i$	the $i^{\text{th}}$ scalar element of a sequence of scalars $\mathbf{X}$
$\mathbf{x}_i$	the $i^{\text{th}}$ vector-valued element of the vector-valued sequence $\mathbf{X}$
$\mathbf{X}_i^j = [x_i, x_{i+1}, \dots, x_j]$	a particular subsequence of the sequence $\mathbf{X}$
$L_X$	the length of the sequence $\mathbf{X}$
$p(x)$	the probability density function for a continuous random variable $x$
$P(x)$	the probability of a discrete event $x$ , the probability mass function
$\mathcal{E}\{x\}$	the expected value of $x$
$Q(\hat{\theta}, \theta)$	the auxiliary function for reestimated and original parameters $\hat{\theta}$ and $\theta$ respectively
$N(e)$	the frequency of an event $e$
$N(\cdot)$	the frequency of all possible events
$\mathcal{S}$	a set of symbols
$M_S$	the number of elements in set $\mathcal{S}$

---

---

## *Contents*

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Speech recognition systems	2
1.2	Spontaneous speech	4
1.3	Motivation and objective	4
1.4	Thesis structure	6
<b>2</b>	<b>Speech recognition fundamentals</b>	<b>8</b>
2.1	Statistical speech recognition	8
2.2	Analysis and transformation of speech signals	10
2.2.1	Speech recognition front-ends	11
2.3	Acoustic modelling using hidden Markov models	14
2.3.1	Basics	14
2.3.2	Probability computation	17
2.3.3	Output distributions	18
2.3.4	Parameter estimation	20
2.3.5	Sub-word units	21
2.4	Continuous speech recognition	22
2.5	Training of model sets	22
2.6	Language modelling	24
2.6.1	N-gram language models	24
2.6.2	Scaling and insertion penalties	26
2.7	Decoding	26

---

<b>3 Hierarchies and sharing in automatic speech recognition</b>	<b>28</b>
3.1 From sentence to signal	28
3.2 Word level	31
3.3 Pronunciation level	32
3.4 Model level	34
3.5 State and mixture component level	34
3.6 Combination of levels	36
3.7 Summary	36
<b>4 Hidden model sequences</b>	<b>38</b>
4.1 A hierarchical formulation of speech recognition	38
4.1.1 Maximum likelihood	41
4.1.2 The Viterbi approximation	43
4.1.2.1 Motivation and analysis	44
4.1.3 An information theoretic perspective	45
4.2 Model sequence modelling	49
4.2.1 Fixed alignment	50
4.2.2 Variable alignment	54
4.2.2.1 Multigrams	55
4.2.2.2 Multigram based model sequence models	56
4.3 Decoding	61
4.4 Summary	62
<b>5 Implementation of HMS-HMMs</b>	<b>64</b>
5.1 Implementation	65
5.1.1 Data sparsity	65
5.1.2 Discounting	66
5.1.2.1 Good-Turing discounting	67
5.1.2.2 Witten Bell discounting	68
5.1.2.3 Absolute discounting	68
5.1.3 Backing off	68
5.1.4 Interpolation	69

---

5.1.5	Pruning and perplexity	70
5.1.6	Model sets	71
5.1.7	Scaling and normalisation	72
5.2	Baseline systems	72
5.2.1	Resource Management	73
5.2.2	Switchboard	73
5.3	Phone models	75
5.4	Phoneme position dependent models	76
5.4.1	Smoothing	81
5.4.1.1	Backoff to simple interpolated distributions	82
5.4.1.2	Backoff to interpolation by deleted estimation	83
5.4.1.3	Discounting methods	84
5.4.2	MSM Initialisation	86
5.4.3	Clustering of distributions	87
5.4.4	HMM initialisation	89
5.4.5	Soft-tying of states	92
5.4.6	Pronunciation modelling	94
5.4.7	Model insertions and deletions	97
5.4.7.1	Scenario A: Phone models	98
5.4.7.2	Scenario B: Position dependent modelling	99
5.4.8	HMS-HMM in combination with standard ASR techniques	100
5.5	Summary	101
<b>6</b>	<b>Summary and conclusions</b>	<b>103</b>
6.1	Review of the Work	103
6.2	Suggestions for future work	105
6.3	Conclusion	106
<b>A</b>	<b>Speech recognition task descriptions</b>	<b>107</b>
A.1	Resource Management	107
A.2	Switchboard	108

<b>B</b>	<b>Single Pronunciation Dictionaries</b>	<b>111</b>
B.1	Construction of single pronunciation dictionaries	111
B.2	Experiments	112
<b>C</b>	<b>Interpolation of model distributions</b>	<b>115</b>
	<b>Bibliography</b>	<b>118</b>

---

## *Introduction*

---

Automatic speech recognition (ASR) by machines is an interesting research topic that attracts attention from researchers all over the world. In the past decade the technology has advanced in a way which allowed commercial and military exploitation of ASR on a large scale. Since the performance of speech recognition algorithms is still far from the capability of humans, many methods and complete frameworks have been developed to circumvent the shortcomings of today's speech recognisers and a considerable effort has been made to enable their use in relatively complicated environments. The rapid development of computers in terms of speed and storage capability has certainly facilitated some of the advances made in the past decades. Whereas in early years work in speech processing generally was considerably constrained by the computational resources available, these constraints appear to be less restrictive today. Other obstacles though have emerged or their importance was increased which place limitations on the way modern speech recognition systems are developed.

The task to classify speech patterns by means of machines attracted researchers in the early days of computing. The most important step forward was made in the 1970s when the use of *hidden Markov models* (HMMs) was introduced into speech recognition by researchers at Carnegie-Mellon University (Baker, 1975) and at the IBM Research Labs (Jelinek et al., 1975). In subsequent years hidden Markov models became the major technique used for automatic speech recognition as well as many other speech classification tasks such as speaker identification and verification, language identification and speech segmentation. Even though the general paradigm is quite flexible, speech recognition systems today are far from generic in the sense that reasonable performance can be obtained under a variety of conditions. The modelling of speech signals using HMMs involves the use of independence assumptions which have been the subject of criticism (Levinson, 1994; Russell, 1997; Bourlard, 1995) and led to the use of alternative models. The most important alternative approach used artificial neural networks (ANNs) to form hybrid HMM-ANN models (Bourlard and Morgan, 1994; Robinson, 1994). However in general, ASR systems based on HMMs show the best performance. For a more detailed description of the history of speech recognition the interested reader is referred to (Gold and Morgan, 1999) and in a wider context to (Jurafsky and Martin, 2000).

Applications of ASR have moved from operation in well controlled environments to more complex tasks where the speech signal is not specifically targeted at a computer, as for example to the transcription of human-human conversations over the telephone or meeting transcription. Increased ambiguity in the speech patterns is observed in these cases. If the complexity of a particular application in terms of the speech content is large, substantially poorer performance than in controlled situations is observed. Part of the difficulty is assumed to originate from an increased variability in the pronunciation of words or short phrases. This thesis investigates specifically the modelling of pronunciation variation.

Work in this thesis has been based on the Cambridge University Hidden Markov Model Toolkit (CU-HTK) which in itself is a modified version of the publicly available HTK Toolkit (Young et al., 1999). Considerable extensions were necessary to implement the ideas and algorithms under investigation. Beyond the standard HTK toolkit other tools were necessary to provide the necessary infrastructure. This includes tools for generation and processing of word graphs and language model training.

## 1.1 Speech recognition systems

The development of an ASR system is complex and requires careful analysis, design and implementation of its individual parts. The basic structure of a speech recognition system is depicted in Figure 1.1. In the first stage (the *front-end*) the signal is compressed into a feature stream which is deemed to hold sufficient information for the task at hand. The actual recognition stage makes use of an *acoustic model* and a *language model* to arrive at a hypothesis for the spoken sentence. If the specific application allows *adaptation*, the output can be used to adjust either one or both models or even the signal processing stage<sup>1</sup>. The output of the second recognition pass using adapted models can show considerably lower error rates. Both acoustic and language models may consist of multiple parts. The *vocabulary* and the language model represent the syntactic and semantic properties of the speech to be recognised whereas the acoustic model is responsible for mapping of the feature stream to individual words. In large vocabulary speech recognition the acoustic models make use of a *pronunciation dictionary* which translates words into smaller units reflecting the pronunciation of each word in the vocabulary. However, the pronunciation information may also be represented by a model that translates words or word sequences into pronunciation networks.

The structure of the core recognition system outlined in Figure 1.1 is mostly independent of the particular application. Nevertheless the construction of a speech recognition system has to take multiple factors into account and the final configuration is often specific to a particular task. Since most techniques are known to interact in non-obvious ways, one particular set of factors describing a particular task, might require substantially different strategies than in standard cases. In essence the design criteria can be divided into those characterising the type of data

---

<sup>1</sup>In this case the boundary between acoustic modelling and front-end signal processing is less clear.

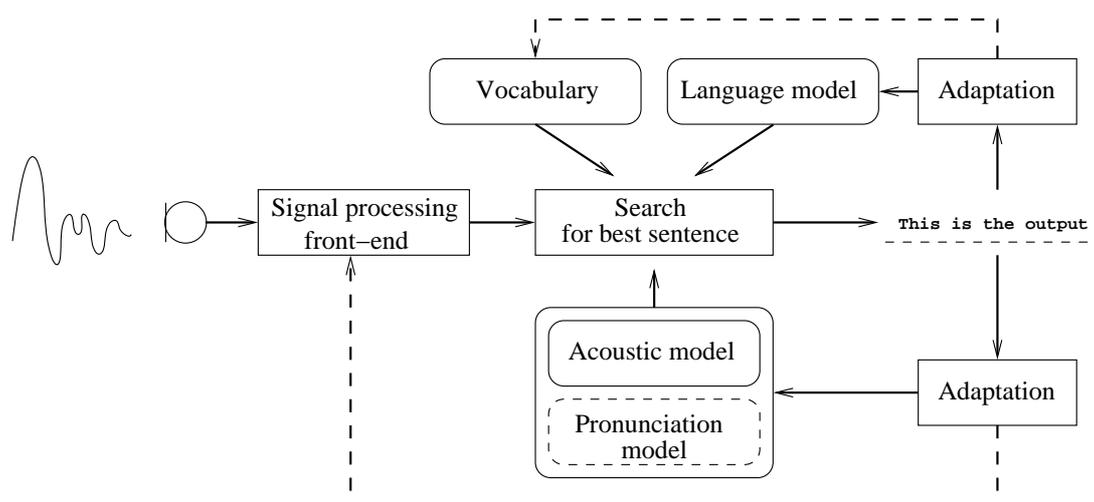


Figure 1.1 *Information processing and knowledge sources in an automatic speech recognition system.*

to be processed, those specifying the interface to the application and finally those specific to the operational platform. Whereas the latter are not of interest in this thesis, they usually have considerable impact on system design. The following list of properties characterises the data to be processed:

- Speech flow:  
Continuous or discrete speech
- Speaker dependence, accent dependence, multi-lingual:  
Built for a known speaker, a known region, for multiple languages
- Speech type:  
Read, spontaneous, or conversational speech.
- Simultaneous speech:  
Overlap of multiple speakers, speaker tracking
- Transmission channels:  
Broadband high quality, telephone or severely distorted speech
- Environmental conditions  
Various levels of noise
- Multi-modal interfaces  
Input from video speaker tracking, lip-reading.

Ideally each set of specifications only has implications for only one part of the core components of Figure 1.1. Some factors only have, for example, implications for the design of the pre-processing stage (e.g. channel bandwidth) whereas others either require interdependent changes

in multiple parts of the system (e.g. noise conditions) or even fundamentally change the system structure and methodology.

Modern speech recognisers are usually designed to process continuous speech from arbitrary unknown speakers. Focus on transmission and noise related problems has been fueled by the increased interest in speech recognition over telephone channels or similarly constrained and distorted conditions as for example can be found in cars. Application specific design in general relates to the size, style and type of the vocabulary as well as syntactic and semantic constraints.

## 1.2 Spontaneous speech

An important design factor is the type of speech which has an influence on all parts of an ASR system. Initial work on speech recognition tried to focus on simple tasks. In terms of speech type, this means well articulated read speech preferably spoken by a trained speaker. The first attempts even required speakers to artificially separate words by short but noticeable pauses during dictation. This requirement was dropped and many new techniques for the transcription of continuous speech were developed, especially using the *Wall Street Journal* (WSJ) task (Paul and Baker, 1992). This task was sufficiently demanding that, among other things, important lessons on how to define suitable acoustic modelling units were learned.

Originally a wider investigation into spontaneous or unplanned speech started with work on the *Air Travel Information System* (ATIS) task (Hemphill et al., 1990). The ATIS corpus contains read and spontaneous versions of particular utterances, which stimulated a range of investigations into the nature of spontaneous speech. However, the nature of this task ensures that a relatively small vocabulary of 2000 words is sufficient and thus limits confusability. More recently work on the much more demanding transcription of Broadcast News (BN) data has been stimulated by the U.S. DARPA programmes (Young and Chase, 1998). The data in the BN corpus contains, among other so called “focus conditions”, a significant portion of spontaneous speech. As is evident from the results obtained by various research groups, error rates achievable on this portion of the data are significantly higher than on read speech in a comparable environment. Work specifically targeted at spontaneous speech on a large scale was started with benchmark tests on the Switchboard and CallHome corpora (see Appendix A.2). Word error rates on this type of data are considerably higher than those observed for the spontaneous speech portions of BN data. Apart from difficulties in language modelling, a major problem appears to be pronunciation variability which has been subject of many investigations (e.g. (Weintraub et al., 1996a; Byrne et al., 1998; Saraçlar et al., 2000; Riley et al., 1999)).

## 1.3 Motivation and objective

Work on clean read speech clearly established the concept of pronunciation in automatic speech recognition, exploiting the fact that “the functional units of which sounds are realisations are

phonemes” (O’Connor, 1973). The properties of read speech allowed the proper design of a pronunciation dictionary by a human expert (e.g. (Gauvain et al., 1994)) using multiple pronunciation variants for a particular word if necessary<sup>2</sup>. Within a word a particular phoneme together with its neighbours specifies the HMM responsible for modelling the acoustic realisations of that word. Since the selection of an HMM varies depending on the phoneme context, models are intended to represent *phones* and the set of HMMs associated with a certain phoneme are meant to model allophonic variation. In construction of a dictionary a phonemic representation of words is usually preferred to a more detailed phonetic transcription. Phonetic transcription is the classification of speech data performed by humans allowing a closer description of speech sounds. Even though a detailed sound description seems desirable, much of this information is potentially unnecessary for classification of words or may even be detrimental to performance at the word level. This makes the choice of symbols and an optimal symbolic transcription of words in a dictionary difficult, even more so in the case of spontaneous speech, where the discrepancy between dictionary baseforms and phonetic transcription of utterances is considerable (Weintraub et al., 1996a; Greenberg, 1996).

Spontaneous speech shows a multitude of special properties concerning all components of a speech recognition system. These properties are not well understood (Jelinek, 2000) and despite considerable effort state-of-the-art systems scarcely model these properties explicitly. An interesting experiment conducted by (Weintraub et al., 1996b) and repeated by (Saraçlar et al., 2000) shows that there exists a considerable difference in performance between real spontaneous speech and imitated or read versions of the same sentences. Apart from the obvious additional hesitations, false starts and confirmation sounds<sup>3</sup> the speech patterns as a whole are considerably more varied (Saraçlar et al., 2000; Greenberg, 1998). A greater variety in terms of speaking rate and intonation patterns goes along with an obviously increased number of phonetic deletion and insertion effects. Humans can filter out the “distortions”, but still are able to distinguish between read and spontaneous speech, even if the actual spoken text is identical. Experiments have shown that human listeners themselves are often inconsistent in transcribing the speech to a significant extent. An increase in the number of pronunciation variants alone potentially increases confusability. Faced with the difficulty of defining proper dictionary entries for spontaneous speech, it is desirable to develop methods which are data driven and thus can adjust automatically to the specific properties of the underlying data (Riley et al., 1999), preferably without manually phone labelled data. Even though some performance improvement is achieved by pronunciation modelling on the level of dictionary symbols, the results fall far short from those envisaged in (Weintraub et al., 1996b). Usually, a unique mapping from phonemes to phone HMMs is provided in the form of phonetic decision trees which are constructed using data driven techniques. One way to describe the modelling of pronunciations at the symbol level is an indirect selection of HMMs through the filter of a phonetic decision tree. Rather than keeping the pronunciations or the decision tree fixed, a joint optimisation may provide better

---

<sup>2</sup>An obvious set of multiple pronunciations is for example required for the words “the” or “read”. Less obvious extensions include for example aspirated versions of words starting with “w” like “where”.

<sup>3</sup>These are sounds like “uh hum” for yes and “uh uh” for no.

modelling. Secondly it is difficult to imagine insertion, deletion and even substitution effects on a phone scale. Modelling sub-phone changes always holds the potential to alter the complete phone as well as change of individual parts and thus is a more powerful modelling technique.

In this thesis a novel method for HMM model topology generation is presented to be used in automatic speech recognition. This method is developed with special focus on the use in the transcription of conversational speech. In this new framework the one to one deterministic mapping between a phoneme and a phone model is replaced by a stochastic mapping. Since this framework assumes that the allophone model sequence of the training data is unknown the term *model sequence models* (MSMs) will be used to characterise models for this stochastic mapping. This additional model adds another layer to the hierarchy of speech recognition systems. It will be shown that a proper training scheme for this new model can be developed using the maximum likelihood criterion in the mathematical framework of Expectation-Maximisation. The framework of hidden model sequences does not dictate a particular implementation. A range of potential realisations is presented, implemented and evaluated using data from read and spontaneous speech corpora. Most importantly a clear distinction is made between models that allow for substitutions only and models that incorporate insertions and deletions on a sub-phone level. As will be described in detail in later chapters this results in the data driven construction of complex word and sentence model topologies. Since the complexity of the acoustic models is increased significantly the methods under investigation have to address the problem of data sparsity. Furthermore in the light of the new framework the necessity of multiple pronunciations in the recognition and training dictionaries is re-investigated.

## 1.4 Thesis structure

This thesis is structured as follows: The following chapter gives a comprehensive overview of state of the art in HMM-based speech recognition. A brief description of the stochastic paradigm used for ASR is followed by a more detailed description of training and decoding with HMMs. Techniques that are fundamental for the implementation of a large vocabulary speech recogniser are described.

Chapter 3 presents a hierarchical view of speech recognition with special focus on schemes aimed at improving HMM selection and/or HMM topology in wider sense. This involves specific modelling at word, pronunciation, HMM and HMM state level. Methods which have an influence on multiple levels are described.

In Chapter 4 the framework of hidden model sequences is presented and the associated terminology is described. A training scheme based on maximum likelihood is developed together with a discussion on the use of the Viterbi approximation. A distinction between model sequence models based on fixed or variable alignment is made and modelling approaches for each of these cases are presented. An N-gram based solution is presented for the case of fixed alignment whereas the natural extension in the variable alignment case was found to be based on multi-

grams. Re-estimation formulae are described and implementation issues are discussed. Due to the increase in complexity the relevant aspects for the decoding process are described.

Chapter 5 is devoted to implementation issues, such as the choice of model structures and the treatment of data sparsity related problems in training of MSMs, are discussed in detail and specific solutions are described. After a description of the baseline systems the remaining sections are devoted to experiments on the use of phone level and sub-phone level hidden model sequence HMMs. The proper choice of structure, initialisation, and training of fixed and variable alignment MSMs is addressed in experiments on the Resource Management and Switchboard corpora. Details about these corpora can be found in Appendix A.

The final chapter presents a summary of the work in this thesis and discusses potential future directions of research, followed by conclusions.

---

*Speech recognition fundamentals*

---

The purpose of this chapter is to introduce the main components of a state-of-the-art continuous speech recognition system and to describe the interaction between them. Modern speech recognition systems are becoming increasingly complex. Nevertheless the core algorithms used in most systems are usually very similar or identical across a wide range of tasks. Essential technologies and those important for an understanding of the remainder of this thesis are described and explained in detail. In particular acoustic modelling using hidden Markov models including training and topology issues, and basic pronunciation and language modelling methods are discussed. Even though in practice speech recognition systems usually have a range of minor differences which have an impact on the performance, the focus here is to retain a unified view as much as possible. In this work HTK was used for experiments and thus serves as a reference implementation. HTK provides a state-of-the-art framework for development of automatic speech recognition algorithms and allows great flexibility for the implementation of new algorithms. Implementational details given in this chapter are based on HTK and the corresponding tools are used for experiments in this thesis. The interested reader is referred to background literature for more details on topics in speech processing or speech recognition (e.g. (Rabiner and Schafer, 1978; Rabiner, 1993; Deller et al., 1993; Bourlard and Morgan, 1994; Jelinek, 1997; de Mori, 1998; Gold and Morgan, 1999; Jurafsky and Martin, 2000; Huang et al., 2001)).

## 2.1 Statistical speech recognition

The task of transcribing continuous speech utterances is simple to define: Given a certain acoustic representation of an utterance  $\mathbf{A}$  find the associated textual representation, which in continuous speech recognition is a sequence of words  $\mathbf{W} = (w_1, w_2, \dots, w_N)$ . Using a stochastic formulation the task is to find the sentence or word sequence  $\hat{\mathbf{W}}$  which is most likely to have produced  $\mathbf{A}$ , i.e.

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmax}} P(\mathbf{W}|\mathbf{A}) \quad (2.1)$$

This describes the *decoding* of an utterance into a word sequence in probabilistic terms. The estimation of the above probability term for continuous speech recognition is intractable. Application of Bayes rule brings

$$\begin{aligned}\hat{\mathbf{W}} &= \operatorname{argmax}_{\mathbf{W}} \frac{P(\mathbf{A}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{A})} \\ &= \operatorname{argmax}_{\mathbf{W}} P(\mathbf{A}|\mathbf{W})P(\mathbf{W})\end{aligned}\quad (2.2)$$

Since the above equation only requires the maximisation over all word sequences the probability of the acoustics  $P(\mathbf{A})$  is irrelevant and thus is omitted. Any statistically based ASR systems must be able to model two complex probability distributions: the probability of some acoustics given a particular word sequence  $P(\mathbf{A}|\mathbf{W})$  or *acoustic model* and the prior probability of an arbitrary word sequence with arbitrary length  $P(\mathbf{W})$  or *language model*.

Even though the above equations appear to be simple the estimation of the probability distribution in the case of continuous speech recognition still remains difficult. In the special case of isolated word recognition the complexity of the task is significantly lower. The word sequence  $\mathbf{W}$  is reduced to a single word  $w$  and the prior distribution is a simple distribution over a finite sets of words which allows the direct implementation of Equation 2.1<sup>1</sup>. However, if such a procedure would be adopted for continuous speech recognition many samples of each sentence that can possibly be spoken would be required. This clearly is infeasible in practice. One solution is to split the task into a combined classification and segmentation problem. This has important implications: the procedure itself must be able to decide on the location of the time boundaries of the modelled units; the procedure must be able to cope with arbitrary length sequences of words and acoustic symbols; and the selection of a word as recognition unit is arbitrary and other units for which a unique mapping to words exists may better serve the purpose.

Considering the large number of potential words in a single language, including all inflected and derived forms, it is evident that still not enough acoustic observations of each word might be available. Smaller phonological recognition units like *phonemes* or *syllables* are used to overcome this problem. Recognition units which are shorter than complete words are called *sub-word units*. By sub-word unit modelling another layer is added to the combined segmentation and classification problem, namely the mapping between actual recognition units and words. In the case of phonetic units this layer is called the *pronunciation model*. The mapping from pronunciations to words usually is not unique and most commonly assumed to be constant regardless of the task<sup>2</sup>(as for example in the systems described in (Woodland et al., 1994; Woodland et al., 1997a; Hain et al., 1999)). Methods which modify this approach are discussed in Chapter 3.

In the case of isolated word recognition it is clear that an actual list of words which can be identified by the presented recognition algorithm must exist. This list is also called the *vocabulary* of

<sup>1</sup>A “word” in this case may be any meaningful utterance.

<sup>2</sup>In practice different tasks might have different vocabularies and thus may implicitly use a different set of pronunciations. Nevertheless the core vocabularies most likely overlap and remain constant.

a speech recogniser. In the case of continuous speech the concept of a constrained vocabulary is maintained. An advantage of this scheme is that speech recognisers do not produce spelling errors, but words spoken might not be contained in the recognition vocabulary and thus will definitely cause an error. In practice those words are called *Out-Of-Vocabulary* (OOV) words and the coverage of the recognition vocabulary of a particular task is measured in terms of the OOV-rate<sup>3</sup>.

This section outlined the basic speech recognition paradigm and presented some of the implications for continuous speech recognition. Note that so far no particular modelling technique for any of the knowledge sources has been assumed. Nevertheless the following sections assume that hidden Markov models are used for acoustic modelling. As mentioned in Chapter 1 other approaches exist, but do not represent the mainstream of speech recognition systems.

## 2.2 Analysis and transformation of speech signals

The acoustic speech signal, regardless of the textual and other information it is carrying for human communication purposes, is constrained in shape and structure by its production mechanism. The most comprehensive description of a speech signal would be by description of the way in which the signal was generated. The quality of a model of speech production can be determined by comparing the output of the model with real speech signals. In many cases a simple parametric description of the essential behaviour is desired and robust estimation of the parameters can only be provided if the number of parameters is small. For the purpose of speech recognition an approach based on speech production only has the disadvantage that the inherent redundancy of speech signals is not fully addressed. The redundant elements of the signal cannot be identified on the signal level alone. A compact description of the speech signal requires incorporation of knowledge about speech perception, which in turn makes the search for an appropriate speech model more difficult. Whereas speech production can, in principle, be understood by the movement of speech production organs, perception not only includes the physical conversion from air pressure waves to electric signals in the human ear, but also the further processing by the human brain. Information about the incoming signal which is irrelevant for speech perception, may still be contained in the information collected by the auditory nerve endings that are attached to the  $\sim 30,000$  sensory hair cells on the organ of Corti (O'Shaughnessy, 1987). Perception of speech is probably the most important task for the human auditory system, but it is not its sole purpose.

The properties outlined in the previous paragraph clearly put a limit to the accuracy of the models representing speech signals. The limitations plus the desire for simple models with few parameters are reflected in the simple *source-filter model* of speech, which is commonly used for speech coding and for model based approaches in speech synthesis and recognition (see for

---

<sup>3</sup>The OOV-rate is defined as the number of unknown tokens (i.e. words not in a specific word list) divided by the total number of tokens in the reference text.

example (Goldberg and Riek, 2000),(Huang et al., 2001)). This model of speech production is based on the assumption that any acoustic stimulus is physically located on one end of a tube-like structure, the *vocal tract*. The stimulus is assumed to either consist of white noise or of a close to harmonic signal, generated by the vibrating vocal chords. The separation into an excitation signal and a frequency response is important for speech recognition. At least for English, the excitation is commonly assumed to be irrelevant for the classification of words. An extraction of the vocal tract response from the observed signal is most commonly achieved by use of a linear prediction filter. The aspect of perception is not included in this type of model.

Speech recognition cannot be performed on the signal directly. It is important to condense the signal to only those parts which are necessary for the speech signal to be intelligible. Ideally a time-discrete representation of the signal is sought which not only removes the recording and environmental effects<sup>4</sup> but which is also speaker independent. On the other hand a representation similar to that entering the information processing stages of the human brain should at least have the potential for similar recognition capability to humans and thus speech recognition front-ends are more concerned with better modelling of perception.

### 2.2.1 Speech recognition front-ends

The first stage of processing of the acoustic signal is the capture by a microphone and subsequent conversion of the analogue signal into a digital representation by sampling at the appropriate frequency. If the speech is not band-limited by for example a telephone channel, the frequency range between 0 and 8 kHz is assumed to contain all necessary information and thus a sampling rate of 16 kHz is sufficient for preserving high quality speech. As bandwidth is restricted speech quality and intelligibility suffer as is obvious in telephone speech, where realistically only the frequency range between 125-3800 Hz can be exploited (Hain and Woodland, 1998). However the performance loss for speech recognition due to such bandwidth limitation is not dramatic (Bernstein et al., 2000). Analogue to digital conversion also quantises the signal in value. Sixteen bits sufficiently cover the dynamic range of speech signals in linear representation.

The spectral representation of a speech signal is much more informative about the speech sounds than the time domain signal and the human ear is found to perform a non-linear frequency analysis (O'Shaughnessy, 1987). Furthermore an inspection of speech spectrograms reveals that in general the energy in higher frequency bands is lower and that the spectrum of a speech signal is quasi stable for a certain duration. The spectral imbalance can be corrected by using a first order pre-emphasis filter (as for example in (Young et al., 1999)). Speech analysis algorithms usually assume that quasi-stationary speech segments exist which can be isolated and treated independently. Model parameters or a Fourier spectrum can be estimated on only that portion of the signal. In order to distort the signal content as little as possible by this operation, the speech signal is multiplied by a windowing function, most commonly the Hamming window

---

<sup>4</sup>E.g. microphone types and variants of analogue to digital conversion.

$$w[n] = \begin{cases} 0.54 - 0.46 \cos(2\pi \frac{n}{N-1}) & \text{if } 0 \leq n \leq N - 1 \\ 0 & \text{otherwise} \end{cases}$$

where  $N$  is the length of the window. The result of this multiplication is a frame of fixed length  $N$  and by stepping along the speech signal at a certain interval and application of the above window function a representation of the complete speech signal is obtained as a sequence of frames<sup>5</sup>. Typically a frame is taken every 10ms and the overall frame size is 25ms and all systems under investigation in this thesis make use of this rate. Non-uniform sampling for example at maximum energy points (Holmes, 2000) and other frame rates are sometimes used either for computational reasons or for building systems similar in performance but different in behaviour (Billa et al., 1998).

In a second stage each frame at time  $t$  in the speech stream is converted independently into a low-dimensional feature vector  $\mathbf{o}_t$  which is suitable for processing in a particular speech recognition system. Most commonly used are either linear prediction (LP) coefficients or their derived compressed and better decorrelated representations (Rabiner, 1993), Mel frequency cepstral coefficients (MFCCs, (Davis and Mermelstein, 1980)) and Perceptual Linear Prediction (PLP) coefficients (Hermansky, 1990). Whereas linear prediction based methods try to emulate the production mechanism the main motivation for MFCC and PLP coefficients is to model perceptual behaviour<sup>6</sup>.

Mel frequency cepstra model three important properties of hearing. The ear appears to be relatively insensitive to phase variations in the sound stimulus as long as group delay variations are small (O'Shaughnessy, 1987). Thus the phase information of the speech spectrum is not taken into consideration. The second property is a non-linear warping of the frequency axis, which is approximated by the Mel warping function (O'Shaughnessy, 1987; Gold and Morgan, 1999):

$$\text{Mel}(f) = 2595 \log_{10}(1 + \frac{f}{700})$$

Thirdly the presence of energy in certain frequency bands affects the perception in neighbouring bands, an effect known as frequency masking. For the computation of Mel frequency cepstral coefficients the magnitude of the discrete Fourier spectrum is computed. A filterbank with triangular filters, linearly spaced in the Mel scale, is applied to obtain a relatively small number of coefficients. An important final stage is the computation of the cepstral coefficients. Using this method the effect of low frequency periodic signals can be suppressed by limiting the number of coefficients retained while making convolutional channel distortions additive<sup>7</sup> (Rabiner, 1993). This fact is exploited for removal of static convolutional channel distortions by cepstral mean

<sup>5</sup>Given that frames overlap and that unlimited numerical resolution is available the speech signal can be fully reconstructed from this representation.

<sup>6</sup>Experiments in this thesis makes use of MFCC and PLP representations.

<sup>7</sup>The real-valued cepstrum is defined as the inverse Fourier transform of the logarithm of the magnitude spectrum. Since a convolution in the time domain corresponds to a multiplication in the frequency domain and the logarithm

subtraction. The  $K$  cepstral parameters  $\{c_k : 1 \leq k \leq N\}$  can be computed by taking the natural logarithm of the Mel-filterbank coefficients  $m_k$  and applying the discrete cosine transform:

$$c_i = \sum_{k=1}^K m_k \cos\left(\frac{\pi i}{K}(k - 0.5)\right)$$

In addition to the cepstral coefficients the signal energy is included in the final feature vector<sup>8</sup>.

PLP coefficients further incorporate the fact that the perceived loudness of tones is not uniformly proportional to the energy of the speech signal over the complete frequency range. Thus an equal loudness curve with a peak at around 3.5kHz is applied to emphasise parts of the spectrum. Furthermore the spectrum is uniformly compressed by using the cubic root of the signal power at each frequency. This should relate the energy of a signal to the actually perceived loudness by humans<sup>9</sup>. In the original description (Hermansky, 1990) the Bark-scale for non-linear warping and special filterbanks are used to model non-linear frequency warping and masking respectively. The PLP implementation used here however retains the Mel filterbank as used for MFCCs (Woodland et al., 1997a). Since the autocorrelation coefficients can be obtained by inverse Fourier transform of the power-spectrum, LP cepstral coefficients can be computed in a straightforward manner<sup>10</sup>.

Hidden Markov modelling of speech assumes that successive feature vectors  $\mathbf{o}_{t-1}$  and  $\mathbf{o}_t$  are not correlated. Since this is obviously a grossly invalid assumption, information about neighbouring frames is introduced by the use of first and second order derivatives. These are commonly termed  $\Delta$  and  $\Delta\Delta$  coefficients<sup>11</sup> and have been shown to improve recognition performance (Lee et al., 1991; Wilpon et al., 1991). Another method which recently attracted increased interest is the use of discriminative dimension reduction schemes which are capable of transforming a higher dimensional vector (such as for example concatenated adjacent feature vectors) into a lower dimensional vector by use of information about the classification model. For example methods based on linear discriminant analysis (LDA) (Duda et al., 2001; Haeb-Umbach and Ney, 1992) are important transformational schemes. Note that the exploitation of interdependence between adjacent vectors is rather associated with generic pattern processing schemes than with the specific properties of speech signals.

---

of two multiplied factors is a sum of the logarithms of those factors, the computation of the cepstrum is a method of blind deconvolution.

<sup>8</sup>In HTK the signal energy is computed as the log of the square sum of the windowed signal:

$$E = \log \sum_{n=1}^N s^2(n)$$

<sup>9</sup>Both methods are approximations since the equal loudness curves also depend on the actual intensity level of the signal.

<sup>10</sup>The 0<sup>th</sup> cepstral coefficient can be used to replace the signal energy term as used for MFCCs.

<sup>11</sup>spoken as “delta” and “delta-delta”

## 2.3 Acoustic modelling using hidden Markov models

In Section 2.1 it was shown that within a probabilistic framework for speech decoding it is necessary to estimate the probability of a certain acoustic event given a particular word sequence or sentence. In the previous section a transformation of a speech signal in digital form into a sequence of  $T$  observation vectors  $\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t, \dots, \mathbf{o}_T)$  was described. Due to a fixed frame rate the number of observation vectors for a particular word may differ considerably, depending on the pronunciation and the speed with which the word is spoken. Thus a paradigm for matching an arbitrary length sequence of continuous-valued events (feature vectors) to another arbitrary length sequence of discrete events (e.g. words) is necessary. If we consider the target sequence to be a sequence of discrete states with arbitrary names<sup>12</sup>, the required framework needs to be capable of segmenting the observation sequence into exactly the same number of segments as present in the target state sequence. Beside the general capability to provide this segmentation a measure for the goodness of fit for at least the complete utterance is required. To satisfy the basic Equation 2.2 this measure has to be an estimate of the probability of that utterance given the sentence  $\mathbf{W}$ .

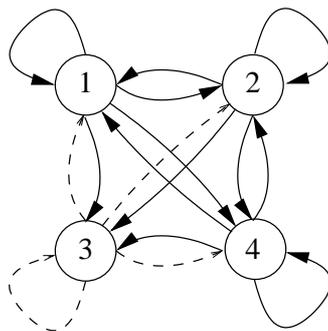


Figure 2.1 Ergodic Markov model

### 2.3.1 Basics

Hidden Markov models provide both segmentation and probability estimation capabilities. HMMs are an extension to Markov models which themselves are stochastic state machines with a finite set of  $N_S$  states  $S = \{q : 1 \leq q \leq N_S\}$ . For the operation of a stochastic state machine a pointer is needed to locate the state which is active at time  $t$ :  $q_t = i$ . The selection of the successive state  $q_{t+1}$  is probabilistic. Given that the probability distributions governing the transitions are constant, the sequence of states is a stationary stochastic process. In particular first order Markov models, which are the basis for hidden Markov models as used for speech recognition, make the assumption that the probability of entering state  $j$  at time  $t + 1$  depends only on the state at time  $t$ .  $\mathbf{q} = (q_1, q_2, \dots, q_t, \dots, q_T)$  denotes a particular sequence of states called the *path*. The probability of this sequence can be computed by

<sup>12</sup>This is only a naming convention, the states could for example be words.

$$P(\mathbf{q}) = P(q_1) \prod_{t=2}^T P(q_t | q_1 \dots q_{t-1}) = P(q_1) \prod_{t=2}^T P(q_t | q_{t-1}) \quad (2.3)$$

The second part of the equation uses the assumption of a first order Markov chain.  $P(q_1)$  is the prior probability of the chain starting in a particular state and  $P(q_t = i | q_{t-1} = j)$  is the probability of a transition from a particular state  $i$  to a particular state  $j$ . This allows us to draw Markov models in a simple diagram. Figure 2.1 shows a specific Markov model. Assuming that the process is started in state 3 a transition to all states in the Markov model including to itself can be made as outlined by the dashed lines in the diagram. The path so far has length 1 and its probability is  $P(q_1 = 3)$ . If a transition is made to e.g. state 2, the probability is multiplied by the transition probability  $P(q_2 = 2 | q_1 = 3)$  which can be thought of as attached to the arrows in the diagram. Naturally the sum of all the probabilities of links emanating from a state is one<sup>13</sup>. The interconnections between states define the *HMM topology*.

Hidden Markov models are a simple extension to the above concept. The sequence of states  $\mathbf{q}$  is a discrete-valued stochastic process which itself can be transformed into a continuous valued stochastic process by use of a stochastic mapping function. Each state corresponds to exactly one event which is either continuous or discrete valued. If  $\mathbf{O}$  is the observed sequence, the probability of that sequence can be computed using

$$P(\mathbf{O}) = \sum_{\mathbf{q}} P(\mathbf{O} | \mathbf{q}) P(\mathbf{q}) \quad (2.4)$$

where  $\sum_{\mathbf{q}}$  denotes the summation over all possible state sequences  $\mathbf{q}$ . While the computation of the state sequence probability is possible by Equation 2.3 the first term  $P(\mathbf{O} | \mathbf{q})$  is yet to be defined. Using

$$P(\mathbf{O} | \mathbf{q}) = \prod_{t=1}^T P(\mathbf{o}_t | \mathbf{o}_1, \dots, \mathbf{o}_{t-1}, \mathbf{q})$$

two important assumptions are made: the probability of the observed symbol at a certain time only depends on (a) the current state  $q_t$  and (b) is independent of any other observations:

$$P(\mathbf{O} | \mathbf{q}) = \prod_{t=1}^T P(\mathbf{o}_t | q_t)$$

In the case of continuous-valued observations the probability  $P(\mathbf{O})$  has to be replaced by the probability density function (PDF)  $p(\mathbf{O})$ . The value of this PDF for a certain observation sequence  $\mathbf{O}$  is called the *likelihood* of  $\mathbf{O}$ . Please note that we further assume only continuous

<sup>13</sup>This reflects the fact that a transition to some state has to be made.

valued observations since the modification of the appropriate equations for the discrete case is trivial. We define the two probability distributions per state, the set of transition probabilities

$$a_{ij} = P(q_t = j | q_{t-1} = i) \quad (2.5)$$

and the probability density function:

$$b_j(\mathbf{o}_t) = p(\mathbf{o}_t | q_t = j) \quad (2.6)$$

The overall likelihood computation simplifies to

$$p(\mathbf{O}) = \sum_{\mathbf{q}} \left( \prod_{t=1}^T p(\mathbf{o}_t | q_t) P(q_t | q_{t-1}) \right) \quad (2.7)$$

A speech utterance has finite duration. In order to force a well defined start and end, special start and end nodes can be incorporated into the HMM. Both nodes are assumed not to emit any observation vectors. A start node has no self transition and an end node has no exit transition. Figure 2.2 shows a simple HMM with non-emitting start and end nodes. As will be described in Section 2.3.5, the concatenation of HMMs for construction of larger models is an important technique. In this case the non-emitting state can serve as glue points where the end node of one HMM is assumed to be identical to the start node of its “successor”.

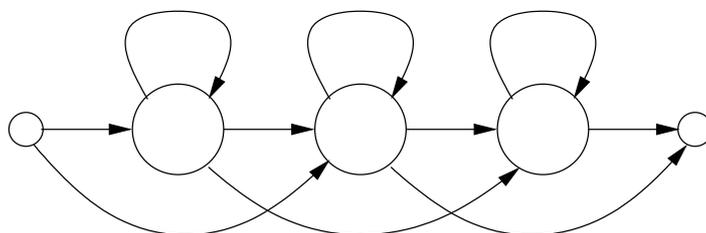


Figure 2.2 A simple 3-state left-to-right hidden Markov model with skips. Note that the minimum duration for this model is one frame.

A complete HMM definition consists of the topology where each state is defined by the associated output probability distribution function and the set of transition probabilities from that state to all the other states. If the HMM model parameters are denoted by  $\lambda$ , Equation 2.4 provides a way to compute the value of  $p(\mathbf{O} | \lambda)$ . If the HMM parameters are available for a finite set of isolated words  $\lambda_w$  the model parameters themselves are the “representatives” of a particular word. According to Equation 2.2 classification of a particular utterance  $\bar{\mathbf{O}}$  requires the computation of

$$\hat{w} = \underset{w}{\operatorname{argmax}} p(\bar{\mathbf{O}} | \lambda_w) P(w) \quad (2.8)$$

### 2.3.2 Probability computation

Equation 2.8 simply demonstrates the need to compute the likelihood of an HMM. Equation 2.4 contains a sum over all possible state sequences for a given utterance. The computational cost for this operation would in normal cases exceed the available resources by far. However, a much simpler implementation for the likelihood computation called the *forward-backward algorithm* is available. Following (Baum et al., 1970) we can define a set of forward/backward probabilities for HMMs. The forward probabilities for a particular model  $\lambda$  are defined as

$$\alpha_j(t) = p(\mathbf{o}_1^t, q_t = j | \lambda)$$

$\alpha_j(t)$  is the joint likelihood that the model generates an output sequence  $\mathbf{o}_1^t = \{\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t\}$  while the state sequence ends in state  $j$ . Using this definition and assuming that the states 1 and  $N$  are non-emitting entry and exit states, the forward probability values at all times and for all states can be computed recursively:

#### 1. Initialisation

$$\alpha_j(0) = \begin{cases} 1 & \text{for } j = 1 \\ 0 & \text{for } 1 < j < N \end{cases}$$

#### 2. Recursion for $1 \leq t \leq T$

$$\begin{aligned} \alpha_1(t) &= 0 \\ \alpha_j(t) &= \left( \sum_{i=1}^{N-1} \alpha_i(t-1) a_{ij} \right) b_j(\mathbf{o}_t) \quad \text{for } 1 < j < N \end{aligned}$$

#### 3. Termination

$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T) a_{i,N}$$

$\alpha_N(T)$  is identical to the probability of the complete utterance. An estimate of the probability of an utterance can also be computed using the backward probabilities

$$\beta_j(t) = p(\mathbf{o}_{t+1}^T | q_t = j)$$

which can be computed recursively in a similar fashion to the forward probabilities:

#### 1. Initialisation

$$\beta_i(T) = \begin{cases} 1 & \text{for } i = N \\ a_{i,N} & \text{for } 1 < i < N \end{cases}$$

2. Recursion ( $1 \leq t < T$ )

$$\beta_i(t) = \sum_{j=2}^{N-1} a_{i,j} b_j(\mathbf{o}_{t+1}) \beta_j(t+1) \quad \text{for } 1 < i < N$$

## 3. Termination

$$\beta_1(0) = \sum_{j=2}^{N-1} a_{1,j} b_j(\mathbf{o}_1) \beta_j(1)$$

Most importantly the combination of forward and backward probabilities can be used to yield important probability estimates. As mentioned before

$$p(\mathbf{o}_1^T) = \alpha_N(T) = \beta_1(0)$$

The joint probability of the complete observation sequence and an occupation of state  $j$  at time  $t$  is given by

$$p(\mathbf{o}_1^T, q_t = j) = \alpha_j(t) \beta_j(t)$$

and similarly the joint probability of observing  $\mathbf{o}_1^T$  and occupying both state  $j$  at time  $t$  and state  $i$  at time  $t - 1$  is:

$$p(\mathbf{o}_1^T, q_{t-1} = i, q_t = j) = \alpha_i(t-1) a_{i,j} b_j(\mathbf{o}_t) \beta_j(t)$$

The above measures are needed for the training of HMM parameters.

### 2.3.3 Output distributions

The general structure and use of HMMs for pattern classification has been outlined in previous sections. Nevertheless nothing has been said about the characteristics of the output probability distributions. The observation sequence can either be discrete or continuous-valued and thus the probability density function  $b_j(\mathbf{o}_t)$  either model discrete or continuous distributions. In early ASR systems the desire to keep model building complexity low as well as to limit the computational cost dictated the use of discrete distributions (e.g. (Jelinek, 1976)). Since the feature stream as described in Section 2.2.1 is continuous-valued a transformation into a discrete space via vector quantisation had to be made, which assigns one out of a finite set of labels to each observation vector. In this case the choice of an appropriate distance metric and clustering procedure becomes important.

HMMs as used in state-of-the-art systems however make use of continuous-valued probability distributions of parametric form. The selection of an appropriate output distribution depends on

the availability of a training scheme for that particular choice. In (Liporace, 1982) a relatively broad class of elliptically symmetric distributions is shown to satisfy the necessary criteria. Most importantly this includes the multivariate Gaussian probability density function:

$$p(\mathbf{o}) = \mathcal{N}(\mathbf{o}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{o}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{o}-\boldsymbol{\mu})} \quad (2.9)$$

where  $D$  denotes the vector dimension,  $\boldsymbol{\mu}$  denotes the mean vector and  $\boldsymbol{\Sigma}$  denotes the symmetric covariance matrix which in most cases is assumed to have diagonal form<sup>14</sup>. Apart from Gaussian distributions similar forms like Laplace distributions (Haeb-Umbach and Ney, 1992), power exponential distributions (Chen et al., 1999), and Richter distributions (Richter, 1986; Gales and Olsen, 1999) have been used in the recent past.

Unfortunately due to a variety of reasons the underlying distributions in speech modelling appear to go beyond the simple elliptical shape defined by Equation 2.9<sup>15</sup>. A simple but effective solution is the use of mixture distributions and again in particular mixtures of Gaussian densities:

$$b(\mathbf{o}_t) = \sum_{m=1}^M c_m p_m(\mathbf{o}_t) = \sum_{m=1}^M c_m \mathcal{N}(\mathbf{o}_t, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \quad (2.10)$$

where  $M$  is the number of mixture components and  $c_m$  are the mixture weights. Since the integral over the mixture PDF has to be one, the mixture weights have to meet

$$\sum_{m=1}^M c_m = 1$$

It is important to note that the covariance matrices themselves are subject to modelling choice. They may be selected to assume an identity, diagonal, block diagonal or a full matrix. HMMs in this thesis make use of *Gaussian mixture models* with diagonal covariance matrices.

An important question in conjunction with continuous Gaussian mixture models is the complexity of the model itself, i.e. the number of parameters which includes the number of mixture components. In many cases the number of components is determined globally<sup>16</sup> by measurement of the recognition performance. More sophisticated criteria are based on observation counts or information theoretic criteria. (Schwartz, 1978; Chen and Gopinath, 1999) try to adjust the number of components or free parameters on a per state basis. In general one is faced with the trade-off between accurate modelling of the training data and the requirement for generalisation.

<sup>14</sup>The individual components of the observation vector are assumed to be independent.

<sup>15</sup>A simple example for the existence of more complex structures is the difference of speech data per gender. The front-ends normally do not provide normalisation mechanism for those.

<sup>16</sup>All mixture distributions in a system have the same number of components.

### 2.3.4 Parameter estimation

In the previous section the principal structure, behaviour and use of hidden Markov models has been discussed. Two sets of parameters have been identified to be sufficient for the characterisation of the statistical properties of an HMM, the ones defining the time progression behaviour (Equation 2.5) and those relevant for determination of the output probability distributions (Equation 2.6). The use of Gaussian mixture distributions (Equation 2.10) requires the definition of three different parameters per distribution, namely the mean vectors  $\mu_m$ , the covariance matrices  $\Sigma_m$  and the mixture weights  $c_m$ .

The most important schemes for the estimation of model parameters are the *maximum likelihood* (ML) estimation and Bayesian estimation methods. Whereas the former assume that parameters are static but unknown the latter assume that the parameters themselves are random variables with some associated prior distribution. A choice of a particular parameter vector is made on the basis of the *a posteriori* distribution given some observed data. Most prominent is the maximum a posteriori training (MAP) scheme, which selects the parameter vector associated with the maximum of the posterior distribution. If the prior distribution over the parameter vector is assumed to be uniform over all possible parameters the MAP solution is identical to the maximum likelihood estimate. The maximum likelihood estimator is often preferred because of its simplicity, comparatively low computational complexity and the available algorithmic solutions. Furthermore under the assumptions of model correctness<sup>17</sup> and well behaved distributions the ML estimator is known to be consistent<sup>18</sup>. More detailed reviews of this topic can be found in (Duda et al., 2001) or (Huang et al., 2001).

Maximum likelihood training uses the likelihood of the observed data given a particular word as objective function

$$\mathcal{F}_{\text{mle}}(\lambda) = p(\mathbf{O}|\lambda, w)$$

In training the model set  $\hat{\lambda}$  for a particular word has to be chosen such that

$$\hat{\lambda} = \underset{\lambda}{\operatorname{argmax}} \mathcal{F}_{\text{mle}}(\lambda) \quad (2.11)$$

It is important to note that ML estimation criteria do not take information about competing classes into account. Hence using this method for training of classifier models relies on the separability of the classes and validity of the training data. Schemes which try to alleviate this problem by taking information about all competing classes into account while training one particular class are called discriminative training schemes (e.g. (Bahl et al., 1986; Chou et al., 1993)).

Unfortunately there is no closed form solution for solving Equation 2.11 for HMMs. Furthermore the surface described by the objective function has multiple local maxima, about which little can

<sup>17</sup>The training samples have been produced by the assumed model, i.e. a certain valid parameter vector plus the choice of distribution.

<sup>18</sup>A consistent estimator guarantees that if the number of training samples approaches infinity the estimate gets closer to the true value with convergence if an infinite number of samples is available.

be said mathematically. Thus any parameter estimation scheme can only hope to find the global maximum. Nevertheless a very general training scheme exists in the Baum-Welch algorithm (Baum and Eagon, 1967) which at least guarantees to reach a local maximum. This maximisation technique can equally be derived by using the more generic Expectation-Maximisation (E-M) algorithm (Dempster et al., 1977) and can be used for parameter estimation of arbitrary elliptic probability density functions (Liporace, 1982). The algorithm is based on the definition of an auxiliary function  $Q(\hat{\lambda}, \lambda)$  of the current model set  $\lambda$  and the re-estimated model set  $\hat{\lambda}$ . The definition of this auxiliary function guarantees that if some re-estimated model parameters  $\hat{\lambda}$  are found which globally maximise the auxiliary function, the overall likelihood for the new model set is higher than for the original model set. A more detailed explanation of this algorithm can be found in Section 4.1.1. The Baum-Welch algorithm is an iterative scheme whereby the parameters of an HMM, namely transition probabilities, means, covariance matrices and mixture component weights, are updated in each iteration. A detailed derivation and the parameter re-estimation formulae can be found in (Rabiner, 1993; Jelinek, 1997).

An important property of Baum-Welch re-estimation of HMM parameters is that the sum over all potential state sequences is taken into account in the re-estimation. The sum can be replaced by the maximum, i.e. the estimate  $P(\mathbf{O})$  is assumed to be almost identical to the  $P(\mathbf{O}|\mathbf{q}^*)P(\mathbf{q}^*)$ , where  $\mathbf{q}^*$  is the state sequence which obtains the highest likelihood. This type of approximation is called Viterbi approximation. In (Merhav and Ephraim, 1991) it was shown that the posterior probability of any state sequence which departs from  $\mathbf{q}^*$  in a single time instance decays exponentially with the number of frames. Consequently the exact and approximate estimates are close with a sufficient number of frames. The Viterbi approximation can be used to simplify and speed up the training process and is often used for methods where a clear assignment of observations to output distributions is desirable.

### 2.3.5 Sub-word units

The theory presented so far assumed that speech consists of isolated words and a separate model can be built for each word. Even in an isolated speech scenario this approach has disadvantages: the appropriate HMM topology of a word model is unclear; it is difficult to obtain sufficient training data for every word in a large vocabulary; the addition of a new word to the vocabulary requires new data.

If a spoken word can be split into parts based on units which may be shared with other words all of the above problems can be solved. Smaller units of which words are comprised are syllables, which have attracted more interest in ASR recently. Even smaller units are phonemes which represent an abstract notion of a sound, not its particular realisation. The HMMs associated with phonemes are called phone models. The use of phone models allows to implement simple HMM topologies which are often assumed to be identical for all phone models.

If multiple HMMs for a particular phoneme exist, we speak of allophone models. A common strategy is to use the phoneme to the left and to the right to select the *triphone* model. Since

the number of phoneme triplets is large not all will have been observed sufficiently often in the training data. Phonetic decision trees (Young et al., 1994; Odell, 1995) are a solution to this problem which allow a model to be selected on the basis of questions about the left and right neighbours. A more detailed discussion of this topic can be found in Section 3.5.

The use of sub-word units adds a new knowledge source, the pronunciation dictionary to the speech recognition system. This dictionary provides a mapping from words to sub-word units by use of a phonemic transcriptions. Each phoneme in the dictionary acts as an HMM selector. The phonemic transcriptions for each word are either produced manually by experts or automatically by grapheme to phoneme mapping algorithms. Thus the use of a dictionary allows the construction of word models without the need of any training data for that particular word.

## 2.4 Continuous speech recognition

This section describes the step from the recognition of isolated words to the decoding of complete sentences. Nevertheless the essential recognition paradigm stays the same. A particular word  $w$  is replaced by a sentence or word sequence  $\mathbf{W}$ . Training of model parameters remains unchanged if the proper concatenation of word HMMs is performed. Decoding of an observation sequence by the simple comparison of likelihoods for each possible sentence is not feasible due the almost unlimited number of potential sentences. Thus the Viterbi algorithm is used, which allows efficient decoding of complete sentences<sup>19</sup>. A description of Viterbi decoding can be found for example in (Huang et al., 2001) and the token passing strategy as used in the HTK decoder `HVite` is described in detail in (Young et al., 1989).

The complexity of continuous speech recognition is much higher compared to the classification of isolated words due to a number of effects in addition to the increase in search space. Since no pauses are usually made between words, the coarticulation effects normally seen within words also span across word boundaries. Modelling with sub-word units has to take this into account by selection of the appropriate sub-word unit HMM. The last phoneme of the preceding word serves as left context for the selection of the first phone HMM of a word. This has considerable impact on the computational complexity of the decoding network. Another effect of major importance for continuous speech is the language model (see Section 2.7).

## 2.5 Training of model sets

The theory of the construction of acoustic models for a speech recognition system was outlined in the previous sections. However, there remain important questions as to how to obtain the information needed to build such a model set. In addition to the availability of a dictionary an initial model set has to be trained. Since training does not necessarily yield the globally

---

<sup>19</sup>Note that the Viterbi algorithm uses an approximation to the likelihood based on the most likely state sequence. See Section 4.1.2 for a detailed discussion.

optimal parameters a whole range of different initialisation techniques have been used for the training of HMM sets. The likelihood surface has many local maxima that will be found by E-M re-estimations with different starting points.

The initial parameters are commonly chosen randomly or according to statistics obtained from other model sets. Bootstrapping with an HMM set, that provides the necessary initial alignment can yield considerably improvement of the parameter estimates. A number of re-estimation steps are performed until the increase in log-likelihood of the training data falls below a certain threshold. Between 4 and 8 re-estimation steps are normally required. If the bootstrap model was trained on different data, the initial alignment was potentially suboptimal and a second complete iteration of training using the model parameters obtained in the first iteration may yield further improvement. Most importantly this involves the restructuring of HMMs, for example, by retraining of phonetic decision trees.

The use of mixtures of Gaussians as the state output PDF introduces an additional level of complexity. Even if the data associated with a particular Gaussian mixture distribution is known, in contrast to the single Gaussian case a solution to find the global maximum of the likelihood function does not exist. Thus the initialisation of HMM sets becomes more important. Either a direct initialisation of HMM sets with mixture distributions or a gradual increase in the number of mixture components is used. The former relies on Viterbi alignment to obtain a hard assignment between output distribution and observation sequence. In consequence clustering of observation vectors, for example with k-means clustering (Duda et al., 2001), can be used to initialise Gaussian mixtures. The second *mixing up* scheme tries to progressively increase the number of mixture components in the model set interleaved with multiple parameter re-estimation steps (Young and Woodland, 1994). Whereas the use of clustering has the advantage that the full power of the model is available from the start of the training process the second method relies less on the quality of the initial alignment. Nevertheless the possibility of poor alignment in training is still a problem since single Gaussian models inevitably display more between-class confusability.

Another important matter is the modelling of non-speech events. In previous sections only the modelling of pure speech was described ignoring the fact that the stream of words is interrupted by pauses. These pauses or so called silence segments however do not only contain silence but also human or non-human generated noise like laughter, street noise or the closing of doors. Thus silence models are usually more powerful than phone HMMs and may be subdivided to model certain noise types specifically, if labelled data is available. Work in this thesis makes use of one model for silence, which is mainly used at the sentence start and end, and a model for between word silence which can be skipped and preserves phoneme context across words (*crossword* modelling).

## 2.6 Language modelling

In Section 2.1 the basic speech recognition paradigm was defined as the solution of Equation 2.2. In continuous speech recognition  $\mathbf{W}$  denotes a sequence of words with unknown length. For obvious reasons a model that provides an estimate of this probability is called a *language model*. Estimates for the prior probability of word sequences are not only used for speech recognition. Other tasks such as optical character recognition, handwriting recognition and machine translation require similar models. It is convenient to decompose the overall language model probability for a sequence of  $L$  words into the product

$$P(\mathbf{W}_1^L) = \prod_{l=1}^L P(w_l | w_{l-1}, w_{l-2}, \dots, w_1) \quad (2.12)$$

of conditional probabilities of a word in position  $l$  given its word history  $w_1^{l-1}$ . Thus if the probability of a certain word sequence is known, the probability of a sentence with one extra word can be computed efficiently by simple multiplication. This allows speech recognisers to process the input speech continuously. If the vocabulary size is small and sufficient training data is available, providing an estimate for the above conditional probabilities inevitably becomes easier. For large vocabulary speech recognition using vocabularies of more than 20000 words and sentences exceeding length 3 it becomes obvious that no recogniser is able to store explicit estimates for all potential word sequences let alone provide sufficient coverage in the training data. Thus it is necessary to restrict the parameter space in order to achieve reasonable sample coverage and robust probability estimation. Better coverage can be achieved by clustering of the set of possible word histories  $\mathbf{W}_1^{n-1}$  into equivalence classes  $h(\mathbf{W}_1^{n-1})$ . The objectives of language modelling are therefore to define reasonable equivalence classes and estimators which yield optimal estimates

$$P(\mathbf{W}_1^L) = \prod_{l=1}^L P(w_l | h(\mathbf{W}_1^{l-1}))$$

This can be seen as a finite state grammar approach where the grammar state is changed with every word (Jelinek, 1997).

### 2.6.1 N-gram language models

One straightforward way of clustering word histories is the simple truncation after a certain number of words, for example in the case of a *trigram* language model

$$h(\mathbf{W}_1^{n-1}) = (w_{n-1}, w_{n-2})$$

i.e. the set of history equivalence classes is confined to the set of all possible word pairs. The straight-forward maximum likelihood solution leads to the trigram language model where probabilities are estimated using

$$\hat{P}(w_l|w_{l-1}, w_{l-2}) = \frac{N(w_l, w_{l-1}, w_{l-2})}{\sum_w N(w, w_{l-1}, w_{l-2})} \quad (2.13)$$

where  $N(\cdot)$  denotes the frequency of word triplets on the training data. Thus in order to provide reliable estimates of the above probabilities the training data has to cover the set of possible word triplets sufficiently<sup>20</sup>. Given a 60000 word vocabulary this amounts to  $2.16 \times 10^{14}$  trigrams. Even though some word triplets may indeed have a probability of zero, complete and sufficient coverage is unachievable. Therefore either models with shorter contexts such as *bigram* or *unigram* models are used<sup>21</sup>. In reality large span dependencies have considerable influence on word probability and thus are desirable to use. In order to achieve robust estimates *smoothing* of the probability mass distribution  $P(w|h)$  is employed. The distribution needs to be smoothed because words may be unseen in the training data or the training data actually may be unbalanced.

One important smoothing technique for N-grams tries to smooth probability estimates obtained from the counts on the training data by allocating a certain amount of the overall probability mass to events which have a count of zero. This method is called *discounting* whereby the primary counts as used in Equation 2.13 are reduced by a certain factor. However, this still allows a range of different ways to compute the discounting factors. The most prominent discounting methods are Good-Turing discounting (Good, 1953; Katz, 1987), Witten-Bell discounting (Witten and Bell, 1991), absolute discounting (Ney and Essen, 1993) and floor discounting (de Mori, 1998).

Discounting alone suggests that the discounted probability mass needs to be uniformly distributed over the remaining unseen events given a certain history. Given that this brings the problem that some of the unseen events are “impossible” events and thus should never obtain a probability greater than zero a better way to model the probability of unseen events was introduced by (Katz, 1987). *Backoff* makes use of distributions which are less refined in history resolution and thus can be estimated more robustly. These distributions are called backoff distributions. The probability distribution over unseen events is taken from the backoff distribution after proper normalisation to the discounted probability mass. Naturally, hierarchical backoff schemes are employed using a strategy to back off from, for example, 4-gram distributions to trigram, bigram and ultimately unigram distributions.

Another important scheme makes use of less specific distributions in a more explicit fashion. The set of unigram, bigram and trigram distributions are interpolated using weights which have to

<sup>20</sup>Word triplets which are at all possible are normally required to appear in the training data.

<sup>21</sup>The unigram model is just the prior distribution over single word occurrence whereas the bigram model exploits dependence on the previous word.

satisfy a sum-to-one constraint. The weights may be adjusted manually to obtain optimal performance on some test set. Alternatively, by using deleted estimation (Jelinek and Mercer, 1980) the interpolation weights may be obtained in a re-estimation procedure based on the E-M algorithm (Dempster et al., 1977). In this case interpolation weights may depend on word history or frequency thereof. Backoff and deleted estimation schemes exhibit comparable performance for speech recognition (Katz, 1987).

## 2.6.2 Scaling and insertion penalties

In practice the combination of language models and acoustic models requires some adjustments to account for certain properties of the statistical estimates. Language modelling tries to provide estimates for sentence probabilities which are usually based on N-gram models trained on large corpora<sup>22</sup>. Even though severe constraints on the modelled dependencies are commonly made these constraints together with a larger training set for language modelling than for acoustic modelling usually makes LM estimates more robust. Furthermore estimates for acoustic likelihoods are assumed to be much too small due to unaccounted observation interdependence. Thus a considerable difference in dynamic range is found. A well known solution is the scaling of the log language model probabilities by a certain factor which normally is assumed to be constant for a particular task.

Another adjustment related to language modelling is the use of word insertion penalties. Word insertion penalties additionally penalise a higher number of words in a sentence. This is desirable since a considerable proportion of recognition errors stem from the insertion of words with a small number of phonemes. The reason for these insertions is the low acoustic “cost” combined with high probability of occurrence for a wide range of contexts. Insertion penalties are used to improve the balance of word insertions versus deletions. If analysed in the context of language modelling a subtraction of a constant factor from the log probability is equivalent to dividing the probabilities by a certain factor. Similar to linear discounting the probability of all words is scaled down by a fixed amount. However since for smoothed language models probability mass is assigned to unseen events in the training data, an interpretation is that the discounted probability mass is allocated to words unseen in the training data. An incomplete vocabulary would increase the out-of-vocabulary rate on test data and consequently the rate of insertions and substitutions.

## 2.7 Decoding

Decoding or recognition is the search for the best possible word sequence  $\hat{\mathbf{W}}$  given an observation sequence  $\mathbf{O}$ .

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmax}} P(\mathbf{W}|\mathbf{O}) = \underset{\mathbf{W}}{\operatorname{argmax}} P(\mathbf{O}|\mathbf{W})P(\mathbf{W})$$

<sup>22</sup>For example Broadcast News language model training data typically exceeds 300 million words.

The optimal word sequence is the one sequence from the set of all possible word sequences which has the highest posterior probability given the acoustic and language model. Using Equation 2.4 this can be extended to include the sum over all state sequences:

$$\hat{\mathbf{W}} = \operatorname{argmax}_{\mathbf{W}} P(\mathbf{W}) \sum_{\mathbf{q}} p(\mathbf{O}|\mathbf{q}, \mathbf{W})P(\mathbf{q}|\mathbf{W}) \quad (2.14)$$

The evaluation of the above equation is computationally very expensive if the number of words in sentences increases to a reasonable amount. The sum over all possible state sequences is computationally costly. In order to avoid this computation the sum in Equation 2.14 can be approximated by the maximum:

$$\hat{\mathbf{W}} = \operatorname{argmax}_{\mathbf{W}} P(\mathbf{W}) \max_{\mathbf{q}} p(\mathbf{O}|\mathbf{q}, \mathbf{W})P(\mathbf{q}|\mathbf{W}) \quad (2.15)$$

Using Equation 2.15 as the recognition paradigm the search for the best word sequence is called Viterbi decoding. It allows us to exploit the fact that all paths leading to a certain state in the space of all possible sentences compete against each other. This means that any progression beyond that state has to choose from one of these paths. Obviously the most likely path has to be selected at that point. Thus the search for the most likely state sequence is most commonly implemented in a time-synchronous fashion and makes use of networks of HMMs. These networks are considerably expanded by the use of context dependent sub-word units across word boundaries. Another factor which leads to an increase in the size of the decoding network is the use of language models. As discussed in Section 2.6 the probability for a word depends on its predecessors. This implies that paths through the networks have to be kept separate between different word histories, in the case of trigram language models depending on two predecessor words. The efficient implementation of Viterbi decoders is the topic of ongoing research.

If non-standard methods for the estimation of acoustic probabilities are tested, the implementation of a full decoder can often become too difficult or computationally infeasible. Nevertheless the algorithms need to be tested in a state-of-the-art speech recognition framework. The commonly adopted solution to this problem is the use of a set of alternative word level outputs generated by some reference system on the test data in so called *acoustic rescoring* experiments. The set of alternatives may be either in the form of a list of  $N$  best hypotheses or in the form of word lattices. Word lattices are directed graphs which for this purpose contain the appropriate language model probabilities. The use of such representations of the test data drastically limits the search space and in the case of  $N$ -best lists allows us to fall back to a linear search strategy. In the case of word lattice rescoring the lattices serve as constrained word networks. Standard Viterbi decoding is required to obtain the best word sequence. In both techniques the assumption is that the system under investigation is not very different to the one which was used to produce the representation of the test data and that the factor  $N$  or the lattice size is sufficiently large.

---

## *Hierarchies and sharing in automatic speech recognition*

---

As outlined in Chapter 2 the acoustic modelling part of automatic speech recognition systems makes use of models representing a word sequence: a wordlist to define words and thus the search space; a dictionary that defines the translation into sub-word units; and an acoustic matching model which maps the sequence of sub-word units onto the observation sequence. The following sections describe the specific modelling and interrelation between these parts from the perspective of a hierarchical system with special focus on the selection of HMMs for the modelling of sub-word units. Each level in this hierarchy is often presumed to be functionally independent from all other levels. In practice, the independence is not accurate and often has to be corrected by the reorganisation of other levels. Among the many techniques of structuring the multiple levels of a speech recognition system, the ones most relevant to work in subsequent chapters are presented. In particular, special focus is given to sharing, tying and clustering of acoustic units of arbitrary size and to pronunciation modelling of spontaneous speech.

The first section discusses hierarchical systems in general and state of the art speech recognition system structures. The following sections are devoted to modelling at the word, pronunciation, HMM and state level, and the last section gives an overview of techniques which make use of level combinations.

### **3.1 From sentence to signal**

Acoustic modelling in speech recognition is concerned with the search for methods to provide estimates for the probability of an utterance given a certain word sequence. In the case of continuous speech one is faced with an enormous acoustic observation space which can never be sufficiently covered by training data. The obvious solution to facilitate the modelling of a system of this complexity is to split the overall system into parts which ideally have the following properties:

- Independence and separability

The functionality of each sub-system is independent. The behaviour of one particular part

is defined without referring to the existence of the remaining parts. The system can be disassembled into its parts and each part can be investigated independently.

- Low dimensional interface  
The dimensionality of inputs and outputs of sub-systems is minimal.
- Minimum complexity  
System parts are simple in structure. They allow the use of simple models which can be trained.

In practice, the independence assumption is often violated due to constraints given by the demands on sub-system complexity or due to insufficient knowledge about the sub-systems. The structure governing the interrelation between sub-systems has to be seen as a system itself for which low complexity is desirable. The system governing the interrelation between sub-systems stands higher in the hierarchy of systems.

The representation of the acoustics in automatic speech recognition requires a highly non-linear and complex system. Apart from the case of very simple speech recognition systems (see Section 2.1) a split into subsystems is necessary. Fig. 3.1 represents a top-down view of an HMM based ASR system. The overall process is organised in strictly hierarchical fashion. Sentences are mapped onto words sequences mainly by use of a wordlist. Next the individual words are replaced by the appropriate sub-word units. The sub-word units are most commonly phoneme-like symbols and the mapping is based on pronunciation dictionaries generated by human experts. Each phoneme is mapped into a stochastic state machine where one state contains a PDF consisting of a mixture of probability distributions. Each state can produce a theoretically infinite number of observation vectors, which then can be used to produce intelligible speech<sup>1</sup>.

Each of these levels is often assumed to be independent and optimisations of a certain level are first carried out independently (Strik and Cucchiarini, 1999) before final testing in a complete speech recognition framework. Note that this particular hierarchical structuring of the speech recognition process is essentially based on time separation of units within each level. Thus models are mostly concerned with modelling of interdependence between the units of the current level. The structure of this hierarchy is derived from the understanding of speech recognition process by humans. However, there are several disadvantages. The behaviour of subsystems can only be tested indirectly and secondly the chosen structure may not necessarily reflect the optimal structure for classification of continuous speech by machine. Another disadvantage of this structure is the strictly perceptually based procedure<sup>2</sup>. Processing of audio signals by the inner ear organs is at least accessible to some degree, whereas understanding of neural processing is beyond human reach. Thus the assumption that pronunciations are sequences of sub-word units which belong to unique classes may be suboptimal and is the subject of current research

---

<sup>1</sup>As discussed in Section 2.2.1 important information about speaker and intonation are lost at the observation level. It is assumed that they could be artificially regenerated. The resulting speech signal would inevitably sound different.

<sup>2</sup>This of course excludes speech production normalisation which may be performed in the front-end.

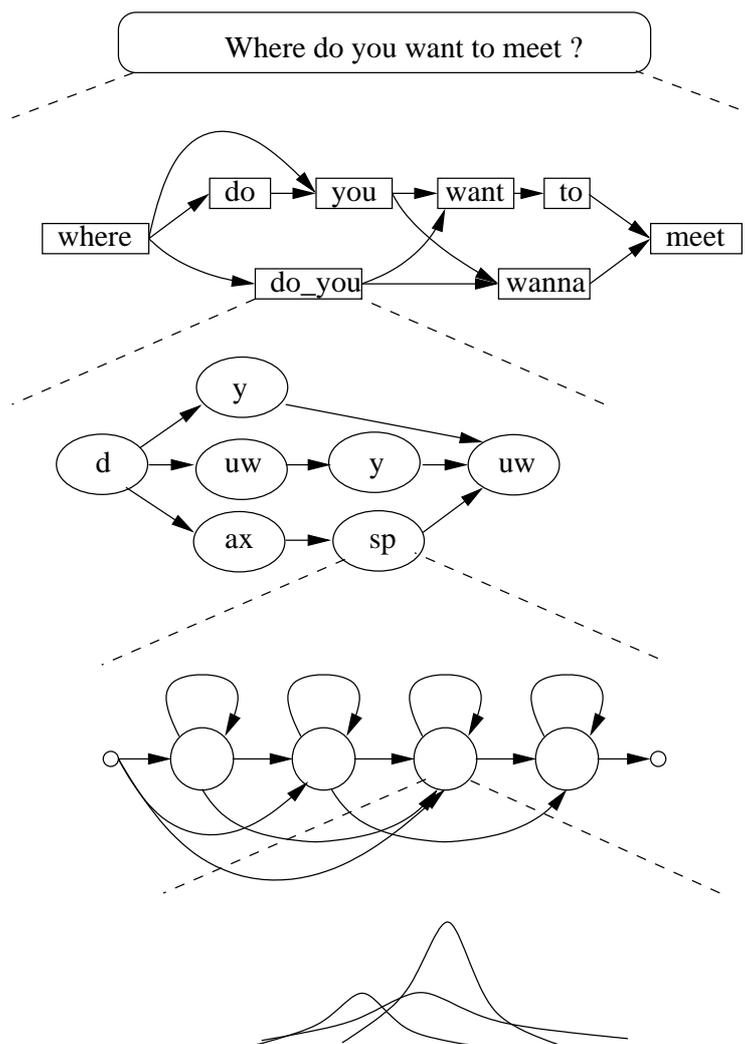


Figure 3.1 *Hierarchies in automatic speech recognition*

(Ostendorf, 1999; Young, 2001). More recently steps have been taken to split up phone models, either by use of a phonological feature based description of speech (e.g. (Eide et al., 1993; Deng, 1997a; Deng, 1997b)), by classification of articulator movements (e.g. (Richardson et al., 2000; Sun et al., 2000)) or by use of factorial HMMs to model multiple parallel feature streams.

These methods extend modelling to explicitly incorporate the speech production mechanism at a high level. Even though a new paradigm for acoustic modelling is highly desirable, the structure outlined in Fig. 3.1 is still the most successful.

Each of the levels intends to deal with a certain range of speech effects and many techniques have been tested to improve the performance of each level on its own and in combination. The most important distinction between techniques at each level appears to be the separation between data-driven and knowledge driven approaches. In many cases a mixed approach prevails. In the recent past the individual levels and their interactions have been investigated in combination and this thesis is no exception in that respect. In particular the main focus of this thesis is to introduce a framework for combined optimisation of levels. To prepare the ground for the

hidden model sequence modelling as presented in Chapter 4 the organisation and structuring of HMMs in state of the art systems are discussed in the following sections for acoustic modelling of word, pronunciation, HMM and HMM state levels.

## 3.2 Word level

A particular word sequence is associated with a certain range of acoustic realisations and thus acoustic observation sequences<sup>3</sup>. Since sentences are sequences of words which cannot overlap in time the signal may be split into segments associated with words. Whereas the definition of a word is trivial for read speech, in spontaneous or conversational speech sounds which normally do not have a well defined written representation are used to convey information as well<sup>4</sup>. Additionally some short sequences of words are combined into a single acoustic event for which the individual words cannot be temporally separated or associated with individual acoustic events, for example *dijha* instead of *did you*. Thus in this case the definition of words has to incorporate the needs of both the acoustic and linguistic representations. A word has to be understood as an interface unit between the locally operating acoustic model and the language model which is concerned with wider context. An important example for the extended notion of words used in a recognition dictionary is the use of multi-words (e.g. (Finke and Waibel, 1997b)) which explicitly model phone or even syllable reductions present in spontaneous speech (Greenberg, 1998). A simple example of this reduction is the use of *going\_to* instead of two separate dictionary entries. Work done at SRI (Stolcke et al., 2000) shows that substantial improvements can be made by both incorporation of a relatively large number of multi-words into language-models and acoustic models. However, experimental results on the use of multi-words have not been consistent among different research groups (e.g.(Ma et al., 1998; Nock and Young, 1998)).

Word level transcripts define the structure of the acoustic model for a particular sentence in training. For conversational speech in particular the transcription of the recorded speech is difficult and the level at which the data should be annotated is often left to speculation. The presence of hesitations, false starts and interjections in the speech signal makes the presence of symbols describing these events in the labelled sequence highly desirable since these are events which are separable in time. The level of detail in the annotation of phonetic modification of particular words necessary to enable improved speech modelling however is unknown. Increased labelling detail is expensive and is normally avoided or is only done on small corpora (e.g. (Greenberg, 1996; Fisher et al., 1986)) for the purpose of analysis. On larger datasets pseudo-phonetic labelling of some very frequent and consistent acoustic effects is taken into account (Shaffer and Picone, 1998). Whereas these problematic effects at the word level are normally ignored in the training of HMMs, a more general network based training scheme which allows slight modification of the training transcript was shown to be beneficial (Finke and Waibel, 1997a).

---

<sup>3</sup>This implies that there is significant loss of information about the speech signal in speech recognition.

<sup>4</sup>For example confirmation or rejection sounds.

The ambiguity in word level transcripts especially for spontaneous and conversational speech, has stimulated the question as to what is the required information from a system. In this context the extraction and identification of named entities allows the recogniser design to focus on the desired information (Weintraub, 1998).

### 3.3 Pronunciation level

Pronunciation modelling is motivated by the structures, expressions and levels used in phonetics. Given a phone, the human brain is assumed to perceive this sound as a particular phoneme (O'Connor, 1973; Hawkins, 1988). A phoneme is defined as the “the smallest meaningful contrastive units in the phonology of a language” (O’Shaughnessy, 1987). The important assumption is that a phoneme only can be regarded as such if words exist for which a difference in phonemes implies a difference in words. Ideally phonemes are defined by minimal pairs where the meaning of a word differs if only one phoneme is changed. Thus the definition of a phoneme is specific to a certain language. The reverse definition of a phoneme makes its relationship to phones clearer: “Phonemes are functional units of which phones are realisations” (O’Connor, 1973). Each phone is associated with one phoneme. “A class of phones which correspond to a specific acoustic variant of a phoneme” (O’Shaughnessy, 1987) are called allophones. Given the normal use of symbols in dictionaries used in speech recognition (see Appendix A) the pronunciations are closer to a phonemic rather than to phonetic transcription of words. In this sense the remainder of this thesis will speak of phonemes denoting the sub-word symbols present in the dictionaries, regardless of the actual content. The term phone will be used only to describe a speech sound or a particular HMM designed to represent that specific sound.

Pronunciation modelling has attracted widespread attention over the years. Whereas sometimes the main task of acoustic modelling is regarded as modelling pronunciation variation (Strik and Cucchiarini, 1999), pronunciation modelling most commonly is concerned with the definition of sub-word units and techniques which allow words to be described with them. Much of the research in this area has used clean read speech and often has disregarded the acoustics completely, although schemes which take the acoustics into account clearly perform better on read speech. Only a brief review of various pronunciation modelling techniques used in particular for modelling of the increased variation in spontaneous American English is given here. For more detailed reviews on pronunciation modelling the reader is referred to (Strik and Cucchiarini, 1999) or (Humphries, 1997).

Manually and automatically generated rules for generation of pronunciation variants have been shown to be effective for modelling of read speech. Spontaneous speech however shows much greater variability in speaking style. An interesting experiment initially conducted by (Weintraub et al., 1996b) and repeated by (Saraçlar et al., 2000) compares conversational speech with other instances of the same sentences: a read version; and a read version imitating the conversation. The experiments show that models trained and tested on read speech show significantly better performance than the original spontaneous versions by more than 15% word error rate (WER)

absolute and performance on imitated speech is very close to that obtained for read speech. Since these experiments deal with data recorded over telephone further experiments in (Saraçlar et al., 2000) and (Bernstein et al., 2000) show that this is not the cause of these differences.

In (Saraçlar, 2000) another experiment was conducted to find evidence of the nature of pronunciation variation in spontaneous speech. In this experiment, a small set of the Switchboard corpus was annotated using manual phone labels (the surface forms) and secondly using a dictionary (the canonical or base forms). Surface and baseforms were aligned to derive a third labelling of the data by pairwise combination of the labels (the combined form). Independent HMMs using only a single Gaussian output distribution were trained on all three types of data labels. It is possible to compare the mean vectors of all three models by projection onto the plane spanned by these vectors and proper normalisation. The result are plots with normalised locations of mean vectors on this plane for surface and baseforms. The points representing the mean vectors of the combination models show no clear preference towards base or surface form and appear to assume a comparatively large distance to both versions. One potential conclusion is that in the case of a non-canonical pronunciation of a word neither the canonical nor the manually labelled surface form appears to be “correct”<sup>5</sup>. This suggests that other effects beyond contextual dependence in pronunciation are the cause for the diversification. In this case a hard decision on certain phone models may be inappropriate. (Saraçlar, 2000) proposes the use of soft-tying of HMM states for pronunciation modelling (see Section 3.5).

Research into pronunciation modelling of spontaneous speech such as the study just mentioned, greatly benefitted from the manual phonetic labelling of 72 minutes of speech of the Switchboard corpus (Greenberg, 1996). Considerable differences between the hand-labelled and the canonical transcription was found with a phone error rate of 78% and a phone deletion rate of 12.5%. The data was used as basis for the construction of pronunciation networks (Wooters and Stolcke, 1994) using decision trees (Weintraub et al., 1996a; Byrne et al., 1998; Riley et al., 1999; Fosler et al., 1996). The methods under investigation also attempted to model insertion and deletion phenomena in spontaneous speech. However only moderate improvements in word error rate were achieved while the process involved retraining and clustering. These results suggest that pronunciation modelling by use of pronunciation variants or networks can not be optimised independently in the case of spontaneous speech.

Joint optimisation of a standard phoneme based dictionary and HMM models was used by (Seong-Yun and Oh, 1999). A phonotactic bigram model was optimised simultaneously with the underlying discrete HMM parameters. Substantial improvements on a small vocabulary task were achieved in this case.

---

<sup>5</sup>Strictly speaking the conclusion can only be drawn if the underlying segment boundaries would actually stay constant. This cannot be guaranteed in the experimental framework chosen.

### 3.4 Model level

As discussed in Section 2.3.5, the standard sub-word units for HMM based speech recognition are determined by the phoneme and its left and right neighbour. Too many of these triplets exist to be able to train phone models for each of them. Clustering schemes using an entropy based distance metric yield so called generalised triphone models (Lee, 1990). These triphone models used VQ codebooks and thus contained discrete probability distributions. In order to deal with the problem of unobserved phoneme triplets (*tri-phonemes*) the models were interpolated with monophone models. In the case of continuous output distributions the interpolation of models is less obvious, requiring another solution for the case of unseen phoneme triplets. This can be provided by the use of phonetic decision trees (Hwang et al., 1993; Young et al., 1994) on the model level. Since the use of phonetic decision tree based clustering was first introduced on the state level and is mostly used on that level, the specifics of this method are discussed in the following section. In case of (Hwang et al., 1993) clustering is based on a similar criterion as used for generalised triphones whereas (Young et al., 1994) transformed clustering at the model level into state level clustering by representing a complete HMM with a single Gaussian distribution.

Another important question is the construction of the proper HMM topology. The standard topology is based on the effect of coarticulation and subdivides a model temporally into starting, middle and ending parts. The automatic learning of HMM topology at the phoneme level can be interpreted as an alternative to clustering of HMM states. One of the most important works in this area was presented by (Ostendorf and Singer, 1997), which implements another form of divisive clustering of states based on the maximum likelihood criterion and which makes use of the phoneme context. However, in addition to the contextual split of states a temporal split is also possible. Consequently each phone model consists of a so called HMnet which contains many parallel paths with a varying number of states. Phoneme recognition experiments indicate a smaller gain for temporal splits than for contextual splits in experiments on read speech data.

Another more recent approach presented by (Richardson et al., 2000) is to construct specific HMM topologies on the basis of articulator configurations. A diphone model is constructed from a set of states, where each state is characterised by its specific articulator configuration in terms of a discrete vector. The model can be constructed by the use of all possibilities to make the transition from one articulator configuration to another at the boundaries of the model. Substantial improvements in word error rates on an isolated word recognition task (Phonebook) were obtained.

### 3.5 State and mixture component level

Modelling at the HMM state and mixture component level is mostly concerned with either sharing or tying methods or combinations thereof to achieve suitable coverage of the training data and generalisation to new data. In this process information from higher levels such as phoneme

context is used. It is important to note that work at this level often disregards the HMM framework. In order to do so, Viterbi alignments of the data at the state level are used to arrive at a unique assignment of data points to each distribution.

The clustering of states rather than that of complete HMMs was presented by (Hwang and Huang, 1992) and used a weighted entropy measure as the distance metric within generalised triphones. The clustering operates across all states and the set of clustered states are called *senones*<sup>6</sup>. Agglomerative clustering of states using a Gaussian divergence based distance metric was proposed by (Young and Woodland, 1993). In this case triphones are merged implicitly if all associated states belong to one cluster. In (Hwang et al., 1993) generalised triphones were replaced by phonetic decision trees and operated on the state level. A sum of weighted entropy measures still serves as distance metric for clustering. In (Young et al., 1994) the divisive clustering process using phonetic decision trees is based on the balance between maximising the likelihood of the training data and sufficient observations per state. In order to facilitate this optimisation important assumptions had to be made: the alignment to state sequences is not altered in the clustering process, the output distributions to be clustered are single Gaussian PDFs and state transition probabilities are ignored. The latter is usually of little importance whereas both other assumptions in conjunction are potentially only reasonable on relatively clean speech data. Models of low complexity used for data with high variability result in poor quality state alignments which are altered considerably by only a small change of the model parameters. This and the issue of using single Gaussian PDFs is addressed in experiments presented by (Nock et al., 1997).

Semi-continuous HMMs (Huang and Jack, 1989) avoid the selection of an appropriate output distribution by sharing of Gaussian distributions across all states in the system. In (Huang and Jack, 1989) the set of Gaussian probability distributions were understood as codebooks which may be adjusted jointly with the state-specific codebook weights using the E-M algorithm. In (Huang et al., 1991) the technique was combined in a clustering scheme derived from generalised triphones and in (Hwang et al., 1994) independent sets of Gaussians were used for each phoneme.

The sharing of Gaussian output distributions was re-formulated in the tied mixture (TM) model (Bellegarda and Nahamoo, 1990) with arbitrary sharing of the mixture components. However, since the number of mixture components in complex recognition systems can be enormous the distributions over all mixture components may become sparsely populated. In order to avoid this problem in more recent implementations the mixture components are phonetically tied or state clustered (Nguyen et al., 1995). Another option is to use clustering to obtain sets of mixture components which define the sets of mixtures for tied mixture modelling (Digilakis and Muvvetts, 1994).

More recent work related to tied mixture models and semi-continuous HMMs was presented by (Luo and Jelinek, 1999). The idea is to share Gaussian distributions among the leaf nodes of

---

<sup>6</sup>The name is derived because of their relationship to *fenones* which are discussed in Section 3.6.

phonetic decision trees in non-reciprocal fashion. This effectively is a constrained and specially initialised version of tied mixture models. The technique allows some of the hard decisions made by state clustering to be corrected. An alternative implementation is presented in (Hain et al., 2000).

### 3.6 Combination of levels

At each level in the standard speech recognition framework, modelling is ideally driven by the specific properties of speech associated with the particular level. Pronunciation modelling has to provide a proper phonemic transcription, whereas the models on the model level are assumed to represent phones. The state and mixture component level provides a stochastic match to the observation vectors. However, the system boundaries in this form do not exist and the different levels interact. This opens the door for methods which operate beyond the particular level. However, these methods have to purely rely on data driven approaches, which are more difficult to control.

An important framework, fenones, which allows the problem of the proper selection of word pronunciations to be avoided, was introduced in (Bahl et al., 1988). Fenones are small sub-word units represented by single state HMMs with potential skips. The desire is to obtain a single purely data driven baseform transcription in form of a sequence of fenones. That sequence is obtained on a number of utterances for each word in the vocabulary. This was later extended to use a fenonic network (Bahl et al., 1991). More recent approaches to define acoustic units on the basis of acoustics usually make use of modified acoustic modelling such as work presented by (Bacchiani et al., 1996) based on segment models (Ostendorf et al., 1996). The model inference process for trajectory based segment models is purely driven by acoustics regardless of data labelling. This breaks the link between acoustics and text and two problems have to be solved. First the relationship between words and the acoustic units has to be found. This is done by building a statistical model for the phoneme to model relationship. This model then is used for generation of a “pronunciation” dictionary. Secondly, models may span word boundaries, which forces the use of multiwords in the vocabulary and in turn new language modelling approaches are necessary. In (Deligne and Bimbot, 1997a) the inference of new acoustic models is based on temporal decomposition of observation vectors and heuristic definition of associated HMMs. The set of HMMs is used to represent the speech data as a sequence of these models. The mapping between the arbitrary length sequence and the associated phoneme sequence is modelled using multigrams (Deligne and Bimbot, 1997b).

### 3.7 Summary

In this chapter a hierarchical view of speech recognition was presented and techniques related to work presented in this thesis have been described. Spontaneous speech effects pose a significant

problem to the acoustic modelling of speech. Most efforts to address the issues, which are presumably involved in the dramatic increase in word error rate compared to other speech types, operate at the pronunciation level and have not so far shown the anticipated performance improvements. However, especially for modelling of spontaneous speech, the structure and performance at different levels is not independent. The joint optimisation of multiple levels is most often approached in conjunction with a complete change of the modelling paradigm.

Modelling at higher levels can always be interpreted in terms of changes at a state or mixture distribution level. It is simple to formulate generic frameworks on this level which in theory host a variety of methods. However, the training of speech recognition systems is based on insufficient data and non-optimal algorithms and thus cannot completely rely on self-organisation required by these frameworks. Apart from steps to completely new modelling approaches, the choice is to either find a proper organisation heuristically or to search for methods which make the necessary self-organisation feasible.

---

## *Hidden model sequences*

---

In this chapter the use of hidden model sequences for automatic speech recognition is introduced. Hidden model sequences (HMS) represent a framework with which a range of different statistical pronunciation modelling and state selection techniques can be described. The objective is to provide a framework in which pronunciation level, model topology and state selection can be described in a uniform way and can be optimised simultaneously. In this framework the concept of a stochastic mapping from a sub-word unit to a particular HMM is introduced. This new mapping can be represented by an additional model, the model sequence model, which should capture model uncertainty in relation to sub-word units. In principle the nature or interpretation of these units is arbitrary. If no particular meaning is assigned to the individual units one can exploit two degrees of freedom to obtain the optimal mapping. A flexibility in the substitution of models can coincide with a flexibility in temporal assignment. The latter is assumed to be important for modelling of spontaneous and conversational speech, where deletion and insertion effects may alter the phonetic realisation of words. The method presented here does not necessarily require the concept of phone models. As long as segments can be assumed to have short-time stationary characteristics optimisation strategies for obtaining a suitable set of sub-word models exist. Due to its hierarchical nature the framework combines the flexibility of unit selection schemes with the advantage of a strict hierarchy to predict the acoustic realisation of words which are unseen in the training data.

Section 4.1 gives an outline of the theoretical framework whereas Section 4.2 discusses the potential realisations of model sequence models. Section 4.3 discusses decoding of the speech signal with HMS-HMMs. A summary concludes this chapter.

### **4.1 A hierarchical formulation of speech recognition**

The objective of a maximum likelihood based training scheme is to globally maximise the likelihood of a set of arbitrary length utterances with respect to the model parameters. The natural levels into which the overall HMM based model for speech recognition between a sentence and

the acoustic realisation can be split are the word level, the phoneme level, the HMM or phone level and the state level.

A sentence is a particular sequence of words. Nevertheless the use of a more flexible and abstract notion of a sentence might be beneficial for some tasks. The simplest way to implement such flexibility is to define equivalent word sequences. For example the word sequence “he is” can be replaced by “he’s” without change of meaning. The variability in the transition from the word to the phoneme level is commonly used by state-of-the-art speech recognisers in the form of pronunciation variants. A phonemic transcription of the training data is usually required for training of HMMs and, if not available, must be determined beforehand. In practice the pronunciation variant for each word in each utterance is obtained by alignment of the training data using a multiple pronunciation dictionary and an existing HMM model set. Once the pronunciation is selected it is kept constant during the remaining training steps.

The HMM sequence is usually derived by using contextual information within the phoneme sequence to select a particular allophone model. Biphone, triphone and quinphone models are intended to model allophonic variation and only differ in the context width. Diphones model the transition between phonemes and can be understood as models which are defined by context only. For a limited vocabulary the number of context dependent phones may be relatively small. In the case of large vocabulary systems not all phoneme triplets will appear sufficiently often in the training data<sup>1</sup>. A sharing approach enables multiple phoneme combinations to share the same HMM or parts thereof. Deterministic rules are determined for the selection of a particular HMM by taking the particular data available into account. In practice sharing at the HMM level appears to yield poorer performance than sharing at the state level. Sharing on the state level implicitly makes the model topology constant over the set of HMMs for which states can be shared. If phonetic decision trees are used for divisive state clustering, the tree structure and the associated questions also provide a way to predict the appropriate model for unobserved phoneme combinations in the training data. In most cases only phonemic context is used to derive the particular phone model under the assumption that this provides sufficient information to identify statistically stationary segments.

The deterministic mapping reflects the desire to retain strictly separable stages. If the mapping from sentence to state sequence is assumed to be nondeterministic as well, a new overall model for speech signals has to be found and trained on the desired training corpus. Using the maximum likelihood objective function, the likelihood of a particular observation sequence  $\mathbf{O}$  given its associated sentence  $\mathbf{S}$ ,  $p(\mathbf{O}|\mathbf{S})$  can be expanded to

$$p(\mathbf{O}|\mathbf{S}, \lambda) = \sum_{\mathbf{W} \in \Omega(\mathbf{S})} \sum_{\mathbf{R} \in \Omega(\mathbf{W})} \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}, \mathbf{M}, \mathbf{R}, \mathbf{W}|\mathbf{S}, \lambda)$$

where  $\mathbf{W}$  denotes a word sequence,  $\mathbf{R}$  a phoneme sequence,  $\mathbf{M}$  a model sequence, and  $\mathbf{q}$  the state alignment sequence of the underlying HMM. The HMM model parameters are denoted by  $\lambda$  and the subspaces for each sequence spanned over the next lower level are denoted by  $\Omega(\mathbf{S})$ ,

<sup>1</sup>Some phoneme triplets may never appear at all.

$\Omega(\mathbf{W})$ ,  $\Omega(\mathbf{R})$  and  $\Omega(\mathbf{M})$ . Apart from  $\mathbf{O}$  all of these sequences are discrete in value and can have arbitrary length. In practice the length of all discrete sequences cannot exceed the length of the observation. In order to simplify this formulation the probability distributions for each level can be separated in a bottom-up fashion

$$\begin{aligned} p(\mathbf{O}, \mathbf{q}, \mathbf{M}, \mathbf{R}, \mathbf{W}|\mathbf{S}, \lambda) &= p(\mathbf{O}|\mathbf{q}, \mathbf{M}, \mathbf{R}, \mathbf{W}, \mathbf{S}, \lambda) \cdot \\ &P(\mathbf{q}|\mathbf{M}, \mathbf{R}, \mathbf{W}, \mathbf{S}, \lambda) \cdot \\ &P(\mathbf{M}|\mathbf{R}, \mathbf{W}, \mathbf{S}, \lambda) \cdot \\ &P(\mathbf{R}|\mathbf{W}, \mathbf{S}, \lambda) \cdot \\ &P(\mathbf{W}|\mathbf{S}, \lambda) \end{aligned}$$

With the assumption that each level depends only on the next upper level, the joint distribution can be formulated by

$$p(\mathbf{O}, \mathbf{q}, \mathbf{M}, \mathbf{R}, \mathbf{W}|\mathbf{S}, \lambda) = p(\mathbf{O}|\mathbf{q}, \lambda)P(\mathbf{q}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda)P(\mathbf{R}|\mathbf{W}, \lambda)P(\mathbf{W}|\mathbf{S}, \lambda) \quad (4.1)$$

which further simplifies the observation likelihood to

$$p(\mathbf{O}|\mathbf{S}, \lambda) = \sum_{\mathbf{W} \in \Omega(\mathbf{S})} P(\mathbf{W}|\mathbf{S}, \lambda) \sum_{\mathbf{R} \in \Omega(\mathbf{W})} P(\mathbf{R}|\mathbf{W}, \lambda) \sum_{\mathbf{M} \in \Omega(\mathbf{R})} P(\mathbf{M}|\mathbf{R}, \lambda) \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}|\mathbf{q}, \lambda)P(\mathbf{q}|\mathbf{M}, \lambda) \quad (4.2)$$

We define the optimisation level in training to be the lowest level in this hierarchy for which the associated sequence is known. Choosing an optimisation level removes the sums over all higher levels in the above formula whereas all lower levels are hidden. Standard HMM systems use the model level as the level of optimisation. In this case the hidden data is the actual state sequence. In order to optimise on this level it must be possible to derive the HMM model sequence  $\mathbf{M}$  from the sentence  $\mathbf{S}$  using a unique mapping function  $\mathbf{M} = f(\mathbf{S})$ :

$$p(\mathbf{O}|\mathbf{S}, \lambda) \simeq p(\mathbf{O}|\mathbf{M} = f(\mathbf{S}), \lambda) = \sum_{\mathbf{q} \in \Omega(\mathbf{M})} P(\mathbf{O}, \mathbf{q}|\mathbf{M}, \lambda) \quad (4.3)$$

However in the notation used in Equation 4.2 the model sequence, the phoneme sequence or word sequence can be modelled as hidden (i.e. unknown, given a sentence and an audio stream). The modelling of pronunciation variation by the inclusion of pronunciation variant probability estimates in this sense has been shown to be useful for example by (Cremelie and Martens, 1995; Hain and Woodland, 1999c; Peskin et al., 1999; Cremelie and Martens, 1999).

The use of one additional hidden layer will introduce a “soft” decision solution for the relations between dictionary entries and acoustic models. One advantage of pronunciation dictionaries is the possibility to simply expand the recognition dictionaries without the need for the associated

acoustic data to be available. In order to maintain this property a representation of words in terms of phonemes is desirable. The remainder of this thesis will assume that both sequences are known unless stated otherwise, the formulation of equations will only take model variations into account. Furthermore in certain scenarios, the use of dictionaries with only a single pronunciation variant will be used to make the link between a sentence and a phoneme sequence truly unique. The inverse of this mapping will never be unique due to the effect of homophones. However, in English the percentage of homophones in a dictionary is small and therefore only a small amount of confusion about the actual word is introduced by them.

#### 4.1.1 Maximum likelihood

The maximum likelihood criterion for parameter estimation was discussed in Section 2.3.4. Intuitively, we would like to set the model parameters  $\lambda$  such that they best agree with the observed training samples. Given a particular observation and model sequence pair  $(\mathbf{O}, \mathbf{M})$ , the ML criterion requires the selection of the parameter set,  $\lambda_{\text{mle}}$ , which achieves the highest likelihood:

$$\lambda_{\text{mle}} = \arg \max_{\lambda} p(\mathbf{O}|\mathbf{M}, \lambda)$$

However, a solution for obtaining the HMM parameters associated with the global maximum is not known. As shown by (Baum et al., 1970), a local maximum for the likelihood of a particular model sequence can be found by repeated maximisation of the so called auxiliary function  $Q(\lambda, \hat{\lambda})$ . This is a specific instance of the Expectation-Maximisation (E-M) algorithm (Dempster et al., 1977) which provides a general strategy for maximising likelihoods. The model under investigation in this Chapter is an extension of HMMs and similarly no algorithm for obtaining the globally optimal parameters of the extended model is known. However, the broader formulation of E-M can be used for the estimation of parameters of the complete model.

The auxiliary function as defined by Dempster is the conditional expected value of the log probability density function of the complete data  $x = (y, u)$  given the re-estimated model set  $\hat{\lambda}$ . The observed data  $y$  and the original model parameters  $\lambda$  are known:

$$\tilde{Q}(\hat{\lambda}, \lambda) = E\{\log(p(x|\hat{\lambda})) | y, \lambda\} \quad (4.4)$$

$u$  represents any unknown or hidden data. In the case of HMMs the complete data is  $x = (\mathbf{O}, \mathbf{q})$  and the observed data is  $y = \mathbf{O}$ . The equation can be simplified to

$$\tilde{Q}(\hat{\lambda}, \lambda) = E\{\log(p(\mathbf{O}, \mathbf{q}|\hat{\lambda})) | \mathbf{O}, \lambda\} = \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{q}|\mathbf{O}, \lambda) \log(p(\mathbf{O}, \mathbf{q}|\hat{\lambda}))$$

To obtain the solution provided by Baum this function needs to be multiplied by the distribution of observed data given the model  $p(\mathbf{O}|\lambda)$

$$Q(\hat{\lambda}, \lambda) = p(\mathbf{O}|\lambda)\tilde{Q}(\hat{\lambda}, \lambda) = \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}|\lambda) \log \left( p(\mathbf{O}, \mathbf{q}|\hat{\lambda}) \right) \quad (4.5)$$

(Baum et al., 1970) shows that for the case HMMs with discrete output distributions this function has a unique global maximum which can be found by setting the derivative to zero<sup>2</sup>. Since the actual HMM parameters have to satisfy “sum-to-one” constraints Lagrange multipliers are used for solving the equations.

Optimisation at the phoneme sequence level requires maximisation of

$$p(\mathbf{O}|\mathbf{R}, \lambda) = \sum_{\mathbf{M} \in \Omega(\mathbf{R})} p(\mathbf{O}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda) \quad (4.6)$$

where  $P(\mathbf{M}|\mathbf{R}, \lambda)$  describes the relationship between the model sequence  $\mathbf{M}$  and the sequence of phonemes  $\mathbf{R}$ .  $P(\mathbf{M}|\mathbf{R}, \lambda)$  is further called the model sequence model (MSM). A locally optimal solution for the extended set of model parameters  $\lambda$  can be found by stepwise maximisation of an extended auxiliary function. The auxiliary function can now be formulated by using Equation 4.4 and the pair  $(\mathbf{q}, \mathbf{M})$  as unknown data:

$$Q(\hat{\lambda}, \lambda) = \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}, \mathbf{M}|\mathbf{R}, \lambda) \log \left( p(\mathbf{O}, \mathbf{q}, \mathbf{M}|\mathbf{R}, \hat{\lambda}) \right)$$

Using the decomposition  $p(\mathbf{O}, \mathbf{q}, \mathbf{M}|\mathbf{R}, \lambda) = p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda)$  the expression can be expanded to

$$\begin{aligned} Q(\hat{\lambda}, \lambda) &= \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda) \log \left( p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \hat{\lambda}) \right) + \\ &+ \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda) \log \left( P(\mathbf{M}|\mathbf{R}, \hat{\lambda}) \right) \end{aligned}$$

The parameter set  $\lambda$  can be split into the parameters of the HMM set  $\phi$  and the parameters of the model sequence model  $\theta$ . If the parameter sets are independent the above formula can be written in the following form

$$\begin{aligned} Q(\hat{\lambda}, \lambda) &= \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \phi)P(\mathbf{M}|\mathbf{R}, \theta) \log \left( p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \hat{\phi}) \right) + \\ &+ \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q}|\mathbf{M}, \phi)P(\mathbf{M}|\mathbf{R}, \theta) \log \left( P(\mathbf{M}|\mathbf{R}, \hat{\theta}) \right) \\ &= \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \left( P(\mathbf{M}|\mathbf{R}, \theta)Q_{\text{HMM}}(\hat{\phi}, \phi|\mathbf{M}) \right) \\ &+ \sum_{\mathbf{M} \in \Omega(\mathbf{R})} \left( p(\mathbf{O}|\mathbf{M}, \phi)Q_{\text{MSM}}(\hat{\theta}, \theta|\mathbf{M}) \right) \end{aligned} \quad (4.7)$$

<sup>2</sup>A wider class of output distributions is covered by (Liporace, 1982).

where

$$Q_{\text{HMM}}(\hat{\phi}, \phi | \mathbf{M}) = \sum_{\mathbf{q} \in \Omega(\mathbf{M})} p(\mathbf{O}, \mathbf{q} | \mathbf{M}, \phi) \log \left( p(\mathbf{O}, \mathbf{q} | \mathbf{M}, \hat{\phi}) \right) \quad (4.8)$$

$$Q_{\text{MSM}}(\hat{\theta}, \theta | \mathbf{M}) = P(\mathbf{M} | \mathbf{R}, \theta) \log \left( P(\mathbf{M} | \mathbf{R}, \hat{\theta}) \right) \quad (4.9)$$

$Q_{\text{HMM}}(\cdot | \mathbf{M})$  denotes the HMM auxiliary function as outlined in Equation 4.5 under the condition that the model sequence is known. Similarly  $Q_{\text{MSM}}(\cdot | \mathbf{M})$  denotes the auxiliary function for re-estimation of the model sequence model only. Since the first part of Equation 4.7 only depends on the re-estimated HMM parameters and not on new MSM parameters and the converse holds for the second part, the two terms in the sum can be maximised independently. The E-M re-estimation steps are therefore

1. compute statistics  $P(\mathbf{M} | \mathbf{R}, \theta)$  and  $p(\mathbf{O} | \mathbf{M}, \phi)$
2. maximise  $\sum_{\mathbf{M} \in \Omega(\mathbf{R})} Q_{\text{HMM}}(\hat{\phi}, \phi | \mathbf{M}) P(\mathbf{M} | \mathbf{R}, \theta)$  with respect to  $\hat{\phi}$
3. maximise  $\sum_{\mathbf{M} \in \Omega(\mathbf{R})} Q_{\text{MSM}}(\hat{\theta}, \theta | \mathbf{M}) p(\mathbf{O} | \mathbf{M}, \phi)$  with respect to  $\hat{\theta}$
4. repeat from step 1 after assigning  $\hat{\theta} \rightarrow \theta$ ,  $\hat{\phi} \rightarrow \phi$

Convergence of this algorithm has to be proven depending on the structure of the MSM. Note that in principle an arbitrary alignment between phone and model sequence is possible. The only assumptions made so far are the decomposition capability of joint probability density functions into their hierarchical structure and the assumption of model parameter independence. The task of providing an estimate for  $P(\mathbf{O} | \mathbf{M}, \phi)$  for all possible pairs  $(\mathbf{O}, \mathbf{M})$  increases the complexity considerably compared to the standard HMM approach, especially since the model sequence may vary substantially in length and the number of symbols.

### 4.1.2 The Viterbi approximation

In order to alleviate the massive additional computational cost introduced by the optimisation of Equation 4.7 one standard approach is to approximate the sums over a set of probability density functions with the one which achieves the maximum value. This approximation style is called the Viterbi approximation and is used both for training of HMMs and for decoding (see Chapter 2).

When training standard HMMs the computational expense may be lowered by simplifying Equation 4.3. The major contribution to the sum over all state sequences is assumed to stem from the most likely state sequence  $\mathbf{q}^*$ :

$$p(\mathbf{O} | \mathbf{M}, \lambda) \simeq \max_{\mathbf{q} \in \Omega(\mathbf{M})} P(\mathbf{O}, \mathbf{q} | \mathbf{M}, \lambda) = P(\mathbf{O}, \mathbf{q}^* | \mathbf{M}, \lambda)$$

If we take a similar approach to HMS-HMMs, the sum over all possible model sequences in Equation 4.6 is simplified by taking the most likely model sequence

$$p(\mathbf{O}|\mathbf{R}, \lambda) \simeq \max_{\mathbf{M} \in \Omega(\mathbf{R})} p(\mathbf{O}|\mathbf{M}, \lambda)P(\mathbf{M}|\mathbf{R}, \lambda) = p(\mathbf{O}|\mathbf{M}^*, \lambda)P(\mathbf{M}^*|\mathbf{R}, \lambda)$$

Again since the HMM and MSM parameters are independent, the two terms can be maximised independently. Similarly an approximate form of the auxiliary function 4.7 can be obtained if the sum is replaced by its maximum:

$$Q(\hat{\lambda}, \lambda) \simeq Q_{\text{HMM}}(\hat{\phi}, \phi|\mathbf{M}^*)P(\mathbf{M}^*|\mathbf{R}, \theta) + Q_{\text{MSM}}(\hat{\theta}, \theta|\mathbf{M}^*)p(\mathbf{O}|\mathbf{M}^*, \phi) \quad (4.10)$$

The approximation of the auxiliary function is a weighted sum of the individual auxiliary functions  $Q_{\text{HMM}}(\hat{\phi}, \phi|\mathbf{M}^*)$  and  $Q_{\text{MSM}}(\hat{\theta}, \theta|\mathbf{M}^*)$  which correspond to HMM and MSM optimisation respectively. Due to parameter independence the functions remain constant if one set of parameters is modified. Furthermore the weighting factors  $P(\mathbf{M}^*|\mathbf{R}, \theta)$  and  $p(\mathbf{O}|\mathbf{M}^*, \phi)$  only depend on the model set  $\lambda = (\phi, \theta)$ . An optimisation of Equation 4.10 with respect to the new model parameters  $\hat{\lambda} = (\hat{\phi}; \hat{\theta})$  can ignore these factors completely. Given the best model sequence  $\mathbf{M}^*$  the search for optimal parameters of HMM and MSM can be carried out by independent optimisation of  $Q_{\text{HMM}}(\hat{\phi}, \phi|\mathbf{M}^*)$  and  $Q_{\text{MSM}}(\hat{\theta}, \theta|\mathbf{M}^*)$ .

The Viterbi approximation at the model sequence level in training and decoding is assumed to have only a minor impact on the performance reported in this thesis.

#### 4.1.2.1 Motivation and analysis

The Viterbi or maximum approximation is commonly used in automatic speech recognition. Its advantage lies primarily in the significantly reduced cost of computation. The maximum approximation can be used on different levels of probability computation and is most commonly used on the state level. However, the Viterbi approximation at the mixture component level was successfully applied both in training and decoding (e.g. (Ney et al., 1993)). Formally the approximation can only yield similar results to the correct estimate if the distributions over for example the log likelihood values of all possible state sequences have a clear peak at the maximum in the first place.

In practice, the many assumptions necessary for construction of an HMM based ASR system as outlined in Chapter 2 inevitably lead to the use of “incorrect” models for the underlying data. Since most of the assumptions made are expected to introduce confusability by broadening of distributions an approach which is based on hard decisions about the structure of the data may help to lower the complexity of decision boundaries. Furthermore recognition of speech in a large vocabulary framework is only feasible when using the Viterbi approximation. The use of the phoneme context, within words and across word boundaries and language models results in an enormous search space which is only tractable using the Viterbi approximation. In training, where computational cost is a less constraining factor the Viterbi approximation is still used by

many research sites. On the one hand the training procedure becomes symmetric to decoding. On the other hand if for example the state sequence is known<sup>3</sup> the temporal degree of freedom is momentarily fixed. Advantage can be taken of this fact for initialisation of mixture component output distributions by the use of clustering techniques such as  $k$ -means clustering.

All of the above would be of minor interest if a reasonable performance improvement could be expected by computing the full sums. Experimental evidence obtained on the Resource Management corpus (see Appendix A.1) however suggests otherwise. A 50 best list for the feb89 test set was generated using a state clustered HMM set and the default word-pair grammar. Figure 4.1(a) shows the difference between the log-likelihood estimates of the best path and the result when using the forward/backward (FB) algorithm for the best hypothesis in the 50-best list in terms of average per frame log-likelihood. As can be seen the difference between the estimates is rather small in relation to the per frame likelihood<sup>4</sup> and the difference appears to be within tight bounds for most utterances. Figure 4.1(b) shows the difference in Viterbi approximated average per frame log-likelihood between the first hypothesis and all others in a 50-best list averaged over all 300 sentences of the test set. Evidently apart from the next best hypothesis the difference between competing hypotheses is much larger than the difference shown in Figure 4.1(a).

A 50-best experiment on the Resource Management corpus using the 300 sentence feb89 test set was conducted to investigate any impact on the word error rate. Using the standard setup of crossword triphone models and a word-pair grammar changed the result for 2 sentences and gave a decrease in word error rate from 3.16% word error rate absolute to 3.08%.

In Section 4.2.1 a particular type of model sequence model is discussed which bears close structural resemblance to state tied HMM systems.

### 4.1.3 An information theoretic perspective

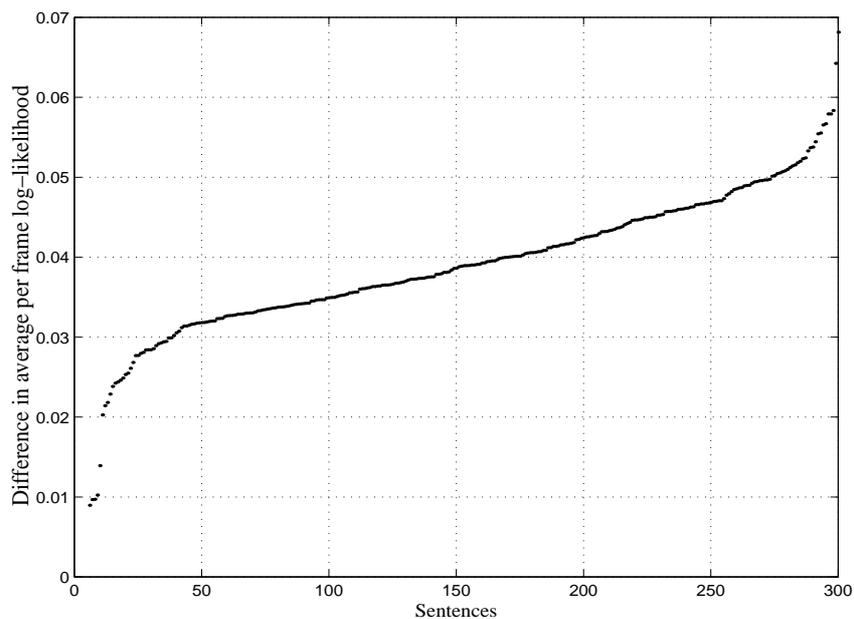
Adding an additional level to modelling introduces more flexibility and conversely more confusability. An increase in flexibility and thus the possibility for better modelling of the underlying data can be assessed qualitatively by an improvement in data likelihood. However, an increase in likelihood does not necessarily guarantee better performance in classification since it can appear as the result of increased between-class confusion. Another method to compare different modelling techniques is the comparison of system behaviour using information theoretic methods<sup>5</sup>. For the purpose of this analysis we assume that the observed feature stream contains sufficient information for the purpose of classification. This is likely to be the case in many speech recognition scenarios, where sufficient data is available. Consequently error free recognition must

---

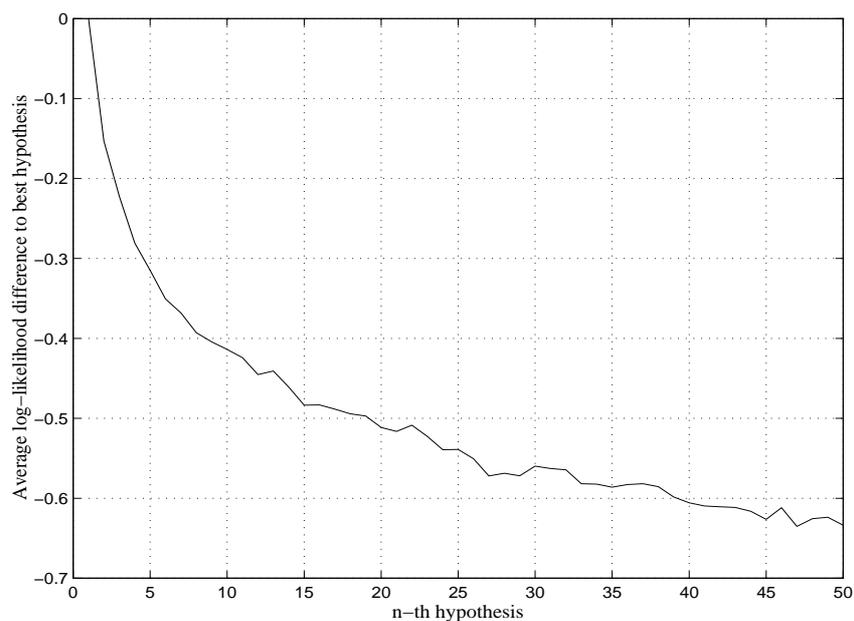
<sup>3</sup>The current state of the sentence HMM is known for each time instance  $t$ .

<sup>4</sup>The overall average per frame log-likelihood in the case of the FB algorithm was found to be -68.81 and in the case of Viterbi -68.84.

<sup>5</sup>For an introduction into information theory the reader is referred to the standard literature, for example (Cover and Thomas, 1991).



(a) Comparison of Viterbi and FB



(b) Comparison of N-best list entries

Figure 4.1 *Difference in average per frame log likelihood between forward backward computation and Viterbi approximation. (a) Values are obtained for each sentence of the Resource Management feb89 test set with a standard crossword triphone models. The values were obtained by alignment of the output of a Viterbi decoding pass using the standard word-pair grammar and have been sorted for illustrative purpose. (b) Difference in average log-likelihood within a 50-best list to the top hypothesis.*

be possible. If errors in recognition occur the cause must be unused or misrepresented information in the decision process. Whereas potential misrepresentation is evident in the design of front-ends, the loss of information in modelling is less clear. However any modelling is bound to induce a loss of information unless the model and the underlying modelling parameters are correct.

The object of this analysis is the investigation of the information transported between the observation sequence  $\mathbf{O}$  and the phoneme sequence  $\mathbf{R}$ . Information about  $\mathbf{O}$  that is irrelevant for a decision about the phoneme sequence will not be able to reduce the entropy  $H(\mathbf{R})$ . Consequently a maximisation of the mutual information is desired.

The observation, model and phoneme sequences are interpreted as random variables. The mutual information between the three random variables is defined as

$$I(\mathbf{O}; \mathbf{M}; \mathbf{R}) = \mathcal{E} \left\{ \log \frac{p(\mathbf{O}, \mathbf{M}, \mathbf{R})}{p(\mathbf{O})P(\mathbf{M})P(\mathbf{R})} \right\}$$

Using the hierarchical level assumption in Equation 4.1 the above can be modified to yield

$$\begin{aligned} I(\mathbf{O}; \mathbf{M}; \mathbf{R}) &= \mathcal{E} \left\{ \log \left( \frac{p(\mathbf{O}|\mathbf{M}, \mathbf{R})P(\mathbf{M}|\mathbf{R})P(\mathbf{R})}{p(\mathbf{O})P(\mathbf{M})P(\mathbf{R})} \right) \right\} \\ &= \mathcal{E} \left\{ \log \left( \frac{p(\mathbf{O}|\mathbf{M})}{p(\mathbf{O})} \right) + \log \left( \frac{P(\mathbf{M}|\mathbf{R})}{P(\mathbf{M})} \right) \right\} \\ &= I(\mathbf{O}; \mathbf{M}) + I(\mathbf{M}; \mathbf{R}) \end{aligned}$$

The amount of information exchanged between the three sequences is the sum of the information transferred between model and observation sequence and the information transferred between model and phoneme sequence alone. Using a similar operation

$$\begin{aligned} I(\mathbf{O}; \mathbf{M}; \mathbf{R}) &= \mathcal{E} \left\{ \log \frac{P(\mathbf{M}|\mathbf{O}, \mathbf{R})p(\mathbf{O}|\mathbf{R})}{P(\mathbf{M})p(\mathbf{O})} \right\} \\ &= I(\mathbf{O}; \mathbf{R}) + \mathcal{E} \left\{ \log \frac{P(\mathbf{M}|\mathbf{O}, \mathbf{R})}{P(\mathbf{M})} \right\} \\ &= I(\mathbf{O}; \mathbf{R}) + H(\mathbf{M}) - H(\mathbf{M}|\mathbf{O}, \mathbf{R}) \end{aligned}$$

the mutual information between observation and phoneme sequence,  $I(\mathbf{O}, \mathbf{R})$ , can be computed:

$$I(\mathbf{O}; \mathbf{R}) = I(\mathbf{O}; \mathbf{M}) + I(\mathbf{M}; \mathbf{R}) - H(\mathbf{M}) + H(\mathbf{M}|\mathbf{O}, \mathbf{R}) \quad (4.11)$$

Using the fact that knowledge about a random variable can only reduce the uncertainty about another event  $H(x|y) \leq H(x)$ :

$$I(\mathbf{M}; \mathbf{R}) - H(\mathbf{M}) + H(\mathbf{M}|\mathbf{O}, \mathbf{R}) = H(\mathbf{M}|\mathbf{O}, \mathbf{R}) - H(\mathbf{M}|\mathbf{R}) \leq 0 \quad (4.12)$$

Identity in this inequality only occurs if the information about the acoustic realisation has no influence on the selection of the appropriate model sequence. The total mutual information  $I(\mathbf{O}; \mathbf{R})$  must be smaller than the information exchanged between model and observation sequence alone:

$$I(\mathbf{O}; \mathbf{R}) \leq I(\mathbf{O}; \mathbf{M})$$

Even though this result is intuitive in the sense that in a hierarchical chain information can only be lost, its implication on modelling has to be investigated further. A model  $\phi$  with a fixed model sequence should be compared with a model  $\lambda$  with a variable model sequence. In general an estimate of the mutual information using a model is only identical to the true mutual information if model correctness and correct model parameters can be assumed. Since two different models are investigated we cannot assume model correctness for both of them. Practically neither the models nor the parameters can be assumed to be correct and thus for an arbitrary model  $\lambda$

$$I(\mathbf{O}; \mathbf{R}) \geq I_\lambda(\mathbf{O}; \mathbf{R}) = \int_{\mathbf{O}} d\mathbf{O} \sum_{\mathbf{R}} p(\mathbf{O}, \mathbf{R}) \log \left( \frac{p_\lambda(\mathbf{O}|\mathbf{R})}{p_\lambda(\mathbf{O})} \right)$$

If we first assume that the model sequence is a function of the phoneme sequence  $\mathbf{M} = f(\mathbf{R})$

$$I_\lambda(\mathbf{O}; \mathbf{R}) = I_\phi(\mathbf{O}; f(\mathbf{R}))$$

If the HMM parameters of the model  $\phi$  are assumed to be part of model set  $\lambda = (\phi, \theta)$  the difference in mutual information between a standard HMM and HMS-HMMs can be formulated:

$$I_\lambda(\mathbf{O}; \mathbf{R}) - I_\phi(\mathbf{O}; \mathbf{M}) = H_\lambda(\mathbf{M}|\mathbf{O}, \mathbf{R}) - H_\theta(\mathbf{M}|\mathbf{R})$$

which as stated before must be negative. However the optimal model parameters for an HMM are not necessarily the optimal parameters for an HMS-HMM. The mutual information between observation and phoneme sequence can be split into

$$I(\mathbf{O}; \mathbf{R}) = H(\mathbf{O}) - H(\mathbf{O}|\mathbf{R})$$

Whereas  $H(\mathbf{O})$  can be assumed to be constant for our purpose, the quantity  $H(\mathbf{O}|\mathbf{R})$  is clearly model dependent. An increase in the uncertainty about  $\mathbf{O}$  (as for example induced by using an incorrect model) under knowledge about the phoneme sequence inevitably decreases the mutual information. Thus a model needs to achieve minimal uncertainty  $H(\mathbf{O}|\mathbf{R})$ . Using Equation 4.11  $H(\mathbf{O}|\mathbf{R})$  can be computed

$$H(\mathbf{O}|\mathbf{R}) = H(\mathbf{O}|\mathbf{M}) + (H(\mathbf{M}|\mathbf{R}) - H(\mathbf{M}|\mathbf{O}, \mathbf{R})) \quad (4.13)$$

$H(\mathbf{O}|\mathbf{M})$  can be interpreted as the negative average log-likelihood of the data given that the model sequence is known<sup>6</sup>. In training the use of an maximum likelihood criterion minimises that entropy. The term in brackets in the above equation is the *a priori* uncertainty about  $\mathbf{M}$  reduced by the entropy which includes knowledge about  $\mathbf{O}$ . The term in brackets is positive and thus represents the balance to an increase in log-likelihood. Since  $H(\mathbf{O}|\mathbf{R})$  represents the total average log-likelihood a comparison of log-likelihoods gives an impression about an increase in mutual information.

In summary HMS-HMMs can increase the mutual information between observation and phoneme sequence if the gain in acoustic log-likelihood<sup>7</sup> is sufficient compared to the increase in uncertainty about the model sequence.

## 4.2 Model sequence modelling

A model sequence model maps an arbitrary phoneme sequence, obtained by translation of a word sequence using a particular dictionary, to an arbitrary model sequence. No particular modelling assumption has been made so far in definition and optimisation about the underlying model structure and the method for providing estimates for  $P(\mathbf{M}|\mathbf{R}, \lambda)$ . The required probability estimate does not put any constraint on the structure and interpretation of the sequences or the underlying model. In practice the use of phoneme sequences should be kept in its current form to retain the advantage of simple dictionary expandability. Acoustic realisations of speech exhibit a relative wide ranging of context dependency. An important factor not described by dictionaries is prosody. However the effect of prosody is assumed to be significantly reduced by front-end processing. A constrained span of dependence of a particular model at a particular time can be expected. Nevertheless coarticulation and more sophisticated reduction or insertion effects can potentially require a mapping from a certain sequence of phonemes to a certain sequence of models. Model sequence models are required to provide a stochastic mapping between arbitrarily sized strings of discrete symbols which stem from independent sets.

The important difference to standard HMMs is the variability in the selection of the appropriate HMMs for a particular phoneme sequence. Standard HMMs constrain the selection of a particular model not only in terms of the mapping but equally in the arrangement of these mappings. For each phoneme in the phoneme sequence one set of rules is used to select exactly one model. A solution for alleviating the constraint in the arrangement of the individual mappings in a maximum likelihood fashion was used successfully by (Ostendorf and Singer, 1997). A similar interpretation can be used for model sequence models. MSMs can in theory alleviate both constraints and thus can be interpreted as having two degrees of freedom:

<sup>6</sup>Note that the average is taken over all possible model sequences.

<sup>7</sup>Acoustic means the match between model sequence and acoustic observation vectors as performed by HMMs.

### 1. Model distribution

Depending on a specific symbol cluster configuration, a hard decision for a single distribution may not be desirable. Acoustic effects of different temporal span may have an impact on the selection of an appropriate model. The set of HMMs may be of arbitrary size. The models themselves however lose the association with a particular phoneme context.

### 2. Sequence symbol clustering

The phoneme sequence and model sequence might have different lengths according to insertion and deletion effects in spontaneous speech. Again these effects may be temporally constrained and thus a local modification of the sentence HMM structure may be necessary. An important question is the definition and search for appropriate strings of symbols in both sequences.

The question about the appropriate set of models is strongly connected to the search for an appropriate HMM topology. Another way of interpretation of the hidden model sequence framework is a constrained maximum likelihood framework for HMM topology optimisation. MSMs in this case act as probability driven generators for context dependent HMMs. A more detailed discussion of this interpretation will be given in the subsequent sections. Another interpretation of the HMS-HMM framework especially in context of the Viterbi approximation is the dynamic selection of locally appropriate models. The change from the standard or most frequent model in this phonetic context to another one may be triggered simply by acoustic distortion or by speaker or accent variation. Finally HMS-HMM can be interpreted as a very flexible parameter tying or sharing scheme. Particularly the family of realisations presented in Section 4.2.1 bear strong relations with the method of soft-tying (Luo and Jelinek, 1999) which also has been used for modelling ambiguity across phones in (Saraçlar and Khudanpur, 2000; Saraçlar et al., 2000; Saraçlar, 2000).

MSMs allow very flexible modelling of the relationship between phonemes and acoustic models. But increased flexibility also introduces more complexity and confusability both in training and decoding. In order to investigate the appropriate scale of flexibility, an attempt is made to investigate the two degrees of freedom independently. This is done by first investigating the case of fixed alignment as detailed in the following section. This scenario is in many ways comparable to standard HMMs and can thus be suitably analysed. The lessons learned in this stage can then be used to investigate the case of flexible alignment. Obviously the number of potential modelling approaches is vast. Due to data and complexity constraints two methods of topology generating mappings have been investigated.

#### 4.2.1 Fixed alignment

In a formal sense model sequence models simply provide an estimate for the probabilities of all possible model sequences given a phoneme sequence  $P(\mathbf{M}|\mathbf{R})$ . The formulae derived in Section 4.1 are valid for arbitrary length model and phoneme sequences. In practice this flexibility

needs to be decreased in order to make the estimation of probabilities for model sequence feasible. First it is evident that for the decision about the most appropriate HMM only a limited range of models and phonemes which can potentially contribute to this decision are available. Secondly the desire to limit confusability demands that the number of potential model sequences in principle is kept low, regardless of the model structure. Another important issue is the trainability of the underlying model parameters. A more flexible mapping will inevitably require more model parameters and thus implicitly require more training data. As will be discussed in detail in Chapter 5, data sparsity is a major problem for construction of MSMs.

In order to fulfill these requirements an obvious approach is to take the concept of a local decision to its extreme. This can be done by the assumption that for each model in a particular sequence there exists a symbol in the phoneme sequence which is solely responsible for selection of that model. In other words this means that the model sequence and the phoneme sequence have equal length.

The underlying constraints can be shown by application of the chain rule to the estimate of the probability of model sequence  $\mathbf{M}$  given a phoneme sequence  $\mathbf{R}$ , both of length  $L_R$ :

$$P(\mathbf{M}|\mathbf{R}) = \prod_{n=1}^{L_R} P(m_n|m_{n-1}, \dots, m_1, \mathbf{R})$$

The complete model history and the complete phoneme sequence may have an influence on the  $n^{\text{th}}$  model selected. The direct computation of a reasonable estimate for the above probability with finite amounts of data is impossible. Furthermore the appropriate model for example within a word at the beginning of an utterance is unlikely to be affected by a word at the end of the sentence, or even the next word<sup>8</sup>. Since the sequences are aligned by definition, for each model  $m_n$  an associated phoneme is known and the dependence on the complete phoneme sequence can be constrained to dependence on that phoneme

$$P(m_n|m_{n-1}, \dots, m_1, \mathbf{R}) = P(m_n|m_{n-1}, \dots, m_1, r_n) \quad (4.14)$$

Furthermore the dependence on the previous models is dropped:

$$P(m_n|m_{n-1}, \dots, m_1, r_n) = P(m_n|r_n)$$

which leaves a simple formula for estimation of the probability of a model sequence:

$$P_{\text{mono}}(\mathbf{M}|\mathbf{R}) = \prod_{n=1}^{L_R} P(m_n|r_n) \quad (4.15)$$

---

<sup>8</sup>The author is aware of speech effects which naturally span across whole sentences or at least multiple words. However for the purpose of modelling it is assumed that proper front-ends eliminate most of these dependencies.

Each phoneme in the phoneme sequence can produce models with a certain probability which will be derived from the training data. Figure 4.2 illustrates the corresponding model network which implicitly is generated by the realisation of Equation 4.15. Whereas the number of potential models may be different for each phone each model is linked with each model associated with pre- and succeeding phonemes. For example, probabilities of the distribution associated with phoneme  $b$  in Figure 4.2,  $P(m_n|b)$ , may be represented by the arcs leading to model  $m_n$ . Consequently a probability of zero eliminates the arc from the graph. From the figure it is clear that this is similar in structure to commonly used networks for pronunciation modelling. The emerging pronunciation network itself is static, however the selection of a path through the network dynamically selects the most appropriate model depending on the present acoustic condition.

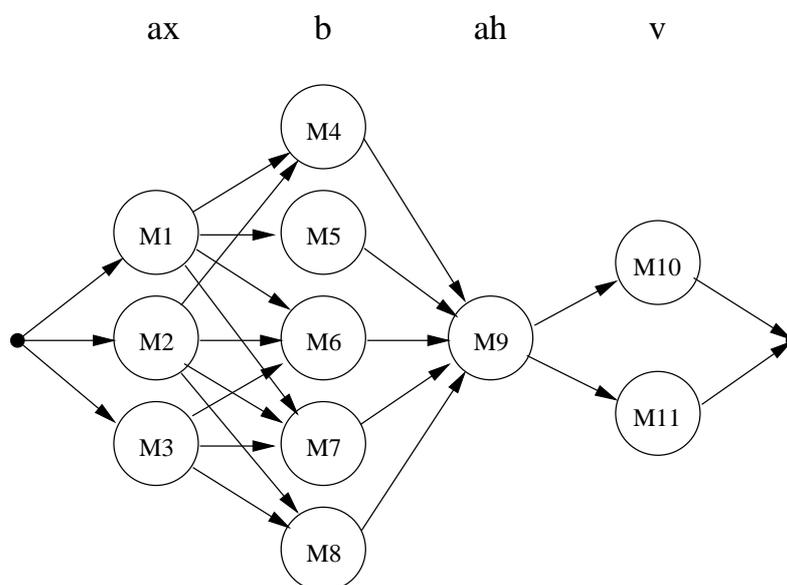


Figure 4.2 Fixed alignment model network topology using different hidden Markov models represented by the nodes of the network. The arcs in the network can be interpreted to hold the probability of the subsequent model given the phoneme. A missing arc denotes a probability of zero.

Estimation of the set of probabilities  $P(m_n|r_n)$  is straightforward. The models  $m_n$  are chosen from a finite set and no natural metric is known to put a set of arbitrary structured hidden Markov models in a meaningful order. Thus the obvious solution is to use a non-parametric representation of the discrete probability distribution. This has the disadvantage that for each potential model one parameter is necessary. A large number of models requires a large number of parameters to represent the distribution. In consequence sufficient training data needs to be provided for training of the large number of parameters. The problem of data sparsity becomes obvious when the number of parameters is put in the context of the potential model rate, i.e. the average number of models per second of speech. For the moment this can be assumed to be identical to the phoneme rate. On spontaneous speech data the average number of phonemes in one hour of speech was found to be approximately 35000<sup>9</sup>. Sufficient training data and proper

<sup>9</sup>This number excludes “silence phonemes”.

smoothing techniques will be necessary to provide useful probability estimates.

As outlined in Section 4.1.1 the maximisation of the overall training data likelihood using E-M under the Viterbi assumption at model sequence level requires the determination of the best model sequence  $\mathbf{M}^*$ . With this sequence known, the independent optimisation of the individual auxiliary functions for the MSM (Equation 4.9) and HMM (Equation 4.8) parameters is possible. The combination of Equation 4.9 and Equation 4.15 yields

$$Q_{\text{MSM}}(\hat{\theta}, \theta | \mathbf{M}^*) = \prod_{n=1}^{L_R} P(m_n^* | r_n) \left( \sum_{l=1}^{L_R} \log \hat{P}(m_l^* | r_l) \right)$$

$P(\mu | \rho)$  is one specific parameter from model set  $\theta$  and  $\hat{P}(\mu | \rho)$  the corresponding parameter in the re-estimated model set  $\hat{\theta}$ . Using Lagrange multipliers and taking derivatives with respect to the new parameters yields the N-gram estimate for arbitrary phoneme  $\rho$  and model  $\mu$ :

$$\hat{P}(\mu | \rho) = \frac{N(\mu, \rho)}{N(\cdot, \rho)}$$

where  $N(\cdot, \rho)$  is the phoneme frequency, whereas  $N(\mu, \rho)$  is the frequency of the joint occurrence of  $\mu$  and  $\rho$ .

Note that in this case both sequences are required to have the same length and that the probability of a model only depends on one phoneme. If this is compared with a standard HMM the model most similar to the one just described is a so called monophone model (see Chapter 2). In that case each phoneme in the dictionary has one model associated with it, the number of phonemes is identical to the number of HMMs. In the case of HMS-HMMs the number of phonemes and the number of models are independent. The number of possible models as well as the models themselves may be different for each phoneme. This however represents a much more powerful model than a standard monophone model. If the number of HMM parameters is assumed to be identical to the monophone case models the HMS-HMM converges to a standard HMM if models are not *shared* between phonemes. If sharing is allowed the HMS-HMM can be interpreted as a soft version of monophone models. The overall number of parameters in the system is increased by the parameters in the MSM. Even though model distributions are represented in non-parametric form the actual number of parameters is small compared to the number of parameters used for the underlying HMMs<sup>10</sup>.

For more demanding tasks monophone HMM sets are considerably less powerful than context dependent models such as biphone or triphone models. It is evident that the assumption made in Equation 4.14 can be extended to arbitrary phoneme context. The MSM represented by the following equation can be interpreted as the HMS-HMM equivalent to an unclustered triphone HMM model set:

<sup>10</sup>The actual number of parameters of a non-parametric N-gram based model will be defined by the number of events in the distribution with non-zero probability since the probability of the remaining events can be derived.

$$P_{\text{tri}}(\mathbf{M}|\mathbf{R}) = \prod_{n=1}^{L_R} P(m_n|r_{n-1}, r_n, r_{n+1}) \quad (4.16)$$

where the triplet  $(r_{n-1}, r_n, r_{n+1})$  is the *tri-phoneme context* of model  $m_n$ . The underlying strategy of model sharing is described by the above equation. In order to relate HMS-HMM to extended tying schemes such as clustered or generalised triphones (Lee, 1990) the precise relationship to models needs to be defined. Furthermore inevitably the problem of contexts with low frequency will require smoothing of the estimated probability distributions. Even more so appropriate schemes for generalisation to tri-phonemes which did not appear in the training data at all, but are necessary for recognition, need to be found.

The formulation so far has assumed that models are intended to cover the duration of a complete phoneme. If the models are to instead represent sub-phonemic units while the phoneme sequence level is retained, an obvious solution is to implement a fixed number of *positions* within in a particular phoneme. The number of positions can even be different for each phoneme context. In practice the separation of phone HMMs into beginning, middle and ending parts to reflect the transitional and stationary parts of a phone has lead to the use of 3 state HMMs. Similarly the assumption of 3 positions within each phoneme will exactly triple the length of a model sequence:

$$P_{\text{tri}}(\mathbf{M}|\mathbf{R}) = \prod_{n=1}^{L_R} \prod_{l=1}^3 P(m_{nl}|l, r_{n-1}, r_n, r_{n+1}) \quad (4.17)$$

This formulation is the HMS-HMM equivalent to state level clustered triphone HMMs (Young et al., 1994).

### 4.2.2 Variable alignment

The transcription of spontaneous speech is one of the most difficult tasks for automatic speech recognition. Among many effects, one of the most apparent changes are the pronunciations which are altered notably compared to those used for read speech (Greenberg, 1998). The effect is amplified when, as in most cases, the comparison is made between read speech produced in a well controlled environment such as a recording studio and spontaneous speech from free conversational situations. Since pronunciation variants are by definition discrete events, any change can be characterised by substitutions, insertions and deletions. Examination of the recognition dictionary used in the Cambridge University HTK system for the transcription of conversational telephone speech (Hain et al., 2000) shows that pronunciation variants, which differ only by phoneme substitution, make up 59.9% of all additional variants. Thus substitutions can be expected to be the dominant effect in pronunciation variation. The next most important effect present in spontaneous speech are deletions. Whereas insertions are expected to be least important, accent effects may appear as insertions relative to the canonical pronunciation.

In all of these cases it is not clear that any of these effects necessarily takes place on the level of complete phones. The assumption of asynchronous phonological feature streams has been of recent interest in automatic speech recognition (e.g. (Deng, 1997a; Ostendorf, 1999; Richardson et al., 2000)). In this paradigm the underlying process of speech production is assumed to be driven by articulatory features, which may change asynchronously. If the feature values are assumed to be discrete one particular combination of features defines a steady state in which the statistical properties are assumed to be constant. This in turn allows the use of HMM states for modelling a combination of features. If the resulting overall HMM topology is mapped back into a phone model structure a different HMM topology and a modification in the tying of parameters can be expected. Since in spontaneous speech some feature appearances may be rather shorter than normal, the number of states required to model the data may vary (insertions and deletions). Consequently the paradigm of asynchronous discrete features implies sub-phonetic variation in spontaneous speech.

In order to model insertions and deletions, the alignment between the model and the phone sequence needs to be flexible. The alignment associates symbol strings from both the model and phoneme sequences. In the case of variable alignment, given a pair of those sequences it is impossible to derive with absolute certainty which cluster of phonemes belongs to which cluster of models. Instead, a decoding procedure using the underlying MSM and its parameters is required to yield the desired result. This leaves the question as to how an appropriate MSM could be defined. The approach outlined in Section 4.2.1 has a natural extension in so called multigrams (Deligne and Bimbot, 1997b).

#### 4.2.2.1 Multigrams

Multigrams are a general framework for the alignment of arbitrary length sequences and have been used in language modelling (Bimbot et al., 1995; Deligne and Bimbot, 1995), for generation of phonemic transcriptions of words (Deligne et al., 1995), for inference of new acoustic units (Deligne and Bimbot, 1997a), or for spectrum representation and speech coding (Cernocky et al., 1998; Cernocky et al., 1997). In its most general form the theory of multigrams allows the segmentation of one or more sequences by assuming that a hidden variable describing the segmentation exists. Whereas in the case of language modelling only one sequence is segmented, in the other cases modelling required the concurrent segmentation of two sequences. The training procedure allows for the optimisation of the joint probability of the two sequences involved. If  $\mathbf{X}$  and  $\mathbf{Y}$  are the two sequences involved, the objective is to maximise  $P(\mathbf{XY})$  on the training data. In order to constrain the probability estimate to using only local dependencies, the application of the chain rule would be desirable. However the fact that the sequences have arbitrary length and no clearly associated symbols makes this separation difficult. The problem can be solved by a summation over all possible segmentations of each sequence under the constraint that the length of the segmentation sequences are identical:

$$P(\mathbf{X}, \mathbf{Y}) = \sum_{\mathbf{K}} \sum_{\mathbf{L}} P(\mathbf{X}, \mathbf{K}, \mathbf{Y}, \mathbf{L})$$

where  $\mathbf{K}$  is the segmentation sequence for  $\mathbf{X}$  and  $\mathbf{L}$  is the segmentation sequence for  $\mathbf{Y}$ <sup>11</sup>. Each of the segmentation sequences may be thought of as a sequence of indices  $k_n$  or  $l_n$  which mark the end of a symbol in the corresponding sequence. Thus the  $n^{\text{th}}$  subsequence or *string* of  $\mathbf{X}$  is given by  $\mathbf{X}_{k_{n-1}+1}^{k_n}$ . Since the segmentation sequences have equal length the element pairs  $(k_n, l_n)$  form the co-segmentation sequence. The knowledge of the segmentation allows the ordering of related subsequences or strings in both symbol sequences:

$$P(\mathbf{X}, \mathbf{Y}, \mathbf{K}, \mathbf{L}) = \prod_{n=1}^{L_K} P(\mathbf{X}_{k_{n-1}+1}^{k_n}, \mathbf{Y}_{l_{n-1}+1}^{l_n} | \mathbf{X}_1^{k_{n-1}}, \mathbf{Y}_1^{l_{n-1}})$$

where  $L_K$  denotes the length of the co-segmentation sequence. It becomes evident that an obvious assumption is the independence of the  $n^{\text{th}}$  strings from the entire history, giving

$$P(\mathbf{X}, \mathbf{Y}, \mathbf{K}, \mathbf{L}) = \prod_{n=1}^{L_K} P(\mathbf{X}_{k_{n-1}+1}^{k_n}, \mathbf{Y}_{l_{n-1}+1}^{l_n})$$

The above equation requires an estimate of the joint probability of two strings. It is important to note that during optimisation only a few segmentations of the two sequences will make significant contributions to the overall likelihood. This however implies that the underlying model will have close to zero probability for many possible co-occurring strings. In this sense the procedure reduces the set of all possible strings to those which allow the likelihood of the underlying probabilistic model to be maximised.

#### 4.2.2.2 Multigram based model sequence models

As outlined in the previous section the original multigram framework (Deligne and Bimbot, 1997b) was designed to use the joint probability as objective function in optimisation. MSMs however only require an estimate for  $P(\mathbf{M}|\mathbf{R})$ . This has the implication that if the multigram framework is used for MSMs only, model strings of arbitrary length are allowed. This represents the case of a 1:M mapping between phonemes and models. However, the clustering of phonemes into strings is less desirable than the clustering of models. Clustering of phonemes would enable more explicit models and implicitly use an increased context range. However, an extension of the context range can be obtained in a similar fashion to the fixed alignment case (see Section 4.2.1). In practice, the question of data sparsity will clearly limit the number of potential strings in this sequence. Thus in the remainder of this thesis, only the model sequence is subject to resegmentation.

Training of multigram MSMs requires the optimisation of the model sequence likelihood which can be computed by

<sup>11</sup>The sum  $\sum_{\mathbf{K}}$  is an abbreviation for the summation over all possible sequences  $\mathbf{K}$ .

$$P(\mathbf{M}|\mathbf{R}) = \sum_{\mathbf{L} \in \mathcal{S}(\mathbf{M})} P(\mathbf{M}, \mathbf{L}|\mathbf{R}) \quad (4.18)$$

where the hidden parameter  $\mathbf{L}$  represents a particular segmentation of the model sequence  $\mathbf{M}$  in the style described in Section 4.2.2.1 from the set of all possible segmentations of that sequence  $\mathcal{S}(\mathbf{M})$ . It is important to note that while the model sequence may have a different length to the phoneme sequence  $L_M \neq L_R$ , in the case of a 1:M mapping the length of the segmentation sequence is required to be equal to  $L_R$ .

In a similar style to that of standard multigrams, the joint probability of the model and segmentation sequences can be simplified by the assumption of independence. In order to simplify the notation we define  $\mathbf{r}_n$  as an phoneme context at the  $n^{\text{th}}$  position with arbitrary span in both directions. In the case of a formulation equivalent to a triphone HMM this expands to  $\mathbf{r}_n = (r_{n-1}, r_n, r_{n+1})$ . Using this notation the modelling assumption is given by the following formula

$$P(\mathbf{M}, \mathbf{L}|\mathbf{R}) = \prod_{n=1}^{L_R} P(\mathbf{M}_{l_{n-1}+1}^{l_n} | \mathbf{r}_n) \quad (4.19)$$

whereby  $\mathbf{M}_{l_{n-1}+1}^{l_n}$  is a model substring of length  $l_n - l_{n-1}$ . It becomes clear that if  $\mu$  is a particular model string and  $\rho$  a specific context the set of probabilities  $P(\mu|\rho)$  for any possible combination  $(\mu, \rho)$  constitutes the parameters of the multigram-MSM. The global maximisation of Equation 4.18 with respect to the model parameters does not have a closed form solution. Thus the E-M algorithm is used to search for suitable parameter estimates, but note that the solution found by this method may only be a local maximum. Using Equation 4.4 and similar operations as those applied in Section 4.1.1, the auxiliary function for multigram-MSMs can be defined to be

$$Q_{\text{mgram}}(\hat{\theta}, \theta) = \sum_{\mathbf{L} \in \mathcal{S}(\mathbf{M})} P(\mathbf{M}, \mathbf{L}|\mathbf{R}, \theta) \log \left( P(\mathbf{M}, \mathbf{L}|\mathbf{R}, \hat{\theta}) \right)$$

Using Equation 4.19 the auxiliary function can be modified

$$\begin{aligned} Q_{\text{mgram}}(\hat{\theta}, \theta) &= \sum_{n=1}^{L_R} \sum_{\mathbf{L} \in \mathcal{S}(\mathbf{M})} P(\mathbf{M}, \mathbf{L}|\mathbf{R}, \theta) \log \left( \hat{P}(\mathbf{M}_{l_{n-1}+1}^{l_n} | \mathbf{r}_n) \right) \\ &= \sum_{n=1}^{L_R} \sum_{\mu \in \mathcal{M}} \sum_{\rho \in \mathcal{R}} \sum_{\mathbf{L} \in \mathcal{L}(n, \mu, \rho)} P(\mathbf{M}, \mathbf{L}|\mathbf{R}, \theta) \log \left( \hat{P}(\mu|\rho) \right) \end{aligned}$$

where  $\mathcal{L}(n, \mu, \rho) = \{\mathbf{L} : \mathbf{M}_{l_{n-1}+1}^{l_n} = \mu \text{ and } \mathbf{r}_n = \rho\}$  is the set of all possible segmentations with the model string  $\mu$  and context  $\rho$  at time instance  $n$ .  $\mathcal{R}$  denotes the set of all possible contexts and  $\mathcal{M}$  denotes the set of all possible model strings. Standard optimisation using Lagrange multipliers with the boundary constraints

$$\sum_{\mu \in \mathcal{M}} P(\mu|\rho) = 1 \quad \forall \rho \in \mathcal{R}$$

yields the solution for the multigram-MSM parameters:

$$\hat{P}(\mu|\rho) = \frac{\sum_{n=1}^{L_R} \sum_{\mathbf{L} \in \mathcal{L}(n, \mu, \rho)} P(\mathbf{M}, \mathbf{L} | \mathbf{R}, \theta)}{\sum_{\mu \in \mathcal{M}} \sum_{n=1}^{L_R} \sum_{\mathbf{L} \in \mathcal{L}(n, \mu, \rho)} P(\mathbf{M}, \mathbf{L} | \mathbf{R}, \theta)} \quad (4.20)$$

$$= \frac{\sum_{\mathbf{L} \in \mathcal{S}(\mathbf{M})} P(\mathbf{L}, \mathbf{M} | \mathbf{R}, \theta) N(\mu, \rho)}{\sum_{\mathbf{L} \in \mathcal{S}(\mathbf{M})} P(\mathbf{L}, \mathbf{M} | \mathbf{R}, \theta) N(\cdot, \rho)} \quad (4.21)$$

where  $N(\mu, \rho)$  denotes the frequency of the pair  $(\mu, \rho)$  in the sequence triplet  $(\mathbf{L}, \mathbf{M}, \mathbf{R})$  and  $N(\cdot, \rho) = \sum_{\mu \in \mathcal{M}} N(\mu, \rho)$ . The next task is to find a simple computational solution to compute  $P(\mathbf{L}, \mathbf{M} | \mathbf{R}, \theta)$ . This can be achieved by a forward-backward procedure similar in style to that used in Baum-Welch re-estimation (see Section 2.3.2).

Define the forward and backward variables as follows

$$\alpha(\eta, n) = P(\mathbf{M}_1^\eta | \mathbf{R}_1^n) \quad (4.22)$$

$$\beta(\eta, n) = P(\mathbf{M}_{\eta+1}^{L_M} | \mathbf{R}_{n+1}^{L_R}) \quad (4.23)$$

where  $\eta$  and  $n$  are time indices. The forward variable  $\alpha(\eta, n)$  is the probability of the model and phoneme sequence co-occurring up to the indices  $\eta$  and  $n$  respectively. Note that the value of the forward variable will be zero for many specific pairs  $(\eta, n)$  since they represent impossible segmentations. The estimate for the probability of the complete model sequence can always be obtained by computation of the forward variable at the end of the sequences  $\alpha(L_M, L_R)$ . The desire is to provide a recursive solution for computation of the forward variable. Since  $\mathbf{M}_1^\eta$  is necessarily composed of strings, the last string in the sequence can be separated from the rest. In the final stage a string  $\mu$  of length  $k$  was appended  $\mathbf{M}_1^\eta = [\mathbf{M}_1^{\eta-k}, \mu]$ . It is important to note that  $\mu$  is solely determined by the length  $k$  since the model sequence  $\mathbf{M}$  is given. When a step in the model sequence is taken, a step to context  $\rho$  in the phoneme sequence is also taken simultaneously. The probability for just the step obviously is  $P(\mu|\rho)$ . In order to take all possible preceding model sequences into account it is necessary to sum over all possible  $k$  up to the maximum string length  $K$ :

$$\alpha(\eta, n) = \sum_{k=1}^K \alpha(\eta - k, n - 1) P(\mathbf{M}_{\eta-k+1}^\eta | \mathbf{r}_n) \quad (4.24)$$

The above formula denotes the forward recursion. The backward recursion can be derived in a similar fashion:

$$\beta(\eta, n) = \sum_{k=1}^K \beta(\eta + k, n + 1) P(\mathbf{M}_{\eta+1}^{\eta+k} | \mathbf{r}_{n+1})$$

Note that the computation of the forward/backward variables is independent. The computation of Equation 4.20 is based on an efficient computation of the sum of  $P(\mathbf{M}, \mathbf{L} | \mathbf{R})$  over all segmentations which yield a specific pair  $(\mu, \rho)$ . The following assumes that  $\mu$  has length  $k$  and  $\delta_{n,\eta}^{\mu,\rho}$  denotes the indicator function for a the model string/context pair at time  $\eta$  and  $n$  respectively:

$$\delta_{n,\eta}^{\mu,\rho} = \begin{cases} 1 & \mathbf{M}_{\eta-k+1}^{\eta} = \mu \quad \text{and} \quad \mathbf{r}_n = \rho \\ 0 & \text{otherwise} \end{cases}$$

This allows the numerator expression in Equation 4.20 to be reformulated as

$$\begin{aligned} \sum_{n=1}^{L_R} \sum_{\mathbf{L} \in \mathcal{L}(n, \mu, \rho)} P(\mathbf{ML} | \mathbf{R}, \theta) &= \sum_{n=1}^{L_R} \sum_{\mathbf{L} \in \mathcal{L}(n, \mu, \rho)} P(\mathbf{M}_1^{l_{n-1}}, \mathbf{L}_1^{n-1} | \mathbf{R}_1^{n-1}) P(\mu | \rho) P(\mathbf{M}_{l_{n+1}}^{L_M}, \mathbf{L}_{n+1}^{L_R} | \mathbf{R}_{n+1}^{L_R}) \\ &= \sum_{n=1}^{L_R} \sum_{\eta=1}^{L_M} P(\mathbf{M}_1^{\eta-k}, \mathbf{L}_1^{n-1} | \mathbf{R}_1^{n-1}) \delta_{n,\eta}^{\mu,\rho} P(\mu | \rho) P(\mathbf{M}_{\eta+1}^{L_M}, \mathbf{L}_{n+1}^{L_R} | \mathbf{R}_{n+1}^{L_R}) \\ &= P(\mu | \rho) \sum_{n=1}^{L_R} \sum_{\eta=1}^{L_M} \alpha(\eta - k, n - 1) \delta_{n,\eta}^{\mu,\rho} \beta(\eta, n) \end{aligned}$$

in terms of the forward/backward variables. The denominator of Equation 4.20 can be derived from the numerator by summation over all possible model strings. Since this eliminates a direct dependency on  $\rho$  the indicator function in this case can be reduced to

$$\delta_n^\rho = \begin{cases} 1 & \text{if } \mathbf{r}_n = \rho \\ 0 & \text{otherwise} \end{cases}$$

Thus all parameters of a multigram-MSM can be computed if the forward/backward variables are known:

$$\hat{P}(\mu | \rho) = P(\mu | \rho) \frac{\sum_{n=1}^{L_R} \sum_{\eta=1}^{L_M} \alpha(\eta - k, n - 1) \beta(\eta, n) \delta_{n,\eta}^{\mu,\rho}}{\sum_{n=1}^{L_R} \sum_{\eta=1}^{L_M} \alpha(\eta - k, n - 1) \beta(\eta, n) \delta_n^\rho} \quad (4.25)$$

Note that  $k$  denotes the length of the model string  $\mu$ . Figure 4.3 shows an example of a model topology with a small number of potential model strings for the word “grid” based on a multigram MSM. The dotted arcs only hold the probabilities for the particular model string chosen. The figure clearly shows the capability to model insertions, however the capability to model deletions is less clear. Multigram models cannot attach a probability to an event not occurring even though the formulae do not explicitly exclude this case (see Equation 4.24) . However a model of length zero is possible at any time instance which leads to an highly overestimated

probability. In order to model sub-phone deletions, the original HMMs need to be split such that for example two models per phoneme context is the standard case. This should not pose a problem in most cases since the standard HMM topology is a simple chain of states.

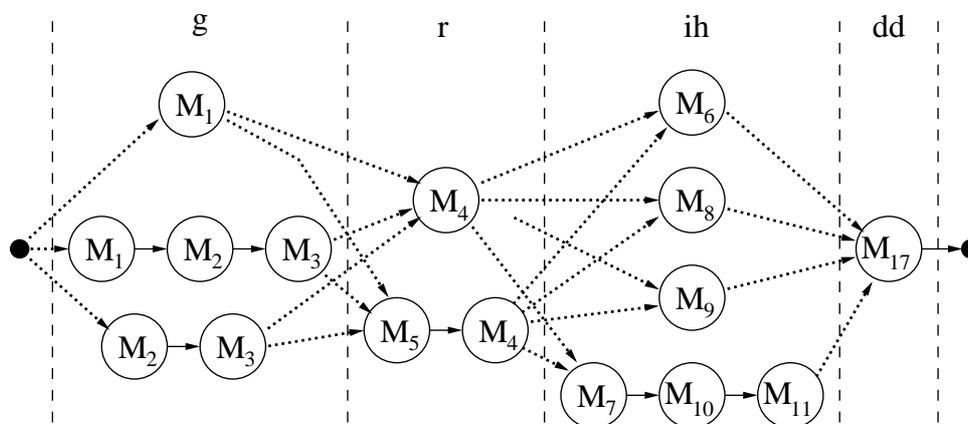


Figure 4.3 Simple topology based on a multigram-MSM showing insertions and deletions. The nodes represent HMMs which may have arbitrary topology.

An algorithmic implementation of the above re-estimation formulae is straightforward using the following procedure:

### 1. Accumulation

For each pair of model and phoneme sequences

- (a) Allocate a matrix to store the forward/backward values
- (b) Compute  $\alpha(\cdot, \cdot)$  and  $\beta(\cdot, \cdot)$  recursively.
- (c) Accumulate for each model string/phoneme context pair  $(\mu, \rho)$ :

$$\text{acc}(\mu, \rho) = \text{acc}(\mu, \rho) + \frac{\alpha(\eta - k, n - 1)\beta(\eta, n)}{\alpha(L_M, L_R)}$$

### 2. Re-estimation

For each phone context  $\rho$ :

- (a) Compute the sum

$$\text{accsum}(\rho) = \sum_{\mu} \text{acc}(\mu, \rho)$$

- (b) Compute new parameters

$$\hat{P}(\mu|\rho) = P(\mu|\rho) \frac{\text{acc}(\mu, \rho)}{\text{accsum}(\rho)}$$

It is important to note that the groups of model strings and associated phoneme contexts still need to be defined. The variable alignment implies that temporally neighbouring contexts share the same models. The extent of this sharing must be found in experiments. Nevertheless it is clear that even though the number of phoneme contexts remains constant the total number of models used in this context is likely to increase. Any data sparsity problem encountered in the fixed alignment case can be expected to be magnified in this framework. Data sparsity is the limiting factor for the maximum model string length.

### 4.3 Decoding

State of the art ASR HMM-based decoders make simultaneous use of acoustic models, a dictionary and a language model and use the Viterbi approximation at the state level. HMS-HMMs add another knowledge source, namely the model sequence model. The objective in this thesis was to investigate potential benefits in word error rate performance from HMS-HMMs. The question of efficiency in decoding was only of secondary importance in so far as the tools and methods chosen should allow experiments with close to optimal recognition performance and which operate at a reasonable real-time factor. The increase in complexity would make a split of the recognition process into its hierarchical parts more desirable. However, due to the potential increase in search errors a split of the decoding process was avoided. This was possible since the structure of the investigated models both for fixed and variable alignment cases can be embedded into a standard decoder framework and secondly because of the existence of other solutions which limit the computational cost of evaluating complex models. Compared to the bottom-up approach of a hierarchical split a much more favourable solution is to limit the search space either by using N-best lists (Makhoul and Schwartz, 1994) or by rescored word lattices which have been produced by a baseline system.

Both in the fixed alignment and variable alignment cases the kernel of the underlying MSMs are distributions over models  $\mu$  depending on the phoneme context  $\rho$ ,  $P(\mu|\rho)$ . Most state of the art speech recognition decoders make use of static or dynamic networks which contain a context expanded representation of the dictionary. In standard HMM based systems the particular tri-phoneme directly selects the appropriate model and thus the phoneme level nodes in the decoding network are identical. In order to avoid unnecessary duplication of paths, decoders take advantage of the fact that triphone models are shared between a range of contexts. Decoding using HMS-HMMs requires the construction of model networks instead of a direct link to the appropriate HMM. However it is important that the network holds the full context rather than a merged version. Decoding and alignment in this work is based on the Viterbi decoder included in the HTK toolkit, HVITE. HVITE is a static network decoder and perfectly suitable for alignment, decoding with medium vocabularies and limited language models and rescored of N-best lists or word lattices.

The large number of parallel models (see Figures 4.2 and 4.3) inevitably results in a dramatic increase in the computational cost. For standard decoders two processes contribute most of the

overall computational cost, output probability computation and search (token propagation in HVITE). Two factors appear to be important for HMS-HMMs:

1. The models used in parallel can be expected to hold similar output distributions. This means that many or all of these models are active simultaneously.
2. Similar phoneme contexts will share the same models. Even though the likelihood of the observation vector for a particular output distribution is cached at a certain time instance, this has an impact on the number of active models.

The above suggests that not only are many more models used in a particular phoneme context and active at any time, but that a model is more likely to remain active over a longer period of time. In order to improve speed in this case the decoder was modified to precompute the output distribution values for a small number of frames. However the reduction in the overall CPU time relative to a version without pre-computation was less than 5%.

Another important fact is a significant increase in memory requirement. Apart from the memory required to store the MSMs, considerably more memory is required to store the considerably larger static networks and the associated structures required for the search process. However, since all models in one position are normally linked with all other models in the subsequent stage the memory requirement can be reduced by sharing of link tables to the destination models.

## 4.4 Summary

In this chapter the framework of Hidden Model Sequence HMMs (HMS-HMMs) for use in automatic speech recognition was presented. This framework is designed to serve as extension to standard HMMs and is targeted at improving the modelling of pronunciation variation on a sub-phonetic level. As such it adds another level between the acoustic layer represented by the HMM model set and the dictionary. This level is represented by a so called model sequence model. The task of this model is to replace the deterministic mapping from a phoneme sequence to a model sequence by a stochastic mapping. This implies that the phonetic decision trees are replaced by a model without any phone specific knowledge. In keeping the phoneme sequence as basis in the overall model the benefit of independent training and test dictionaries can be retained. It was shown that under a mild level independence assumption<sup>12</sup> the overall model can be trained by using maximum likelihood estimation. The Viterbi approximation at the model sequence level simplifies the overall training scheme. Using the maximum approximation at this relatively high level the effect on word error rate performance can be expected to be minor. The advantage lies in symmetry with recognition, a local decision oriented framework and lower algorithmic complexity. An information theoretic analysis showed that the addition of an hierarchical level in principal reduces the flow of information under the assumption that this flow had been optimal

---

<sup>12</sup>It was assumed that the probability of one sequence only depends on the status of the next higher level.

before. In practice however, the information flow between observation and phoneme sequence can be improved if the increased confusability by a new model is not too large to diminish the anticipated increase in acoustic log-likelihood.

Model sequence models can operate in one of two modes. Either they assume fixed or variable alignment between the model and phoneme sequences. The fixed alignment property allows an immediate association of each phoneme with a particular model without knowledge of the underlying MSM. This case allows the modelling of phone or sub-phone substitutions. In the more complex case of variable alignment the two sequences are not necessarily of equal length and allow for the modelling of HMM insertions and deletions, which are assumed to be important for spontaneous speech. In both cases, there are forms of models that are closely related to clustered triphone models. Decoding using HMS-HMMs can be implemented by a change of standard decoders. However due to the computational cost and memory requirement the rescoring paradigm will be used to obtain recognition results on more complex tasks.

---

## *Implementation of HMS-HMMs*

---

In Chapter 4 the framework of hidden model sequence HMMs as a method to implement stochastic mappings between model and phoneme sequences was described. As such the standard phonetic decision trees which provide a deterministic mapping between the two sequences are replaced by the so called model sequence model. A general distinction was made between the case of fixed and variable alignment between model and phoneme sequences. The general modelling approaches, namely an N-gram model in the case of fixed alignment and a multigram model in the case of variable alignment have been presented. In this chapter a set of important options from a large range of possibilities for implementation of HMS-HMMs are investigated and the implementational details are presented. A range of experiments on two different speech corpora are conducted in order to address the different issues involved. The Resource Management task (see Appendix A.1) has a relatively small vocabulary, is based on read speech and uses a dictionary with one pronunciation per word. The Switchboard transcription task (see Appendix A.2) is based on a large corpus of conversational telephone speech and is comparatively complex and difficult. Apart from a reduced bandwidth and adverse channel conditions for speech over the telephone, pronunciation variation in spontaneous speech is assumed to be a major contributor to the high word error rates found with this task.

This chapter is organised as follows. The first section discusses implementation issues, most importantly with respect to data sparsity for MSMs. Section 5.2 describes the standard HMM baseline systems on the two transcription tasks. In Section 5.3 an initial set of experiments with fixed alignment HMS-HMMs using phone models is described. Section 5.4 discusses experiments based on modelling on a “phoneme plus position within phoneme” dependent level. In that section a range of smoothing techniques, different initialisation techniques, and the modelling of sub-phone insertions and deletions are discussed and experimental results are presented. The last section gives a summary of implementation issues and the experimental results obtained for HMS-HMMs.

## 5.1 Implementation

A fixed alignment between the model and phoneme sequences allows associated pairs to be identified without any knowledge about the underlying model sequence model. The probability of a model sequence is described by the concatenation of probability distributions over the potential set of models drawn from a model sequence model. The complete model sequence model consists of a set of probability mass functions  $P(m|\rho)$  over  $m$  where  $\rho$  is an arbitrary context. The quantities  $P(\mu|\rho)$  can be found by an iterative training procedure. In Section 4.1.2 the decision was taken to use the Viterbi approximation at the model sequence level since the potential impact in any direction is assumed to be minor.

Regardless of the type of MSM used, an E-M derived algorithm is used, which requires that the training of HMS-HMMs has to be performed iteratively. Due to the use of the Viterbi approximation each iteration may be split into 3 parts: the search for the best model sequence using the MSM and HMM set from the previous iteration which are stored for the complete training set; the re-estimation of HMM parameters using the Baum-Welch algorithm; the training of the underlying MSM. The MSM training degenerates to simple N-gram counting in the case of fixed alignment whereas in the case of variable alignment the E-M algorithm is used for maximisation of the data likelihood.

### 5.1.1 Data sparsity

Modelling based on N-grams has to deal with the effects of data sparsity. If the context  $\rho$  in  $P(\mu|\rho)$  is a single phoneme each context is likely to be observed sufficiently frequently, since, with a well defined set of phonemes, good coverage on the training corpus is expected. In the case of tri-phoneme contexts with a normal sized phoneme set not all context triplets can be expected to be seen in the training corpus. Incomplete coverage of the training corpus is unavoidable and is evident in the number of different words in the training corpus and the number of words used in recognition dictionaries on the same task. Unseen tri-phoneme contexts would not pose a problem if the vocabulary in training and recognition were identical and all word-pair combinations had been observed in training<sup>1</sup>. However, language model and dictionary generation is often driven by much larger text corpora which cover a considerably wider range of material. The difference between dictionaries and grammars in training and test introduces unseen tri-phoneme contexts for which some solution has to be found<sup>2</sup>. An obvious solution is to use some other distribution which depends on a less complex context. In the case of MSMs this can be a bi-phoneme context or if that has not been observed as well, a single phoneme can be used. Since the context extends in both directions it is not clear whether the use of the left or

---

<sup>1</sup>Work in this thesis assumes that context spans across word boundaries. This is natural since word boundary information is not part of a phoneme sequence.

<sup>2</sup>The problem of unseen tri-phoneme contexts in standard HMMs is solved by the use of phonetic decision trees which decide on the basis of context questions.

right bi-phoneme context<sup>3</sup> is optimal for this purpose. In this case interpolation of distributions, which will be discussed in more detail in Section 5.1.4, can be used.

Another problem is the sparsity of events within a distribution and the problem of *unseen events*. The problem of unseen tri-phonemes just denotes that the frequency of the context  $\rho$  in the training data  $N(\rho)$  is zero. It is evident that a low frequency  $N(\rho)$  must equally well lead to unreasonable estimates for probabilities of associated models. This ties in with the problem of unseen events, i.e. models have not been observed in a particular context in the training set. Since both unseen contexts and unseen models are in so far related as they all occur with small  $N(\rho)$  a common solution to both problems is desirable. Methods designed to soften the hard decision effects introduced by insufficient training samples are called smoothing techniques.

In language modelling the problems estimating probabilities from sparse data are well known. Specialised estimation techniques have been developed to improve performance in these circumstances. Using a maximum likelihood estimate the unseen events are assigned a probability mass of zero. (Good, 1953) developed an estimate for the probability of events with low frequency known as the Good-Turing estimate. (Katz, 1987) introduced the idea of a contribution of an event with a certain count to the probability of unseen events (discounting). Instead of using a uniform distribution (Katz, 1987) improves the probability estimates for the unseen events by using the probability distribution of less refined contexts for estimation of probabilities for those events (backoff). A different approach is taken by (Jelinek and Mercer, 1980) whereby the smoothing is accomplished by interpolation. The latter solution is in general assumed to yield similar results to discounting and backoff. In the context of MSMs interpolation however is of interest due to the open question of the proper choice of a backoff distribution.

### 5.1.2 Discounting

(Katz, 1987) interprets the process of probability estimation for unseen events as one in which the estimates for all seen events are decreased (discounted) by a certain amount. The discounted probability mass is assigned to unseen events. This interpretation is used by (Ney et al., 1995) to formulate the probability estimate as a function of an explicit discounting factor  $d$ . If a total of  $L(\rho)$  elements are possible in context  $\rho$  and  $M(\rho)$  different symbols have been observed the probability of model  $\mu$  in this context can be computed by

$$P(\mu|\rho) = \begin{cases} \frac{N(\mu,\rho)-d}{N(\cdot,\rho)} & \text{if } M(\rho) < L(\rho) \\ \frac{N(\mu,\rho)}{N(\cdot,\rho)} & \text{if } M(\rho) = L(\rho) \end{cases}$$

The discounting factor  $d$  may depend on the actual count  $N(\mu, \rho)$ . One can distinguish between linear discounting where  $d = aN(\mu, \rho)$  is a linear function of the event frequency and general absolute discounting where  $d$  does not depend on the joint frequency (Federico and de Mori,

<sup>3</sup>Apart from the centre phoneme a left bi-phoneme includes the phoneme to its left whereas a right bi-phoneme includes the phoneme to the right of the centre phoneme.

1998)<sup>4</sup>. Note that even if the amount of probability mass to be assigned to unseen events is known, the choice of either linear or absolute discounting determines how to spread the reduction in probability across the observed events. Experience in language modelling shows better performance by using absolute discounting schemes.

### 5.1.2.1 Good-Turing discounting

This discounting method is the most prominent in language modelling where it yields good results both in terms of perplexity reduction and speech recognition error rates. The method uses the Good-Turing estimate for the probability of event  $\mu$  which (Good, 1953; Nadas, 1985) poses as an alternative to the maximum likelihood solution. The estimation is based on Bernoulli trials and the assumption that events with equal frequency in the training data have to obtain equal probability estimates. If  $q_r$  denotes the probability estimate for any event that occurs  $r$  times and  $C_r$  denotes the number of those events, the Good-Turing estimate is

$$q_r = \frac{(r+1)C_{r+1}}{(N(\cdot, \rho) + 1)C_r}$$

For a detailed discussion of this topic see for example (Niesler, 1997). Using the above estimate the probability mass associated with unseen events can be computed.

$$P_{\text{unseen}} = C_0 q_0 = \frac{C_1}{N(\cdot, \rho)}$$

Under the assumption that the discounting value is subtracted in equal measure from all events the discounting value is

$$d = \frac{C_1}{M(\rho)}$$

This discounting method obviously relies on the fact that a sufficient coverage of elements seen once can be guaranteed or in other words the distributions have smooth long tails. In the case of a large set of events as is the case for language models this is likely to be the case. However in MSMs the number of models per distribution is comparatively small and distributions cover only a few models with poorly modelled tails. Thus in the case of  $C_1 = 0$  the discounting value was artificially set to  $d = \frac{1}{M(\rho)}$ . Another degenerate case is if  $C_1 = M(\rho)$  i.e. all events are observed once. In this case some constant  $c$  is used for discounting.

Since Good-Turing discounting computes the probability mass associated with unseen events on the basis of the frequency of singletons, the estimation can become unstable if singletons themselves are rare events.

---

<sup>4</sup>The term “absolute discounting” is often just used for the case where  $d$  is a constant. This case is discussed in section 5.1.2.3.

### 5.1.2.2 Witten Bell discounting

A more heuristic approach, originally designed for text compression purposes, was taken by (Witten and Bell, 1991). The basic idea is based on the estimation of the probability that the next event observed in a series will be novel. The estimate is based on Laplace' law of succession and the probability is artificially decreased as more events are observed.

$$P(\text{next event will be novel}) = \frac{M(\rho)}{N(\cdot, \rho) + M(\rho)}$$

This is identical to the probability of unseen events. By equal distribution of the above probability mass over all  $M(\rho)$  events observed so far the discounting factor  $d$  is

$$d = \frac{N(\cdot, \rho)}{N(\cdot, \rho) + M(\rho)}$$

### 5.1.2.3 Absolute discounting

The basic idea of absolute discounting (Ney and Essen, 1993) is to leave the high counts virtually unchanged. This is motivated by the fact that a certain event  $(\mu, \rho)$  that occurs in one training corpus with a certain frequency  $r$  can be assumed to occur in another corpus with approximately the same frequency. The difference should be small, e.g. within  $\pm 1$  range of  $r$ . To take account of this fact a non-integer constant  $b$  is used.

### 5.1.3 Backing off

Discounting solves the problem of estimation of probabilities for unseen events by assigning a certain probability mass to them. Given that the number of potential unseen events is finite this probability mass could be uniformly distributed over those events. However, this assumes that nothing is known about those particular events. In many cases knowledge about the events involved can be derived from distributions with less refined context. If for example  $\rho$  is a tri-phoneme context  $(r_{n-1}, r_n, r_{n+1})$  the *primary model distribution*  $P(\mu|\rho)$  may be sparse. However if  $b(\rho) = (r_{n-1}, r_n)$  the distribution  $P(\mu|b(\rho))$  can be expected to have fewer unseen events. (Katz, 1987) proposed to use that portion of the probability mass function of the *secondary distribution* that is associated with the unseen events of the primary distribution. The relevant part is scaled by the discounted probability mass and used as an estimate for the unseen events. Formally we can define the set of unseen events to be  $\mathcal{U}$  and the set of potential events in the primary distribution  $P_p(\mu|\rho)$  to be  $\mathcal{E}_p$  and the sets of potential events in the backoff distribution  $P_b(\mu|b(\rho))$  is  $\mathcal{E}_b$ . The normalisation factor can be simply computed by division of the probability mass assigned to the unseen events in  $P_p$  divided by the mass associated with the elements in  $\mathcal{U}$  in  $P_b$ :

$$\sigma(\rho) = \frac{1 - \sum_{\mu \in \{\mathcal{E}_p - \mathcal{U}\}} P_p(\mu|\rho)}{1 - \sum_{\mu \in \{\mathcal{E}_b - \mathcal{U}\}} P_b(\mu|b(\rho))}$$

and with that scale factor the probability for the unseen events can be computed:

$$P(\mu|\rho) = \begin{cases} P_p(\mu|\rho) & \mu \notin \mathcal{U} \\ \sigma(\rho)P_b(\mu|b(\rho)) & \text{otherwise} \end{cases}$$

Note that the equations remain valid if the two distributions do not share all potential events.

The probability estimates for the unseen events will be more reliable the closer the backoff distribution is to the primary distribution. This implies that only modest context restrictions are desired. However for example, the step to a left bi-phoneme distribution may result in a backoff that again exhibits unseen events, then the backoff chain is a solution. In this example, the left bi-phoneme would backoff to the relatively wide mono-phoneme distribution<sup>5</sup>.

### 5.1.4 Interpolation

Another approach for smoothing of non-parametric discrete probability distributions is the use of interpolation with less specialised distributions (Jelinek and Mercer, 1980; Jelinek, 1997; Ney et al., 1997). Given a certain phoneme context  $\rho$  and some mapping function  $g_i(\rho)$  for selection of parts of the context and the associated distributions  $P_i(\mu|g_i(\rho)) = P_i(\mu|\rho)$  the  $K$ -fold interpolated distribution may be written as

$$P(\mu|\rho) = \sum_{i=1}^K \lambda(i|h(\rho))P_i(\mu|g_i(\rho)) \quad (5.1)$$

Note that the interpolation weights  $\lambda(i|h(\rho))$  depend on the independent context  $h(\rho)$  and have to satisfy the sum-to-one condition  $\sum_{i=1}^K \lambda(i|h(\rho)) = 1$ . In the case of very constrained interpolation weight context, for example context independence, the interpolation weights are often determined manually. However the likelihood of the training data can be maximised using the model depicted in Equation 5.1. Maximisation of the training set likelihood by using the interpolation weights can be performed using the E-M algorithm (see Appendix C) under sum-to-one constraints for distributions and interpolation weights:

$$\hat{\lambda}(i|\bar{\rho} = h(\rho)) = \frac{1}{N(\cdot, \bar{\rho})} \sum_{\rho \in h^{-1}(\bar{\rho})} \sum_{\mu \in \mathcal{M}(\bar{\rho})} \left( N(\mu, \rho) \cdot \frac{\lambda(i|\bar{\rho})P_i(\mu|\rho)}{\sum_j \lambda(j|\bar{\rho})P_j(\mu|\rho)} \right)$$

$$\hat{P}_i(\mu|\bar{\rho} = g_i(\rho)) = \frac{\sum_{\rho \in g_i^{-1}(\bar{\rho})} \left( N(\mu, \rho) \frac{\lambda(i|h(\rho))P_i(\mu|\bar{\rho})}{\sum_j \lambda(j|h(\rho))P_j(\mu|\rho)} \right)}{\sum_{\rho \in g_i^{-1}(\bar{\rho})} \sum_{\mu \in \mathcal{M}(\bar{\rho})} \left( N(\mu, \rho) \frac{\lambda(i|\rho)P_i(\mu|\bar{\rho})}{\sum_j \lambda(j|\rho)P_j(\mu|\rho)} \right)}$$

<sup>5</sup>The backoff can be extended depending on the set of contexts sharing the same models. In this work the backoff chain ends at mono-phoneme contexts.

Note that the inverse of a context mapping function  $g_i^{-1}(\bar{\rho})$  defines a set of contexts with the appropriate context width. The formulae are based on an estimate of the posterior probability that the  $i^{\text{th}}$  distribution is responsible for generating the training data.

The interpolation solution has the advantage that the sometimes heuristic approaches described in Section 5.1.2 can be replaced by a mathematically sound framework. Not only are the interpolation weights estimated on the training data, all probability estimates themselves are adjusted accordingly. One disadvantage is the fact that the E-M algorithm provides an iterative solution and does not guarantee to yield the global optimum. Another drawback is that the interpolation weights cannot be estimated on the complete training set since the result would degenerate for each context  $h(\rho)$  to an interpolation weight of one the distribution with the most refined context. A held-out set has to be used for estimation of interpolation weights. In practice, the held-out set can be much smaller than the complete training set since only a few parameters need to be estimated. Linear interpolation is found to yield similar probability estimates to those obtained by backing off to less refined distributions in hierarchical fashion.

### 5.1.5 Pruning and perplexity

Depending on the particular realisation of a model sequence model the number of models per phoneme context may well exceed 100. Systems implemented in this thesis vary between an average of 1 to 150 models per phoneme context. The number of models for each phoneme context must be multiplied to obtain the number of potential model sequences. In the case of a straightforward implementation of the search for the best model sequence, the computational cost is multiplied by the number of all potential model sequences. However, the Viterbi approximation allows the multiplication factor to be limited to approximately the average number of models per distribution or the average branching factor. The average branching factor is also known as the corpus perplexity (Bahl et al., 1983) and is commonly computed to characterise a test set in relation to a specific model. The perplexity given the pair of sequences  $(\mathbf{M}, \mathbf{R})$  can be computed by:

$$PP = P(\mathbf{M}|\mathbf{R}, \theta)^{-\frac{1}{L_M}}$$

In the case of fixed alignment

$$\log PP = -\frac{1}{L_M} \sum_{n=1}^{L_M} \log(P(m_n|\mathbf{r}_n, \theta)) \quad (5.2)$$

Due to the nonlinearity of the logarithm events with small probability make the largest contribution to the perplexity, which in turn decreases the likelihood of the training data. Beside the disadvantageous effect of increased computational cost, the discussion in Section 4.1.3 made it clear that the increase in confusability is not desirable. Thus a scheme to limit the number of

models was used throughout the experiments conducted. In this scheme the  $M(\rho)$  elements  $\mu_i$  of a distribution are sorted according to their probability estimate:

$$P(\mu_{i-1}|\rho) \leq P(\mu_i|\rho) \leq P(\mu_{i+1}|\rho) \quad i \in [1 \dots M(\rho) - 1]$$

All elements of the distribution with index  $k \geq j$  are deleted if the accumulated sum over all elements with higher probability exceeds a certain threshold  $\gamma$ :

$$\sum_{i=1}^j P(\mu_i|\rho) \geq \gamma$$

In some cases multiple symbols obtain identical probability estimates. If only some of these symbols fall below the threshold, all are kept. The threshold is the amount of probability mass retained. Unless stated otherwise, 95% of the probability mass was kept throughout this thesis during training and 97% during tests. In specific cases a severe constraint on the number of models was necessary. In these cases a hard limit on the number of models regardless of the probability mass discarded was also used.

### 5.1.6 Model sets

An MSM is a set of probability distributions dependent on the phoneme context. The elements of these probability distributions, namely HMMs, are drawn from an *a priori* fixed and finite set. In principle each model may occur in any context. This is similar in concept to the use of tied mixture models (Bellegarda and Nahamoo, 1990), albeit on the level of complete HMMs rather than individual Gaussian mixture components. This however inevitably leads to a dramatic increase in confusability which can hardly be matched by the anticipated gain in likelihood. Thus a more constrained model sharing approach needs to be chosen.

Triphone models are a natural extension to monophone models in that the original *centre* phoneme context is extended by its left and right neighbour. Thus the set of all models associated with phoneme triplets which share the same centre phoneme are the most likely candidates for replacement. This fact is also used in model smoothing in the case of generalised triphones (Lee, 1990) and more importantly for parameter tying based on phonetic decision trees (Young et al., 1994). Apart from the clustering aspect, decision trees are required to predict the appropriate model for tri-phoneme contexts unseen in the training data. This prediction is likely to be better if at least the centre phoneme remains constant.

It is not the objective of this work to explore the wealth of possible ways in which models can be shared. The obvious solution stated above is sufficient for investigation of the general suitability of HMS-HMMs for speech recognition. Thus apart from experiments conducted in Section 5.4.6 a particular set of models will be shared across all tri-phoneme contexts which share the same centre phoneme. In the case of phoneme position dependent modelling (state level models, see

Equation 4.17) the set of models is further limited to originate from the same position within a phoneme.

### 5.1.7 Scaling and normalisation

The use of scale factors for scaling the language model log-probabilities is a standard technique to adjust the dynamic ranges between the acoustic model likelihood and the language model probability estimate. It is assumed that the scaling is required since the estimates for the likelihood of the observation sequence are much too small (see Section 2.6.2). The MSM adds another layer between the acoustic model and the language model, which is not trained independently as is the case for language modelling. However the coupling is only via computation of the best model sequence. Thus it is natural to assume that the use of a scale factor may improve the performance. Since the MSMs operate on a model level rather than on the word level the optimal scale factor can be assumed to be equal or lower than the factor used for scaling the language model. In the same way as scaling of the LM probabilities, the scaling is implemented as a linear scaling in the log domain. The appropriate scale factor has to be found by a search for the factor which yields optimal performance. However this is not conducted for each HMS-HMM configuration individually. Experimental evidence suggests that the optimal scale factors are mostly task dependent and can be assumed to be independent of the test set.

Another issue which was found to improve performance mainly on Switchboard, was the re-normalisation of the probability distributions over models in a certain phoneme context such that the highest probability in the distribution assumes a value of 1.0. This is designed to alleviate the imbalance in the number of models for particular phonemes. This normalisation has no effect in training and was found not to improve performance on RM.

## 5.2 Baseline systems

HMS-HMMs have been tested and compared with standard HMMs on two different speech transcription tasks. The first rather simple task is based on the Resource Management corpus and serves as an example of a clean read speech transcription task. The low complexity of the task allows relatively fast turnaround time. Initial experiments on this corpus usually give an indication about the performance of a certain technique under investigation, unless specifically targeted at other speech types. A thorough comparison with standard methods under a limited number of influence factors is possible with this task. However an important objective of the work presented is an improvement in acoustic modelling of spontaneous speech. Work on spontaneous speech often uses the Switchboard corpus (Godfrey et al., 1992) which is a large database of two speaker conversations spoken over a standard telephone channel. The transcription of speech data drawn from this corpus is presently one of the most difficult tasks in automatic speech recognition (Young and Chase, 1998). Comparable state-of-the-art speech transcription systems in terms of complexity yield much higher word error rates on this task than for example on

portions of the Broadcast News corpus. An important proportion of this difference has been attributed to differences and variation in pronunciation (Weintraub et al., 1996b; Saraçlar et al., 2000). However pronunciation modelling on the phoneme level only yields small improvements in word error rate (see e.g. (Riley et al., 1999)). The implementation of HMS-HMMs for this task focuses on the modelling of sub-phonetic variation. The following two sections give a brief description of the respective standard HMM baseline systems.

### 5.2.1 Resource Management

The Resource management (RM) task is based on transcription of read queries on the status of Naval resources. A detailed description of the Resource management task is presented in Appendix A.1. The training set is relatively small with 3.8 hours of speech, but 4 test sets, namely the feb89, oct89, feb91, and sep92 sets, with a total of 1.1 hours of speech. A setup similar to the one presented in (Young et al., 1994) was chosen as the baseline scenario: the speech was encoded as standard 12 dimensional MFCC parameters together with the filterbank energy computed at a frame rate of 10ms. The addition of first and second order derivatives gave a 39 dimensional feature vector. Triphones models were clustered using phonetic decision trees. The setup uses phone HMMs with a standard left-to-right 3 state model topology (see Figure 2.2). A system tied on the model level was required to serve as baseline for experiments in Section 5.3, whereas the baseline system for Section 5.4 uses phonetically clustered states. In both cases the standard HTK implementation (Young et al., 1999) was used for clustering and training of model sets and the phoneme context was computed across word boundaries.

System	feb89	oct89	feb91	sep92	Average
Model-tied	3.79	5.51	4.91	8.28	5.63
State-tied	3.16	3.80	3.30	6.17	4.11

Table 5.1 *RM baseline systems: %Word error rates using state and model tied systems on all four test sets*

Table 5.1 shows %word error rates for both model and state tied triphone model sets. The model-tied model set used contained 1153 different HMMs with 4 mixture components per output distribution<sup>6</sup>. The state tied model set consists of 1581 tied states with 6 Gaussian mixture components per state. Even though the model-tied system has 46% more parameters than the state tied system the relative performance difference in word error rate is almost 27%.

### 5.2.2 Switchboard

The Switchboard corpus is a large corpus of conversational telephone speech and has been designed to serve as the basis for the NIST IVCSR evaluations (Young and Chase, 1998). The speech data is transcribed and segmented manually. A training set containing approximately 18

<sup>6</sup>A larger number of mixture components showed a increase in word error rate due to over-training.

hours of speech called MiniTrain was used for all experiments on this corpus<sup>7</sup> in conjunction with two test sets called MTtest and WS96devsub which in total contain approximately 1.06 hours of speech data. A more detailed description of the corpus and the training and test sets is presented in Appendix A.2.

The baseline model sets make use of the data encoded in 12 PLP coefficients (see Section 2.2.1) and the zeroth cepstral coefficient, which together with first and second order derivatives yields a 39 dimensional feature vector. Training of models on the Switchboard corpus requires special techniques to reflect the properties of the data. The relationship of the number of speech segments against the total amount of speech reveals a relatively short average utterance duration. Thus per utterance cepstral mean normalisation is not effective and is replaced by normalisation per conversation side. The strict speaker/side separation allows the very effective use of vocal tract length normalisation (VTLN) in training and test. VTLN was shown to be even more effective if per conversation side variance normalisation was used (Hain and Woodland, 1998). Side-based variance normalisation is also used for the training of non-VTLN models. Experiments to be presented in Section 5.4.8 will make use of VTLN models.

The training set vocabulary includes 8582 different words including annotated false starts and hesitations. The test vocabulary consists of 24157 words which were obtained from the complete Switchboard corpus. The dictionaries used for training and test are based on an extended version of the LIMSIS 1993 WSJ dictionary (Gauvain et al., 1994). The average number of pronunciation variants for the test dictionary is 1.12 and for the training dictionary 1.15.

Model setup	VTLN	MTtest	WS96devsub	Average
baseline-1	no	43.68	46.32	45.04
baseline-2	no	42.72	45.97	44.39
baseline-2	yes	38.76	39.01	38.89

Table 5.2 *HMM baseline word error rates for state clustered triphone models and a trigram language model. Baseline-2 depicts a system with approximately equal number of states, but slightly modified silence model.*

Results in Table 5.2 show the results for state clustered triphone baseline model sets trained on the MiniTrain training set. Experiments were conducted using a trigram language model trained on approximately 2.5 million words (see Appendix A.2). The baseline-1 setup was used for all experiments the Sections 5.3 and 5.4 apart from those presented in Section 5.4.8. Through the course of work on this thesis the silence model structure was found to be suboptimal which was improved in the baseline-2 setup<sup>8</sup>. The results in Table 5.2 indicate considerable improvement in word error rate by using VTLN of 12% relative which is slightly more than can normally be expected. The baseline-1 system has 2954 tied speech states with 12 mixture components

<sup>7</sup>Apart from experiments presented in Appendix B.2.

<sup>8</sup>CU-HTK systems make use of 2 silence models, one standard pause model *sil* and one for modelling of short pauses between words *sp*, where the latter is context-preserving. The modified setup uses identical 3 state left to right topology for *sil* and *sp* with shared states whereby the *sp* model can be skipped. In the original setup the two models are untied and *sp* consist of only one state which may be skipped.

whereas the baseline-2 system uses 3088 tied speech states with the same number of mixture components per state.

The baseline-1 and baseline-2+VTLN model sets have been used to generate lattices in a two stage process. Initial lattices are produced using a bigram language model which was trained on the same data as the trigram model. The bigram lattices were expanded with the trigram model and subsequently pruned. The lattices have an oracle word error rate<sup>9</sup> of approximately 10%<sup>10</sup>. These lattices have been used for rescoring experiments with HMS-HMMs.

### 5.3 Phone models

A set of initial experiments on RM was designed to investigate the behaviour of HMS-HMM systems and to make an initial comparison of a standard HMM setup with HMS-HMMs based on fixed alignment. In particular the set of experiments should establish how the HMS-HMM framework interacts with the use of Gaussian mixture distributions. For this purpose 3 versions of the model-tied RM system described in Section 5.2.1 with 1, 2 and 4 mixture components per speech state were trained and used to obtain baseline performance. Apart from serving as a baseline the respective model sets were used to produce 20-best lists for rescoring with an associated 1, 2 and 4 mixture component HMS-HMM model set. Furthermore the baseline system served as a system for bootstrapping associated HMS-HMMs. The 1153 clustered 3 state left-to-right triphone models were formed into 47 sub-sets such that all triphone HMMs sharing a particular centre phoneme belong to one sub-set of models. This sub-set of models is used as the basis for all probability distributions associated with the same centre phoneme.

The HMS-HMMs use 3 sets of statistics: a set of distributions over models dependent on the mono-phoneme context; a set of distributions over models dependent on the left bi-phoneme context; and a set of probability distributions depending on the set of tri-phoneme context. The left bi-phoneme is smoothed by discounting using the Good-Turing based scheme as described in Section 5.1.2.1 and Katz backoff to the corresponding mono-phoneme distribution which employs no smoothing. The tri-phoneme distributions are smoothed by Good-Turing discounting and backoff to the left bi-phoneme distributions or if those are not present either, to the associated mono-phoneme distribution. In the initialisation stage uniform distribution over all models possible for a particular mono-phoneme is assumed.

Table 5.3 shows word error rate performance for the standard HMM and the HMS-HMM setup on the feb89 test set by rescoring of 20-best lists generated by the associated baseline system. The relative word error rate comparison indicates an improvement between 8% and 14% relative over the baseline systems and no adverse effect from the joint use of HMS-HMM and mixture components was found. The HMS-HMM were trained with no pruning of MSM distributions and a scale factor of 5.

---

<sup>9</sup>The oracle error rate is the minimal word error rate of all potential paths through a word graph.

<sup>10</sup>The computation of the oracle error rate on this type of data is difficult due to the effect of equivalent forms. The computed number can only give a rough indication about the lattice quality.

#mix comp	HMM	HMS-HMM	$\Delta\%$ WER relative
1	7.38	6.68	-9.5
2	4.92	4.53	-7.9
4	3.79	3.24	-14.5

Table 5.3 %WERs and relative improvement on the RM feb89 test set using 1153 crossword model-clustered triphone HMMs and an increasing number of mixture components (#mix comp). Both HMM and HMS-HMM use the same number of HMM parameters. HMS-HMM results were obtained by 20-best rescoring.

System	feb89	oct89	feb91	sep92	Average
HMM	3.79	5.51	4.91	8.28	5.63
HMS-HMM	3.16	5.10	3.86	6.99	4.79

Table 5.4 %WERs on all RM test sets using model clustered HMM models and equivalent HMS-HMMs. Both systems use 1153 HMMs with 4 mixture components. HMS-HMM results were obtained using 40-best rescoring.

Table 5.4 gives individual and overall results for all four RM test sets for systems using 4 mixture components. Word error rates were obtained by rescoring of 40-best lists generated by the baseline system. The improvements for each test set vary greatly from 7 to 22% relative. Overall an improvement over the baseline of 0.84% WER absolute or 14.9% relative was achieved. Again the system used no pruning of MSM distributions and an MSM scale factor of 5.0. Note that the number of HMM model set parameters in both systems is  $\approx 1,090,000$ . In the case of the clustered HMM system the number of parameters could not be increased by adding more mixture components without loss of performance. The final number of parameters in the MSM is only 33,102.

## 5.4 Phoneme position dependent models

As outlined in Section 5.2.1 the performance of model-tied systems is generally poorer than that obtained by models sets clustered at the state level. Clustering of states or output distributions using decision trees is based on maximisation of the data likelihood. It requires certain assumptions during the clustering process (see Section 3.5), most importantly constant state to observation alignment with models of low complexity. Fixed alignment HMS-HMMs can be viewed as an alternative to using tied states. None of the necessary assumptions for state tying are required for modelling using HMS-HMMs which is an important property of this framework. One important difference between phonetically tied systems and HMS-HMMs is that the latter do not include an inherent clustering procedure as is the case for state-tied systems. The model sets and the sets of contexts sharing these sets has to be assumed or defined a priori.

The set of output distributions to be clustered is commonly determined by the centre phoneme and the position of the state within the 3-state left-to-right model topology. The HMS-HMM

equivalent of state-clustered systems makes use of the concept of phoneme positions (see Equation 4.17) explicitly by assuming that a certain constant number of consecutive models is produced by a certain phoneme configuration. The location of a model is well defined by the definition of the phoneme context and an index within that context.

The HMS-HMM equivalent of state-tied HMM sets is initialised by using the HMM phone states from the baseline system. The states of the state clustered model set are reorganised into separate single state HMMs. Those state models that were associated with a particular centre phoneme and a particular position within the phone model are combined into a set and shared between the all distributions with that particular centre phoneme and within-context position.

	feb89	oct89	feb91	sep92	Average
HMM	3.16	3.80	3.30	6.17	4.11
HMS-HMM	2.77	3.13	2.62	5.20	3.43

Table 5.5 %WERs on all RM test sets using crossword state-tied triphone HMMs with a total of 1581 states for both HMM and HMS-HMM. The word-pair grammar was used in decoding.

Table 5.5 shows results on all RM test sets for both the baseline HMM and the HMS-HMM systems. The HMS-HMM system used the same type of backoff chain as in the model-tied case: from tri-phoneme to left bi-phoneme to mono-phoneme. In the case of an unseen tri-phoneme context the appropriate left bi-phoneme distribution was chosen. However Witten-Bell discounting was found to yield slightly better performance than Good-Turing discounting. The optimal MSM scale factor was chosen to be 6.5 and this value was kept for all subsequent experiments using HMS-HMMs on RM. HMS-HMM results were obtained by Viterbi decoding with the standard word-pair grammar. The WER improvement over all test sets was 16.5% relative and is well balanced across the individual test sets.

After initialisation from the baseline system a total of 7 iterations of model level Viterbi based training was performed and pruning of MSM distributions at 95% level was used in training and at 97% level in test. Figure 5.1 shows the change in average per frame log-likelihood. The dashed line represents the log-likelihood of the training data obtained with the baseline model set. The curve representing the HMM part of the system shows much higher likelihood of the observed data as was predicted in Section 4.1.3. The figure shows that the step of 0.65 in average log-likelihood is made during the initialisation phase whereas the change in later iterations is moderate. The change in the total log-likelihood stems from an improved MSM model. After 4 iterations the total log-likelihood of the HMS-HMM is higher than that of the baseline HMM, but the difference remains small in further iterations. After 5 or 6 iterations the change in log likelihood becomes very small.

A similar set of experiments was conducted using the Switchboard corpus. The step to this task represents a change in the amount of training data, speech type, language model and dictionary. All experiments on this corpus are conducted by using the MiniTrain training set and a combination of the MTtest and WS96devsub test sets. An approximate comparison of the

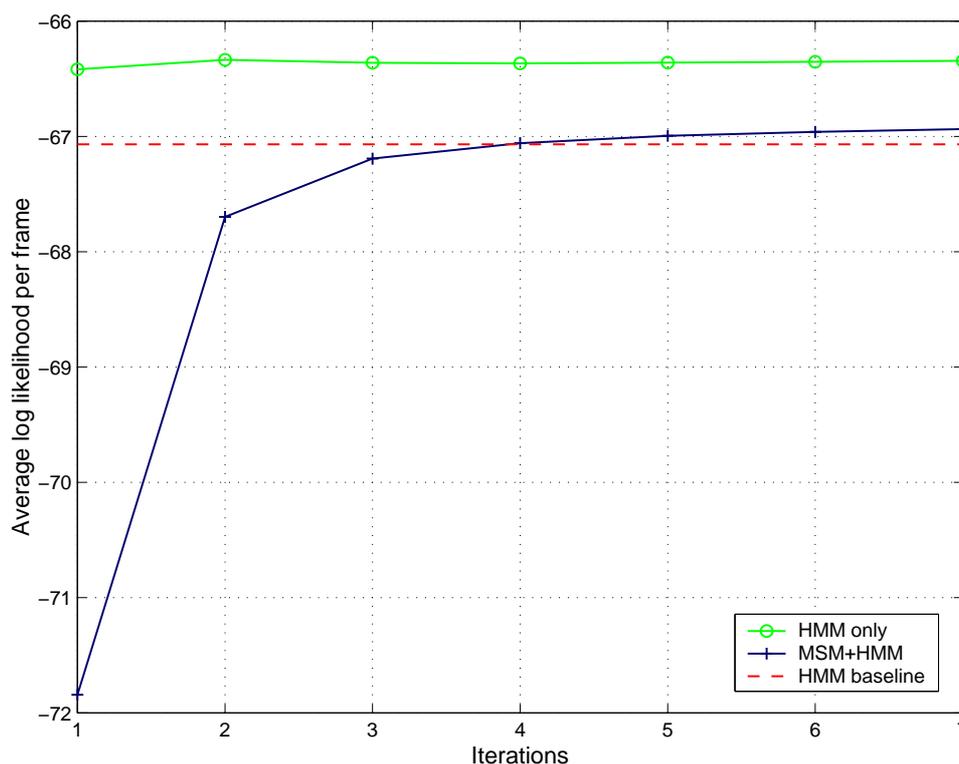


Figure 5.1 Average per frame log-likelihood on RM over several training iterations. (*HMM only*) is the likelihood of the observation sequence given the best model sequence  $\frac{1}{L_o} \log P(\mathbf{O}|\mathbf{M}^*)$ . (*MSM+HMM*) denotes the total average log-likelihood of the HMS-HMM system  $\frac{1}{L_o} (\log(P(\mathbf{M}^*|\mathbf{R})) + \log P(\mathbf{O}|\mathbf{M}^*))$ . The dashed line represents the average log-likelihood per frame of the HMM baseline system.

system setup with that on the RM task shows a 24-fold increase in vocabulary size and the use of a trigram language model with a perplexity of 100 instead of a word-pair grammar with a perplexity of 60 (Young and Chase, 1998). The acoustic models have approximately 4 times as many parameters with doubling both in the number of states and the number of mixture components. In order to cope with computational complexity the lattice rescoring paradigm is used for all subsequent experiments on this corpus.

As for the RM experiment, the Switchboard HMS-HMM was initialised by reorganisation of the baseline HMM set. Single state models belonging to a certain centre phoneme and position in the baseline model set were collected into 135 model sets. Figure 5.2 shows the number of models for each of the 45 phonemes (see Appendix A.2)<sup>11</sup>. The number of models for each position is plotted in a separate line. First note the large variation in the model set sizes ranging from one model to a maximum of 67. Phonemes with fewer than 5 models associated with them were /em/, /en/, /oy/, /ch/, /ix/ and /jh/. Phonemes with more than 40 models were (in increasing order) /r/, /ih/, /iy/, /ax/, /n/, and /t/. Interestingly the number of models does not vary greatly between phone positions. In Figure 5.2 the frequency of all phonemes on the training set was normalised and added. From the similarity of all curves in that figure it is

<sup>11</sup>Silence models are excluded from this figure.

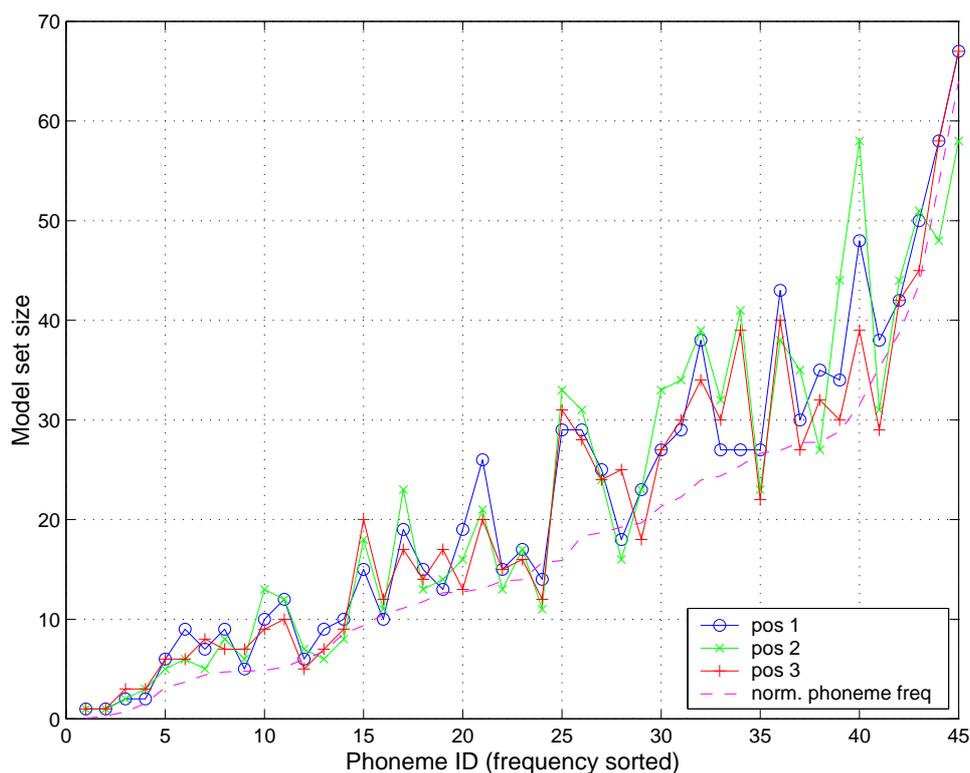
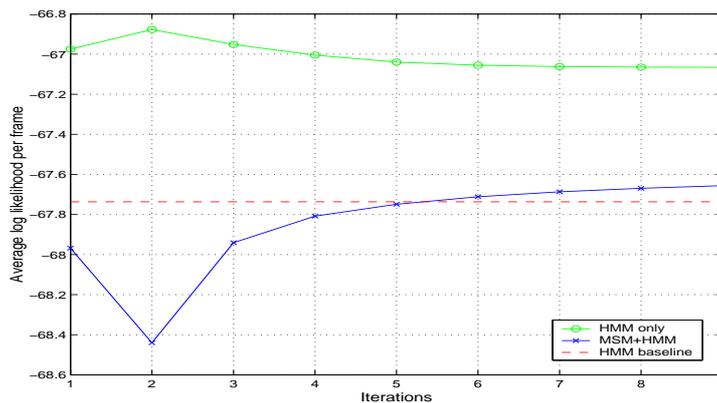


Figure 5.2 Sorted number of single state models used for fixed alignment HMS-HMMs for each of the 3 phoneme positions. The dashed line shows the normalised frequency of phonemes in the training data. Normalisation was performed such that the the highest frequency was identical to the average of the highest number of models in each position.

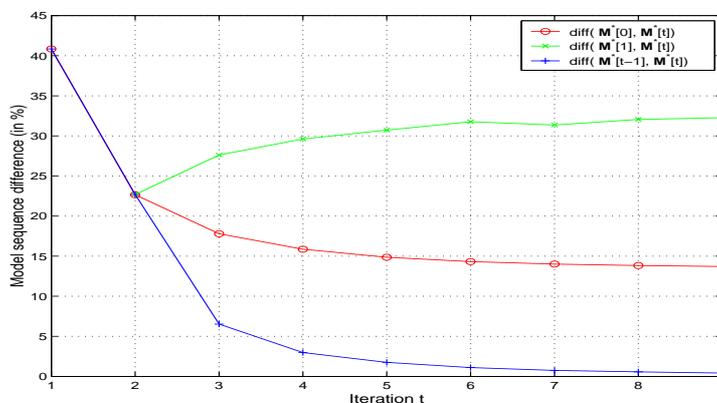
obvious that the amount of data is the predominant factor in the determination of the number of models during state clustering.

During the first iteration of HMS-HMM training, a uniform distribution over all models in each associated set of contexts was assumed. Further training iterations used pruning at the 95% level and an MSM scale factor of 4.0. The MSM model backoff structure was identical to the one used in the previously presented RM experiments. Figure 5.3 shows graphs characterising the training procedure. Figure 5.3(a) corresponds to Figure 5.1 obtained for HMS-HMMs on RM. The change in log-likelihood is less prominent in the first iterations than for RM. Similarly the log-likelihood of the HMM part is considerably higher than for the baseline system and barely changes with further training iterations. After the 5<sup>th</sup> iteration the total log-likelihood of the training data for the HMS-HMM is higher than for the baseline system.

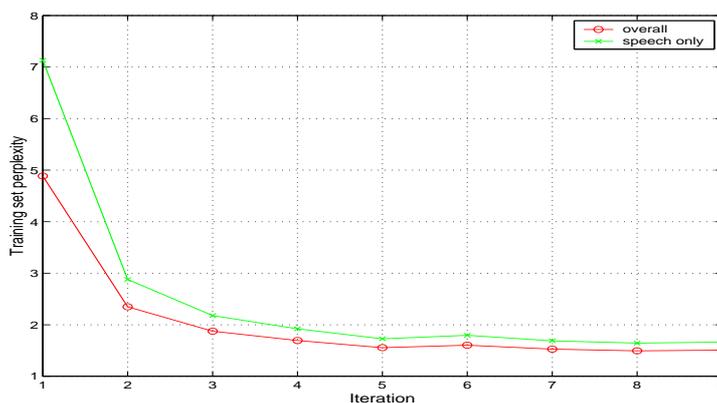
The change in log-likelihood would suggest only minor changes to the underlying models. Figure 5.3(b) shows the difference between model sequences produced in subsequent iterations  $\text{diff}(\mathbf{M}^*[t-1], \mathbf{M}^*[t])$ , the difference to the model sequence  $\mathbf{M}^*[0]$  reflecting the state clustering from the initial HMM set  $\text{diff}(\mathbf{M}^*[0], \mathbf{M}^*[t])$  and the difference to the model sequence obtained during the first iteration  $\text{diff}(\mathbf{M}^*[1], \mathbf{M}^*[t])$ . The difference is measured in terms of the ratio between the number of models changed versus the total number of models in the sequence. The



(a) log-likelihood



(b) Model sequence difference



(c) Perplexity

Figure 5.3 Training of HMS-HMMs on Switchboard. (a) shows the average HMM and HMS-HMM log-likelihoods; (b) displays the difference between model sequences of successive iterations and to the first iteration; (c) shows the training set perplexity of the MSM.

change in the model sequence between  $M^*[0]$  and  $M^*[1]$  is 40.84%, where  $M^*[1]$  is generated using a uniform distribution over all possible models. The change between the first and the second iteration is 22.69%. Note that the difference to  $M^*[0]$  is steadily decreasing while the difference to  $M^*[1]$  is rising. The change in difference gradually is reduced and after 9 iterations the difference to  $M^*[0]$  is 13.71%. However, after 6 iterations only minor changes are observed. It was noticed that in most cases only a single model is changed rather than a complete sequence of models which suggests that dependence on previous models is of minor importance. A further indication that major changes in the model structure are made can be deduced from the change in perplexity (see Equation 5.2) on the training data in Figure 5.3(c). The perplexity is an estimate on the average number of parallel models and starts at a high level of almost 5. After 5 iterations the small change in model sequences is reflected by only minor changes in perplexity to a level around 1.5. Note that this number is lowered by the distributions for silence models where only one choice of model is possible. This effect can be corrected and the perplexity for speech models only converges to around 1.65. The complexity of the task can be shown by a comparison with the corresponding perplexity on the RM system. The speech only perplexity on the RM training set is 1.33 which is 19% lower than that obtained with the Switchboard model set.

	MTtest	WS96devsub	Average
HMM	43.68	46.32	45.04
HMS-HMM	43.07	43.95	43.52

Table 5.6 %WERs on Switchboard data for state clustered HMMs using a phonetic decision tree and for state level HMS-HMMs. HMS-HMMs use a backoff to left bi-phoneme distributions. All models are trained on MiniTrain and results are obtained by rescoring of trigram lattices.

Table 5.6 shows word error rate results on the two test sets obtained by rescoring of trigram lattices. The overall performance difference is 1.5% WER absolute. An improvement was achieved on both test sets, however results on the WS96devsub were significantly better with a change by 2.37% WER absolute.

#### 5.4.1 Smoothing

The MSM in the previous experiments used Witten-Bell discounting and a backoff scheme to left bi-phoneme distributions for the smoothing of probability estimates. Even on larger training set data sparsity is a problem, not least due to the imbalance of the training data with respect to the frequency of phonemes (see Figure 5.2), even though the problem of sparsity is mitigated by the fact that the number of potential models is smaller with the frequency of the context. Clearly effective smoothing techniques are needed: the following sections explore a range of options for smoothing of MSM distributions.

### 5.4.1.1 Backoff to simple interpolated distributions

Distribution backoff requires the existence of secondary distributions which depend on less refined context. All experiments presented in this chapter use a tri-phoneme context, that is for the  $n^{\text{th}}$  model in the sequence the triplet  $(r_{n-1}, r_n, r_{n+1})$  is used. This leaves a range of possibilities for use in backoff, namely the bi-phoneme contexts  $(r_{n-1}, r_n)$ ,  $(r_n, r_{n+1})$ . For experiments so far the left bi-phoneme context  $(r_{n-1}, r_n)$  was arbitrarily chosen for primary backoff. Since the right context is not a subset of the left context the only sensible option to include right context information into backoff is to interpolate both model distributions:

$$P(m_n|r_{n-1}, r_n, r_{n+1}) = \lambda P(m_n|r_{n-1}, r_n) + (1 - \lambda)P(m_n|r_n, r_{n+1})$$

A further set of experiments was conducted with the improved backoff scheme. The interpolated distribution serves as the secondary distribution<sup>12</sup>. Each bi-phoneme distribution is smoothed by Witten-Bell discounting and backoff to the mono-phoneme distribution. If a bi-phoneme distribution does not exist the corresponding mono-phoneme distribution is used.

Table 5.7 shows word error rate results on RM for the improved backoff scheme in comparison to the HMM baseline and the backoff to left bi-phoneme distributions only. The interpolation weight was set to  $\lambda = 0.5$ . Word error rates improved on three out of the four test subsets and an absolute improvement of 0.13% WER over all test sets was obtained. Compared to the HMM baseline this constitutes a relative WER reduction of 19.7%. This result and the associated model set serves as basis for further investigations of HMS-HMMs.

System	Backoff	feb89	oct89	feb91	sep92	Average
HMM	-	3.16	3.80	3.30	6.17	4.11
HMS-HMM	left bi-phoneme	2.77	3.13	2.62	5.20	3.43
HMS-HMM	interpolated	2.62	3.20	2.54	4.81	3.30

Table 5.7 %WERs on all RM test sets. Results for HMS-HMMs are obtained with backoff to left bi-phoneme or to interpolated left and right bi-phoneme distributions.

The statistical significance of these results has been tested. The same tests as used for comparison of speech recognition systems by the U.S. National institute of Standards and Technology (NIST) are used (Gillick and Cox, 1989; Martin, 1995). These tests are the *matched pair sentence segment test* (MP), the *signed paired comparison test* (SI), the *Wilcoxin signed rank test* (WI) and the *McNemar test*. In all tests the null hypothesis is that there is no performance difference between systems. The most important of these tests is the MP test which is based on the null hypothesis that the mean difference in the number of word errors per sentence is zero. SI and WI compare word error rates per speaker. The McNemar test is only applicable in cases where errors can be considered to be independent and is not used in this thesis. A difference at a level of 0.05 is considered to be statistically significant. An analysis of the outputs from the HMM baseline

<sup>12</sup>See section 5.1.4.

system and the HMS-HMM using interpolated backoff showed that there is significant statistical difference in the output of the systems, The individual tests showed statistical difference at a level less than 0.001 for the MP and WI tests and 0.007 for the SI test<sup>13</sup>.

System	Backoff	MTtest	WS96devsub	Average
HMM	-	43.68	46.32	45.04
HMS-HMM	left bi-phoneme	43.07	43.95	43.52
HMS-HMM	interpolated	42.80	43.92	43.38

Table 5.8 %WERs on the Switchboard. Results for HMS-HMMs are obtained with backoff to left bi-phoneme or to interpolated left and right bi-phoneme distributions.

The results for the corresponding experiment on the Switchboard corpus are presented in Table 5.8. The training set perplexity was lowered by only 0.03, but a small improvement in word error rate by 0.14% WER absolute was obtained. Overall the system shows a 1.66% WER absolute improvement over the baseline. Again the test for statistical significance was carried out to assess the difference between the two modelling strategies. The MP, SI and WI tests showed significant difference at a level of less than 0.001, 0.046 and 0.002 respectively. Thus the WER obtained with the HMS-HMM system is significantly different to the HMM baseline.

#### 5.4.1.2 Backoff to interpolation by deleted estimation

Experiments in the previous section suggest that interpolation is an interesting option for HMS-HMMs. The manual tuning of interpolation weights remains unsatisfactory unless the effect of other strategies is minor. In Section 5.1.4 a detailed description of the underlying theory for optimisation and selection of interpolation weights was given. Due to a number of disadvantages of this technique the use for construction of MSMs focused on the optimisation of the set of backoff distributions.

The disadvantages of the method are the necessity to use the E-M algorithm, the requirement for iterative optimisation and the fact that held-out data must be used for estimation of the interpolation weights. Due to the non-optimality of the E-M algorithm proper initialisation of model parameters is required. The iterative optimisation can be constrained such that only one E-M re-estimation is performed during one HMS-HMM training iteration to avoid rapid over-training. With respect to held-out data (Jelinek, 1997) mentions that is possible to use the complete data for re-estimation of the distribution probabilities and a subset for optimisation of the interpolation weights. This strategy was used for training of HMS-HMMs. In comparison with a full division of the data this method was found to perform better.

The sets of distributions to be used in interpolation are left and right bi-phoneme distributions and the mono-phoneme distribution for a particular centre phoneme and position within that phoneme, which was also chosen to serve as context for the sets of interpolation weights. The

<sup>13</sup>The tests results were obtained using the NIST SCTK software package, version 1.2.

	feb89	oct89	feb91	sep92	Average
backoff	2.62	3.20	2.54	4.81	3.30
10% heldout	2.54	3.43	2.42	4.92	3.33
20% heldout	2.62	3.06	2.78	4.88	3.33
30% heldout	2.66	3.02	2.42	5.04	3.29

Table 5.9 %WERS on all RM test sets for systems trained with deleted estimation of interpolation weights.

primary context distributions still use Witten-Bell discounting and backoff to the interpolated distributions.

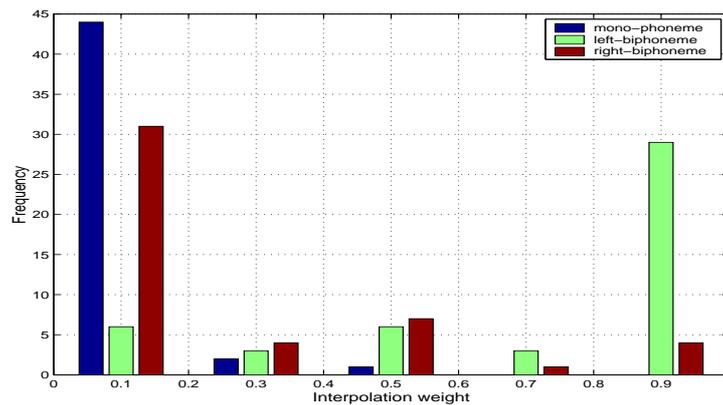
Experiments were conducted on RM by initialising from the HMS-HMM model set as presented in the previous section. The best model sequence from the last training iteration of the initialisation model set was used for initialisation of the MSM. The model distributions were initialised by simple use of ML estimates and the interpolation weights were assumed to be uniform. Table 5.9 shows word error rate results for the baseline system and different amounts of held-out data. As can be seen from the results for estimation of the small number of interpolation weights 10% of the training data is sufficient. The total word error rate is not significantly different to that obtained by use of the simple interpolation scheme described in the previous section. After 8 iterations the backoff weights degenerate in most cases to make use of only one distribution.

Figure 5.4 shows the distribution of interpolation weights for each position within a phoneme. First of all it is evident that the mono-phoneme distributions have very small weights. As expected in the case of the leftmost position 1 the weight for the left bi-phoneme distribution is one in the majority of cases whereas for the right position 3 the right bi-phoneme is of greatest importance. At position 2 the situation is mixed with more phonemes with a weight of almost one for the right bi-phoneme distribution.

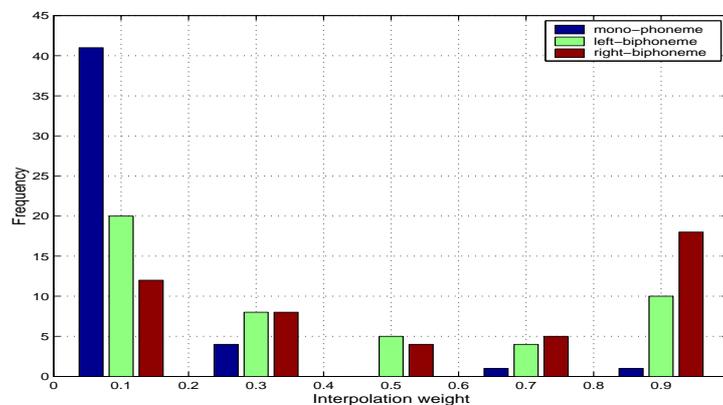
#### 5.4.1.3 Discounting methods

The third option to optimise smoothing is the proper choice of the discounting method. HMS-HMM distribution modelling is different to language modelling in that the number of potential events in each distribution is much smaller. Still the distributions are usually narrow and in practice make use of only a subset of the potential candidates. In many cases, a high frequency is observed for these cases. Still potentially unobserved events appear to be of importance for smoothing of MSM distributions.

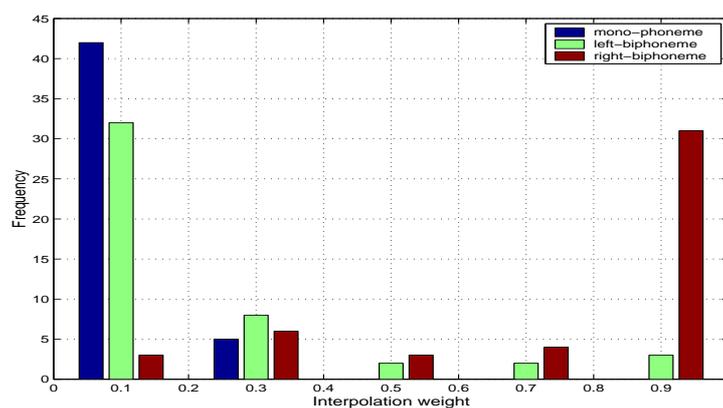
The different discounting methods under investigation, namely Witten-Bell, Good-Turing and absolute discounting, were presented in Section 5.1.2. Witten-Bell discounting decreases the discount factor with the number of elements observed whereas the Good-Turing discounting factor hinges on the number of events observed only once (singleton events). Absolute discounting uses a constant value to subtract from the event frequency. The discounted probability mass thus increases with the number of elements observed and decreases with the frequency of events. In conjunction with Witten-Bell and absolute discounting another heuristic smoothing



(a) position 1



(b) position 2



(c) position 3

Figure 5.4 Histograms obtained on RM over interpolation weights in each position within a phoneme, over 47 phonemes. Interpolation weights for the mono-phoneme, left and right bi-phoneme distributions (displayed in blue, green and red respectively) were estimated.

	Discounting	MTtest	WS96devsub	Average
HMM	-	43.68	46.32	45.04
HMS-HMM	Witten-Bell	42.99	44.11	43.57
HMS-HMM	Good-Turing	43.03	44.48	43.78
HMS-HMM	Absolute	43.32	44.67	44.01
HMS-HMM	Witten-Bell (ev1)	43.03	44.80	43.95
HMS-HMM	Absolute (ev1)	43.52	44.45	44.00

Table 5.10 %WER results on Switchboard. Comparison of discounting methods. (ev1) denotes the deletion of all events occurring once.

method has been tested. In these cases all events which occur only once are deleted from the model distribution. This should avoid spurious events and improve training.

Table 5.10 shows word error rate results on the Switchboard corpus for the baseline HMM model set and the discounting methods under investigation. All HMS-HMMs were initialised from the baseline model set and apart from the discounting method used an MSM setup as described in Section 5.4.1.1. The results indicate that the best performance is obtained with Witten-Bell discounting. The deletion of singleton events had a negative impact on performance. The results shown in Table 5.10 are consistent with observations made on RM.

#### 5.4.2 MSM Initialisation

In previous experiments HMS-HMM models and model sets for use in MSMs have been initialised by rearrangement of the baseline model parameters. In the first HMS-HMM training iteration a uniform distribution over all potential models is assumed. This is a computationally expensive procedure and it induces considerable change in the model sequences generated in subsequent training iterations as well. HMS-HMM parameter optimisation, like HMM parameter optimisation, does not guarantee to yield the globally optimal solution. Thus initialisation of model parameters is an important issue in training. Another option for the initialisation of model distributions is the reuse of information present in the fixed mapping of the phonetic decision tree associated with the baseline model set. This type of initialisation is of interest since it allows a closer comparison of HMS-HMMs with the HMM baseline.

The state clustered phonetic decision tree associated with the baseline HMM model represents a unique mapping from phoneme to model sequence and thus can be translated into a corresponding model sequence. The initial MSM is trained on that model sequence using the methodology described in 5.4.1.1. The first question is whether the unique mapping is now deeply encoded in the model distributions. If this is the case, decoding with the initialised MSM without re-estimation of HMM parameters should result in similar performance to the baseline system. In particular, a direct comparison can be made by restriction to the one model with the highest probability for each distribution. The results in Table 5.11 denoted with “single model” show an

	Reestimation	MTtest	WS96devsub	Average
HMM baseline	-	43.68	46.32	45.04
HMS-HMM baseine	-	42.80	43.92	43.38
HMS-HMM	no, single model	43.49	45.84	44.70
HMS-HMM	no	43.25	45.81	44.57
HMS-HMM	yes, with pruning	43.21	45.81	44.55
HMS-HMM	yes, no pruning	42.99	44.98	44.01

Table 5.11 %WERs on Switchboard. The MSM is initialised from the baseline HMM state clustered model sequence. Results are obtained with and without re-estimation of model parameters.

improvement in WER by 0.3% over the baseline system. This means that the backoff structure is a better predictor for the unseen tri-phonemes than the phonetic decision tree in this case. If standard distribution based pruning is used the error rate is approximately 0.5% absolute lower. Disappointingly further re-estimation steps with pruning at a 95% probability mass level do not yield an improvement. This type of MSM initialisation forces a too narrow distribution and leaves little room for improvement which is also reflected in an initial training set perplexity of only 1.025. If the pruning in training was discarded the smoothing techniques relaxed the constraints sufficiently to allow further improvement by 0.5% WER absolute. However the performance is still poorer than for the HMS-HMM baseline system.

Scale factor	MTtest	WS96devsub	Average
2.0	43.10	45.08	44.12
3.0	43.39	44.91	44.17
4.0	42.99	44.98	44.01
5.0	43.33	45.44	44.42

Table 5.12 %WER on Switchboard. Results obtained by consistent use of scale factor in training and decoding and decoding. The MSM is initialised from the baseline HMM state clustered model sequence.

However, the constrained MSM initialisation does allow a quicker search for the appropriate MSM scale factor. Table 5.12 shows word error rate results obtained after 5 iterations of training for different scale factors. Only small differences can be observed for scale factors below 5.0. Experiments so far have been using a scale factor of 4.0 for which the table indicates to be the lowest error rate. Since a higher scale factor implies a lower perplexity the highest possible scale factor was used.

### 5.4.3 Clustering of distributions

Experiments conducted in Section 5.4.1 were focused on the improvement in smoothing of model distributions. Another method for dealing with data sparsity, which is inherently different to smoothing is the use of parameter tying or bucketing. In construction of HMM sets for

speech recognition both methods have been tested (e.g. (Lee, 1990)) but parameter tying in the form of state clustering by use of phonetic decision trees has been most successful. In the case of state tying tri-phoneme contexts are grouped into disjoint sets which share one set of parameters, in this case the complete output distribution of a particular state. On top of this the phonetic decision tree provides a prediction for unobserved contexts.

In the case of the MSM equivalent to state-clustered triphone HMMs, clustering implies the collection of counts from multiple distributions dependent on different phoneme and phoneme position contexts. Two options were explored to arrive at the proper set of clustered contexts: the first options simply used the set of clustered tri-phoneme contexts used by standard HMMs. Since the state clustering procedure involves the use of a minimum frequency threshold sufficient observations should be available for training of each clustered distribution. Consequently no further smoothing of model distributions was used in this case. Since phonetic decision trees also cluster contexts unobserved in the training data no secondary model distributions were required.

	Clustering	Pruning	feb89	oct89	feb91	sep92	Average
HMM	yes	-	3.16	3.80	3.30	6.17	4.11
HMS-HMM	no	yes	2.62	3.20	2.54	4.81	3.30
HMS-HMM	yes	yes	2.62	3.35	3.22	6.92	4.02
HMS-HMM	yes	no	2.58	3.24	2.70	5.94	3.62

Table 5.13 %WERs on all RM test sets. Results for clustering (“bucketing”) of tri-phoneme distributions as used for the baseline HMM set. Pruning denotes the use of MSM pruning in training.

Table 5.13 shows results obtained on RM for the baseline HMM set and HMS-HMMs trained with clustered contexts. A uniform initialisation of model distributions was used. The normal training procedure for HMS-HMMs used so far prunes model distributions at a 95% level. The use of this procedure shows only minor improvement over the baseline HMM. However, the result can be substantially improved by using the complete distributions in training and thus shows a 0.5%WER absolute improvement over the baseline. However this is still 0.3% poorer than the result for the baseline HMS-HMM. Interestingly the difference appears to stem mostly from a substantial difference in performance on the most difficult test set sep92.

The second option is to perform the clustering of phoneme contexts on the basis of initial MSM distributions. Experiments using an agglomerative clustering scheme and a Kullback-Leibler (KL) based distance measure (Cover and Thomas, 1991) were conducted. In order to merge two model distributions with different contexts  $P_1(\mu)$  and  $P_2(\mu)$  the distance metric  $D(P_1, P_2)$  has to fall below a preset threshold. The distance measure is defined by the sum of KL distances between the merged distance  $P_m(\mu)$

$$P_m(\mu) = \frac{1}{N(\rho_1) + N(\rho_2)} (N(\rho_1)P_1(\mu) + N(\rho_2)P_2(\mu))$$

and the individual model distributions:

$$D(P_1, P_2) = d(P_1||P_m) + d(P_2||P_m)$$

Note that this distance measure is a symmetric function in  $P_i$ , but does not constitute a distance metric. For each set of contexts the distance between all distributions is computed and the distributions with smallest distance are merged. The process continues until the distance exceeds a certain threshold. In the same way as used in the HMM state clustered baseline, all contexts corresponding to a particular centre phoneme and phoneme position are allowed to be merged. Since there is no limit set on the minimum number of observations for a particular distribution, the smoothing scheme was kept in place whereby the backoff distributions have been clustered accordingly.

Threshold	%Dist retained	feb89	oct89	feb91	sep92	Average
-	100	2.62	3.20	2.54	4.81	3.30
0.01	88.75	2.54	3.43	2.78	4.65	3.35
0.02	82.48	2.62	3.50	2.66	4.88	3.42
0.05	68.27	2.46	3.54	2.74	5.08	3.46
0.1	52.26	2.77	3.28	2.74	5.04	3.46
0.2	34.32	2.85	3.39	2.90	4.96	3.53

Table 5.14 %WERs on RM. Results of clustering of tri-phoneme context distributions using a KL distance metric as a function of the clustering threshold. The second column shows the percentage of distributions retained.

Results on RM data in Table 5.14 are obtained by initialisation from the HMS-HMM baseline presented in Section 5.4.1.1. Experiments were conducted with different cluster distance thresholds. An increase in threshold implies that a lower number of distributions is used. The table shows the percentage of tri-phoneme distributions remaining in the respective MSM. The results indicate a slow degradation in performance with an increase in the clustering threshold. Note that the number of distributions in case of clustering with threshold 0.2 with 6336 is large compared to 1581 distributions in the case of initialisation from an HMM state clustered system<sup>14</sup>.

#### 5.4.4 HMM initialisation

In previous sections HMS-HMMs were initialised by using the HMM model parameters directly from the baseline model sets, which in most cases were standard state clustered HMMs. The clustering procedure in these cases however is based on maximisation of data likelihood rather than on a distance metric between the underlying model distributions. This has the effect of forming a strong link between the amount of data available for training and the number of models (see Figure 5.2). A different clustering technique based on model closeness rather than the

<sup>14</sup>Another set of experiments using an identical model setup was performed by including modelling of state skips as will be described in section 5.4.7. The results observed in that case showed a minor improvement of 0.04%WER absolute over the corresponding baseline in the case of using a clustering threshold of 0.01.

overall data likelihood was used to initialise the HMM parameters of HMS-HMMs. The implementation of such a scheme as provided in HTK was assumed to be suitable for this task (Young and Woodland, 1994; Young et al., 1999). The toolkit provides an implementation capable of agglomerative clustering of output distributions. In the case of single Gaussian distribution a Gaussian divergence related distance metric is used.

As in standard HMM training, 3 state models with a single Gaussian are trained for all phoneme triplets contained in the training set. The sets of output distributions for potential merging were again defined by the centre phoneme and position within a phoneme. Distributions associated with different contexts were merged until a certain threshold was reached. The threshold needs to be adjusted to obtain the desired total number of states. The sets of HMMs obtained for each centre phoneme and position were used as the set of models for all model distributions which have this particular combination as part of the context. Due to starting from single Gaussian models, the HMS-HMM training scheme has to be extended in stages which each perform a small increase in the number of mixture components. The MSM is left unchanged during this process<sup>15</sup>.

System	Discounting	#state	#mix	feb89	oct89	feb91	sep92	Average
HMM	-	1577	6	3.16	3.80	3.30	6.17	4.11
HMS-HMM	Good-Turing	1579	5	3.08	3.69	3.10	6.33	4.05
HMS-HMM	Witten-Bell	1579	5	2.54	3.35	2.66	6.02	3.65
HMS-HMM	Good-Turing	1172	6	3.63	3.18	3.38	6.37	4.13
HMS-HMM	Witten-Bell	1172	6	3.36	3.35	3.46	5.63	3.95

Table 5.15 %WERs on RM test sets. Initialisation based on model sets obtained by agglomerative clustering using a Gaussian divergence based metric. (#state) denotes the number of output distribution and (#mix) is the number of Gaussian mixture components used per speech model state. HMS-HMMs use either Witten-Bell or Good-Turing discounting and backoff to left bi-phoneme distributions only.

The initialisation scheme was tested both on RM and Switchboard data. Table 5.15 shows results for the baseline HMM set and for various HMS-HMM sets trained with Witten-Bell or Good-Turing discounting. Discounting was revisited due to a potential increase in spurious events. Two model sets for initialisation were investigated. The first set had approximately the same number of output distributions as the baseline HMM set. The second set uses 26% fewer distributions. The first observation is that Witten-Bell discounting again outperforms Good-Turing discounting. Secondly the HMS-HMM system with a comparable number of states could not be trained to 6 mixture components. The word error rate for this model set is 0.2% absolute higher than that for the HMS-HMM baseline-based initialisation (see Table 5.5). If the number of states is reduced substantially the HMS-HMM system still outperforms the baseline HMM model set with considerably fewer parameters.

In an identical procedure single density triphone models were estimated on the Switchboard

<sup>15</sup>Uniform initialisation at these stages gave poorer word error rate performance.

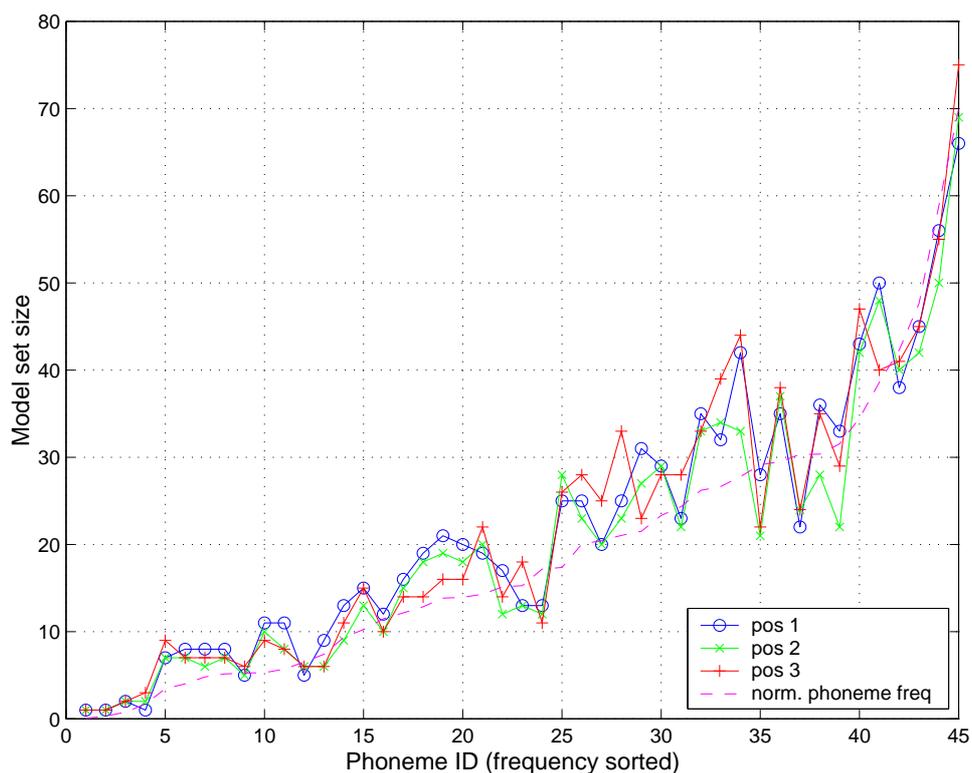


Figure 5.5 Sorted number of single state models after agglomerative clustering for each phoneme and each of the 3 phoneme positions. The 4th line shows the normalised frequency of phonemes using identical normalisation as in Figure 5.2.

MiniTrain training set and subsequently clustered using HTK clustering. The objective was to obtain approximately the same number of models as present in the baseline HMM or HMS-HMM systems. The model set used had 47 fewer states than the baseline system and the model set sizes obtained after the clustering procedure are presented in Figure 5.5. Not only is the correlation with the amount of training data for each phoneme higher than in the case of state clustering before (see Figure 5.2), but the variation between different position within a phoneme is similarly decreased.

Table 5.16 shows word error rates for the baseline HMM and HMS-HMM systems and two related HMS-HMMs based on the modified initialisation scheme. The first model set investigated "first training", the HMS system was initialised with approximately the same number of states as the reference system, but showed a higher error-rate even than the baseline HMM. In a second training run the reuse of the MSM obtained in the first training helped to improve performance. However, the model distributions remained relatively broad, resulting in low MSM likelihoods and a considerably higher perplexity of 2.50. Pruning these model distributions aggressively by retaining only 90% of the probability mass reduced the perplexity down to 1.95 which in turn is reflected in a considerable decrease in word error rate.

Given the results on RM the different initialisation has a tendency to produce more compact mixture densities. However, in the case of modelling Switchboard data the high variability and

	#states	#mix	MTtest	WS96devsub	Average
HMM	2957	12	43.68	46.32	45.04
HMS-HMM <sup>16</sup>	2957	12	42.88	44.70	43.82
First training	2910	12	45.18	46.15	45.68
Second training	2910	6	45.84	46.46	46.16
	2910	8	45.09	46.43	45.78
	2910	10	44.90	46.37	45.66
	2910	12	44.16	45.67	44.94
+aggressive pruning	2910	12	43.00	45.34	44.20

Table 5.16 %WERs on Switchboard. HMS-HMMs initialisation is based on model sets obtained by data driven triphone clustering. (#state) denotes the number of output distribution and (#mix) is the number of Gaussian mixture components used per speech model state.

confusability of this type of data induces a large perplexity and thus a considerable amount of confusability which cannot be sufficiently compensated.

#### 5.4.5 Soft-tying of states

The structure of HMS-HMMs based on fixed alignment between model and phoneme sequence can be interpreted as a hierarchical tying of mixture distributions. Thus an important question is if a straightforward tying of mixture components can achieve similar performance improvements on the recognition tasks investigated. One scheme for obtaining a suitably constrained tying of mixture distributions was presented by (Luo and Jelinek, 1999; Luo, 1998) which presented a scheme to improve data sharing in a non-reciprocal manner and thus effectively softening the hard decisions as to the appropriate state made in the early stages of model training. In (Hain et al., 2000) a simplified but approximately equivalent alternative solution to this scheme was tested on the Switchboard corpus. The scheme uses a distance metric based on the overlap of Gaussians (Povey and Woodland, 1999) to find a fixed number of closest neighbouring states for each state in the model set. Subsequently the mixture components of those neighbouring states are added to those already present. This scheme is non-reciprocal in the sense that the neighbouring states may not necessarily have that particular state in their list of nearest neighbours.

In order to fulfill the sum-to-one constraint for probability density functions the mixture component weights are normalised. This is achieved by the assumption of a certain weight of the original mixture distributions and equal sharing of the remaining weights among the new components. Table 5.17 shows results for initialisation with different initialisation weights. The optimal initialisation weight for the original mixture components was 0.7.

Table 5.18 shows the results for a variation in the number of states. Note that the optimal value was achieved by soft-tying of 3 or 4 states. The use of all available mixture components for a

<sup>16</sup>The HMS-HMM baseline system uses an MSM structure as described in 5.4.1.1, however the system was trained with a suboptimal MSM scale factor of 3.0. The experimental evidence however is assumed to remain valid.

Init Weight	feb89	oct89	feb91	sep92	Average
1.0	3.16	3.80	3.30	6.17	4.11
0.8	3.12	3.43	2.90	5.59	3.76
0.7	3.16	3.46	2.78	5.82	3.81
0.5	3.16	3.46	2.90	5.59	3.78
0.33	3.28	3.46	2.90	5.98	3.91

Table 5.17 %WERs for soft-tying on RM with the 3 closest neighbours under variation of the weight for the original mixture components.

#shared states	feb89	oct89	feb91	sep92	Average
1	3.16	3.80	3.30	6.17	4.11
2	3.01	3.54	3.10	6.17	3.96
3	3.16	3.46	2.78	5.82	3.81
4	3.08	3.46	2.74	5.94	3.81
5	3.05	3.73	3.10	5.43	3.83
6	3.12	3.73	3.14	5.78	3.95
7	2.85	3.99	3.22	5.67	3.94
max	3.71	3.95	4.23	6.10	4.49

Table 5.18 %WERs for soft-tying on RM with the 3 closest neighbours under variation of the number of shared states. A weight of 0.7 was used for the original mixture components.

particular centre phoneme and position showed poorer results than the baseline due to increased confusability. However the initialisation of the model parameters is potentially suboptimal. In the optimal case soft-tying brings a 7% relative reduction in word error rate.

	feb89	oct89	feb91	sep92	Average
HMM	3.16	3.80	3.30	6.17	4.11
HMM/ST	3.08	3.46	2.74	5.94	3.81
HMS-HMM	2.62	3.20	2.54	4.81	3.30
HMS-HMM/ST	2.58	2.91	2.42	4.49	3.10

Table 5.19 Combination of HMS-HMMs and soft-tying (ST) on RM.

Table 5.19 shows the combination of soft-tying of mixture components and HMS-HMMs with Witten-Bell discounting and backoff to interpolated left and right bi-phoneme distributions. The HMS-HMM setup performs significantly better than the soft-tied model set. Furthermore a combination of the two techniques was tested. Starting from the baseline HMS-HMM model set a total number of 3 states was tied using the scheme described previously. A further reduction by 0.2% absolute was observed. Clearly the gain from these methods is not completely additive but still the methods are complementary.

Again an equivalent set of experiments was conducted on the Switchboard corpus. The results

#shared states	MTtest	WS96devsub	Average
1	43.68	46.32	45.04
2	43.19	45.19	44.22
3	42.89	44.97	43.96
4	43.07	44.07	43.58
5	43.00	44.44	43.74
6	43.22	44.26	43.76
7	43.13	44.41	43.79

Table 5.20 *Soft-tying of mixture components on Switchboard. An initialisation weight of 0.7 was for the original mixture components*

for variation in the number of states shared presented in Table 5.20 show optimal performance for sharing of 4 states and a substantial improvement in word error rate by 1.46% absolute. A further increase in the number of states used for soft-tying only yields a moderate increase in word error rate. Table 5.21 shows a comparison between the baseline HMS-HMM system and the soft-tied model sets. Note that the gain on the WS96devsub set is substantially larger than that obtained on MTtest. The difference between the soft-tied HMM set and the HMS-HMM models is only 0.2% absolute. However the combination of HMS-HMMs with soft-tying yields a further improvement by almost 0.5% WER absolute. The difference between these results and those obtained on RM is substantial and can only be attributed to the much larger variability beyond strict dependence on phoneme context.

	MTtest	WS96devsub	Average
HMM	43.68	46.32	45.04
HMM/ST	43.07	44.07	43.58
HMS-HMM	42.80	43.92	43.38
HMS-HMM/ST	42.60	43.24	42.93

Table 5.21 *Combination of soft-tying (ST) of states and HMS-HMMs on Switchboard.*

#### 5.4.6 Pronunciation modelling

The selection of model sets, i.e. groups of models shared between certain sets of phoneme contexts, is a crucial element in the construction of model sequence models. In order to avoid confusability which cannot be controlled, the model sets have been constrained to belong to a certain phoneme and position within the phoneme. Given the substantial difference in performance gain between models trained for RM and models trained for Switchboard one important difference directly related to HMS-HMMs is the fact that a multiple pronunciation dictionary is used in the Switchboard setup whereas only one pronunciation is present in the dictionary used on RM. A set of experiments was designed to answer the question whether HMS-HMMs provide

a method that can overcome a deficiency in pronunciation modelling. Consequently this may result in better performance of HMS-HMMs when used with dictionaries containing only a single pronunciation for each word and further gain should be possible with the controlled sharing of models between different phonemes. (Saraçlar et al., 2000) successfully used a related scheme where the parameters from different model sets and associated with different phonemes are soft-tied and used in decoding. However initialisation in his work is based on manual labelling and the conceptual difference between surface and base-form pronunciations.

The RM system already makes use of only one pronunciation per word, and substantial improvement was obtained with HMS-HMMs in this setup. In order to extend the method to use models across different phonemes the following scheme was used for initialisation:

1. The sets of models as defined for the baseline HMS-HMM system are used for initialisation. Each model set is associated with a certain phoneme and is linked with all phoneme contexts that share this centre phoneme.
2. The models in each set are single state models. For each state in the model set a finite number of nearest states belonging to model sets outside the current set is determined. The Gaussian overlap distance measure (Povey and Woodland, 1999) is used for this purpose. The nearest state models are added to the current model set. This is similar to the soft-tying method presented in Section 5.4.5.
3. After this is done for each model set normal HMS-HMM training is resumed.

	Model sharing	MSM init	feb89	oct89	feb91	sep92	Average
HMM	-	-	3.16	3.80	3.30	6.17	4.11
HMS-HMM	within phoneme	uniform	2.62	3.20	2.54	4.81	3.30
HMS-HMM	between phonemes	uniform	2.30	3.02	3.14	4.57	3.26
HMS-HMM	between phonemes	weighted	2.38	3.06	2.13	4.53	3.03

Table 5.22 *Sharing of models in MSMs across different centre phonemes on RM. The weighted initialisation use 30% of the probability mass for the new models.*

Two experiments with different MSM initialisation have been conducted: in the first case the model distributions were initialised with a uniform probability distribution over all models within a model set; in the second case 30% of the total probability mass was discounted from the original model distribution (as described in step 1.) and was used for uniform distribution over the new model entries (as described in step 2.) in the model set. An equal amount was subtracted from the probability of each of the original models. Results in Table 5.22 indicate that uniform initialisation results in almost no difference in the overall word error rate. However, an improvement was made on the sep92 set which has the highest word error rate. Note that this procedure implies initialisation of the HMM parameters using the baseline HMS-HMMs. If

the same type of initialisation is applied without any additional models no difference in performance is observed. The results with weighted initialisation indicate a 9% relative improvement in word error rate over the HMS-HMM baseline. The dependence on the initialisation suggests that an arbitrary increase in the number of models per model set is undesirable. Even though the additional models differ more from the ones in the original set, the increase in confusability cannot be sufficiently constrained by uniform initialisation over all potential models.

	#PronVar	MTtest	WS96devsub	Average
HMM	multiple	43.68	46.32	45.04
HMM	single	43.90	45.82	44.89
HMS-HMM	multiple	42.80	43.92	43.38
HMS-HMM	single	41.82	44.33	43.12

Table 5.23 %WER results on Switchboard using models trained and tested with dictionaries containing one and multiple pronunciations (#PronVar).

In order to perform a similar set of experiments on Switchboard a dictionary using only a single pronunciation variant has to be constructed. This is done using pronunciation variant frequencies obtained by alignment of more than 265 hours of speech from the Switchboard and CallHome corpora. The detailed procedure is presented in Appendix B.1. Using the new pronunciation dictionaries for training a new baseline HMM model set was trained. This required re-clustering of the context dependent HMMs and consequently a new phonetic decision tree. Table 5.23 shows the results for the single pronunciation baseline system. Astonishingly the average word error rate obtained with only one pronunciation variant was lower than that obtained with multiple variants. In order to confirm this surprising result a series of experiments with larger training and test sets from the Switchboard and CallHome corpora was conducted and similar results have been obtained (see Appendix B.2). The use of pronunciation variants as used in the standard CU-HTK dictionary gives poorer performance than the use of a specific selection of these pronunciations. This differs from the expected scenario of a small loss in performance which could partly be regained by HMS-HMMs. Nevertheless HMS-HMMs initialised from the single pronunciation HMM set show a small improvement of 0.3% over the HMS-HMM baseline and thus perform slightly better with single pronunciation dictionaries. Note that the improvement originates from a difference by 1% absolute in the MTtest result.

	Model sharing	Init Weight	MTtest	WS96devsub	Average
HMS-HMM	within phoneme	-	41.82	44.33	43.12
HMS-HMM	between phonemes	0.3	42.69	44.29	43.52
HMS-HMM	between phonemes	0.5	42.74	44.11	43.45

Table 5.24 %WERs on Switchboard using HMS-HMMs which use models across different phonemes. Dictionaries with single pronunciation variants are used in training and test.

The single pronunciation models sets were extended using the scheme presented at the begin-

ning of this section. In contrast to results obtained on RM no performance improvement was observed (see Table 5.24). This result appears to be consistent with the poorer performance observed with modelling of pronunciation variation. Note that performance on WS96devsub remains at the same level whereas the word error rates for MTtest are back at the level obtained for multiple pronunciation variants.

#### 5.4.7 Model insertions and deletions

Model sequence models in theory allow arbitrary mappings between model and phoneme sequences. The theoretical background was discussed in detail in Section 4.2.2. In practice, the initialisation of such a scheme is of vital importance. Thus the 1:1 phone to model mapping was relaxed gradually by first introducing *skips*, i.e deletions of models and then insertions in a second stage.

Two different scenarios have been tested. The first set of experiments (Scenario A) was aimed at using multigrams in a straightforward fashion. For each phoneme context a large set of model strings was used. The model strings themselves consist of three single-state models. In a next stage the modelling of skips was implemented by adding all possible alternative models in which a single state was deleted. After several E-M training steps using a standard N-gram MSM, in the second stage the N-gram model was replaced by a multigram model. Sharing of states between adjacent phone contexts was allowed.

Initial model strings for Scenario A were obtained from a state-level HMS-HMM system. All state-model triplets within each triphone context were collected and added to a new HMS-HMM set. The HMS-HMM then operates with models on the phone level. Since the number of such triplets is large, sparsity problems require a considerable amount of pruning of model distributions. The sparsity of distributions is the major disadvantage throughout all stages of this scenario, which even required hard limits to be placed on the number of allowable alternative models during training.

To overcome the above difficulties, a second approach, Scenario B, tried to make more robust estimates of the model sequence probability by using MSMs at a state level. An explicit skip HMM (i.e. with no output distribution) was added to the list of possible models within each phoneme context. For this stage it is necessary to avoid the situation that skips are used for every position within a phone model. The standard left to right model topology had to be modified for this purpose. Figure 5.6 shows the modified topology. The model distributions M1, M2 and M3 represent the non-skip models for each position whereas SKIP1, SKIP2 and SKIP3 represent the skip in the corresponding distributions. Each of the nodes M1, M2 and M3 can hold multiple parallel models and the arcs in the graph denote a full interconnect, for example all models in M1 are connected with all models in M2.

A common initial value for the probability of the skip models in all contexts was chosen. After several HMS-HMM E-M iterations an estimate of the probability of skips in each phone context

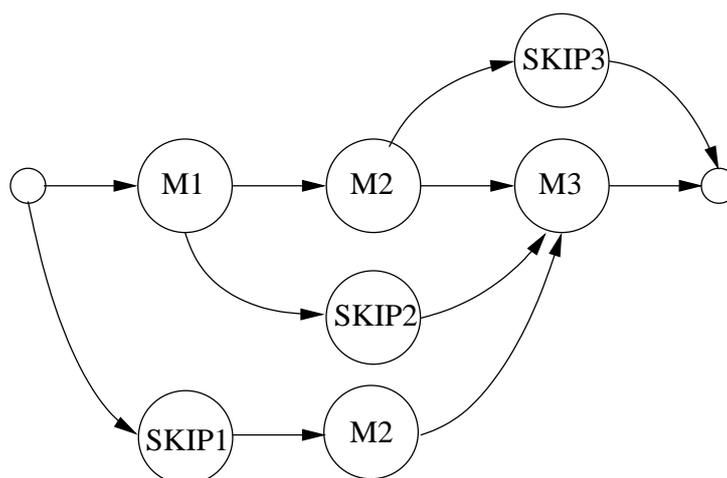


Figure 5.6 Modified model topology for the use of skip models.  $M1$ ,  $M2$  and  $M3$  denote sets of parallel models whereas  $SKIP1$ ,  $SKIP2$  and  $SKIP3$  represent model skips.

was obtained. Two different second stages were tested. In one case all pairs of models from the same phone position were added to the model distribution. This can be sufficiently modelled using N-grams. In the second case models from adjacent phone positions within a particular phone were added. After some re-estimation steps using N-grams, a multigram MSM with a maximum of two elements per model string was used.

All MSMs make use of smoothing by Witten-Bell discounting and backoff to interpolated left and right biphone models. This is also true for the use of multigrams where each set of distributions depending on context is re-estimated independently. In the case of the use of multigrams the HMS-HMM training procedure used one multigram E-M iteration within each overall re-estimation.

#### 5.4.7.1 Scenario A: Phone models

All insertion/deletion experiments used an existing HMS-HMM model set for initialisation. Thus all model sets in the experiments on a particular corpus have exactly the same number of states and mixture components per state.

	MTtest	WS96devsub	Average
HMM baseline	43.68	46.32	45.04
HMS-HMM <sup>17</sup>	42.88	44.70	43.82
HMS phone level	43.69	45.25	44.49
+ max model	43.36	44.63	44.01
+ deletions	43.41	44.38	43.91
+ multigram	43.14	44.17	43.67

Table 5.25 Scenario A: %WER on Switchboard using phone level HMS-HMMs with modelling of insertions and deletions. (max model) denotes a hard limit on the number of models per distribution.

Due to the problems of data sparsity, experiments for Scenario A have only been conducted on Switchboard. Table 5.25 shows word error rates for the baseline HMM and HMS-HMM systems and for the various stages within Scenario A. As can be seen, the step towards phone unit modelling brings a performance degradation which can be partly recovered by setting a hard limit to the maximum number of models per phone context (max model). The addition of deletion modelling brings only minor improvements and still has a poorer WER than the HMS-HMM baseline. The use of multigrams finally brings slightly better performance than the baseline HMS-HMM system.

#### 5.4.7.2 Scenario B: Position dependent modelling

Scenario B was assumed to be less vulnerable to data sparsity problems and thus was tested on both RM and Switchboard. Table 5.26 shows word error rates on RM for the baseline systems and for modelling of sub-phone deletions and insertions. Compared to the HMS-HMM baseline, based on Witten-Bell discounting and backoff to interpolated left and right bi-phoneme context distributions, a further slight improvement could be made by added skip models. However on this data the gain from added insertions was larger. Modelling of sub-phone insertions and deletions within the same phone position gave a 5.7% relative reduction in word error rate over the HMS baseline and thus a 24.3% relative reduction over the HMM baseline. Using skips brought a 0.09% absolute reduction both in word deletions and substitutions but an increase in insertions. Modelling insertions further reduced the number of deletions while other substitution and insertion errors remained virtually unchanged. Using models from adjacent phone positions gave an overall word error rate of 3.20% and poorer results if further extended to using multigrams. The most likely cause for this is the fact that model skips are used in all model sets and their use is not limited in the parameter re-estimation procedure. The result is an overestimated probability for the skip transitions.

	feb89	oct89	feb91	sep92	Average
HMM	3.16	3.80	3.30	6.17	4.11
HMS-HMM	2.62	3.20	2.54	4.81	3.30
+ skips	2.66	3.09	2.86	4.34	3.24
+ insertions	2.23	2.94	2.74	4.53	3.11

Table 5.26 Scenario B: %WER on RM using state level HMS-HMMs.

Table 5.27 shows Scenario B results on Switchboard. In contrast to the results on RM a further WER reduction by 0.8% can be achieved by the use of skip models. Whereas the number of word deletions remained approximately the same, most of the improvement stems from a reduced number of word substitutions.

Again in contrast to RM, insertions brought no significant reduction in word error rate, even though a considerable increase in acoustic log-likelihood was observed. The probable reason for

<sup>17</sup>The HMS-HMM baseline is based on a scale factor of 3.0 in contrast to the one presented in Section 5.4.1.1.

this is the broadness of the MSM distributions which could not be controlled even with stricter pruning. Thus an overall improvement of 2.4% WER absolute over the HMM constitutes the best result so far. Similarly to the situation on RM, the use of models from adjacent positions and multigrams actually gave considerably poorer performance.

	MTtest	WS96devsub	Average
HMM	43.68	46.32	45.04
HMS-HMM	42.80	43.92	43.38
+skips	42.14	43.11	42.64
+insertions	41.81	43.29	42.57

Table 5.27 Scenario B: %WER on Switchboard obtained by rescoreing of trigram lattices.

#### 5.4.8 HMS-HMM in combination with standard ASR techniques

Due to the complexity of state of the art speech recognition systems, the testing of a new technique in conjunction with other standard technologies may provide valuable insights into properties of the algorithms under investigation. Since word error rates are substantially higher for baseline HMM systems than they are for more complex multistage systems the operating point may change the behaviour and conclusions considerably. Thus a small set of experiments with HMS-HMMs with MSMs based on fixed alignment between model and phoneme sequence was conducted in conjunction with vocal tract length normalisation (VTLN) (Eide and Gish, 1996; Hain et al., 1999) and maximum likelihood linear regression (MLLR) (Leggetter and Woodland, 1995; Gales and Woodland, 1996). VTLN can be classified as a constrained version of speaker adaptive training whereas MLLR is a generic HMM adaptation technique to adjust the output distribution parameters to properties of the test data.

	VTLN	MLLR	MTtest	WS96devsub	Average
HMM	no	no	42.72	45.97	44.39
HMM	yes	no	38.76	39.01	38.89
HMM	yes	yes	36.84	36.24	36.53
HMS-HMM/skip	yes	no	37.82	37.23	37.51
HMS-HMM/skip	yes	yes	34.98	35.06	35.02

Table 5.28 %WERs on Switchboard for baseline HMM model sets and HMS-HMMs .

An HMS-HMM model set using a fixed alignment between model and phoneme sequences and modelling of skips was initialised from an HMM model set trained on VTLN normalised data. Table 5.28 shows the results for the baseline HMM trained on VTLN data. Note the exceptionally large difference between HMM model sets trained on VTLN data and the standard models. The performance difference of the HMS-HMM to the baseline HMM was 1.4% absolute which is a smaller improvement in absolute and relative terms than that obtained on non-VTLN data

(see Table 5.27). Both the baseline HMM and the HMS-HMM model parameters were adapted independently using one global speech transform<sup>18</sup>. The difference in word error rate between the HMM and HMS-HMM model sets became slightly larger after the use of MLLR. Using the same HMS-HMM models a similar result was obtained on the eval97sub test set (see Appendix A.2).

## 5.5 Summary

In this chapter the implementation issues for HMS-HMMs were discussed in detail in Section 5.1 and solutions for these issues were chosen and presented. The modelling of speech with HMS-HMMs requires the handling of data sparsity appropriately and special care was placed on this subject. A variety of options to deal with data sparsity have been presented: discounting and backoff; interpolation and deleted estimation; and clustering of model distributions. Furthermore two different initialisation mechanisms for each of the MSM and HMM part of an HMS-HMM system have been described and tested. HMS-HMMs were used to improve the modelling of pronunciation variation by either sharing of models across different phonemes to model substitutions on a sub-phone level or by direct modelling of insertions and deletions. A final series of experiments was designed to investigate the performance of HMS-HMMs in conjunction with other techniques. Most experiments were conducted on data from two independent corpora: Resource Management for investigations in a simple constrained environment in terms of type of speech, amount of data, vocabulary size, language model complexity and pronunciation modelling; and Switchboard to investigate performance behaviour on spontaneous speech and on a system with all properties of a modern speech recognition task.

A performance improvement with HMS-HMMs over standard HMM sets based on phonetically clustered states was found in all experiments on both corpora, given that initialisation and training were set to take care of the effective breadth of model distributions or equivalently perplexity of the model sequence on the training data. An implementation of HMS-HMMs, initialised from the baseline HMMs using relatively simple discounting and backoff schemes for smoothing outperforms the baseline systems by a statistically significant margin. Initialisation from either a baseline HMM or from scratch both results in better performance than the equivalent phonetically state clustered HMM. This is attributed to the fact that HMS-HMM do not require the assumptions necessary for construction of decision trees which are difficult to overcome (see Section 3.5). Thus HMS-HMMs pose a reasonable alternative to phonetic tying of HMM output distributions.

The smoothing scheme selected due to its simplicity and performance are Witten-Bell discounting and Katz backoff to interpolated sets of less refined distributions. Smoothing by backoff was found to outperform bucketing schemes and the particular discounting scheme showed lower word error rates compared to absolute and Good-Turing discounting. A deleted estima-

---

<sup>18</sup>A further transform is used to adapt the silence models.

tion scheme to reestimate the interpolation weights brought no further improvement over the original scheme.

HMS-HMMs in the configuration outlined can be interpreted as a specific framework of parameter sharing. The use of soft-tying approaches after training of state clustered HMMs were compared with HMS-HMMs. Whereas only little interference between the two techniques was found on RM, on Switchboard the difference in word error rate halved between soft-tied HMMs and soft-tied HMS-HMMs compared to the non-soft-tied version.

Modelling of pronunciation variation by modelling of sub-phone models between phonemes was found to be effective on RM, which uses a dictionary with only one pronunciation variant. The lowest word error rate of 3.03% over all four RM test sets in this thesis was achieved in that configuration which reflects a 26% relative reduction in word error rate over the HMM baseline. A similar setup on Switchboard however showed better performance than a version using multiple pronunciation variants. While in the former setup the HMS-HMM results were better, the sharing of models across phonemes was not beneficial. The surprising difference between setups using single and multiple pronunciations appears to be reflected in that behaviour.

Modelling of insertions and deletions with multigram MSMs showed only limited success. The major drawback of this type of modelling is the requirement to hypothesize a large number of potential models which then are reduced to a manageable number in training. This incurs substantial data sparsity problems which are difficult to control. In order to obtain better smoothing and thus improved performance, the step back to position dependent models had to be made, in which case the multigrams fail at modelling of state skips. However the introduction of skip models into N-gram models based on fixed alignment gave the lowest Switchboard word error rates in this thesis with a total reduction in word error rate by 2.4% absolute over the HMM baseline.

The use of HMS-HMMs was tested in conjunction with speaker adaptive schemes such as vocal tract length normalisation and adaptation using maximum likelihood linear regression on Switchboard. The difference between an HMS-HMM system and the HMM baseline on non-VTLN data was 1.7% absolute, whereas training and testing on VTLN data gave a 1.4% difference. After adaptation with MLLR the performance difference to the baseline was 1.5%.

---

## *Summary and conclusions*

---

This thesis has addressed the problem of pronunciation variation in the construction of large vocabulary speech recognisers based on hidden Markov models. In particular the transcription of spontaneous speech was addressed by the implementation of a scheme which accounts for variation at the sub-phone level. An additional knowledge source is added to the hierarchical structure as used by state of the art automatic speech recognisers. Appropriate methods for the joint optimisation with hidden Markov models, particular implementations and special cases were tested on two substantially different recognition tasks and the performance of the new modelling approach was analysed in detail and compared with state of the art techniques.

### **6.1 Review of the Work**

Most systems for automatic recognition of speech are based on hidden Markov models and the work presented in this thesis is no exception. Considerable research progress in this area together with a substantial increase in computational power available on relatively inexpensive computer platforms, has lead to work on more and more complicated tasks. Among other important topics the step towards transcription of more natural speaking styles has lead to an increased interest in modelling of pronunciation variation. Still word error rates in this area remain considerably above those obtained for speech recorded in more controlled environments.

In this thesis the novel framework of hidden model sequence modelling of speech based on hidden Markov models was presented. As described in Chapter 4 the framework allows arbitrary mappings between model and phoneme sequences. A stochastic model for this mapping, the model sequence model may be trained using a maximum likelihood criterion. In particular the E-M algorithm is used to obtain an iterative solution for the joint optimisation of the underlying HMM model set and the MSM parameters. The set of arbitrary mappings is divided into two cases: the alignment between model and phoneme sequence is a priori known or fixed; the case of variable alignment where both sequences may differ in length. In the case of fixed alignment the framework models substitutions of HMMs with arbitrary topology. In that sense

the modelling of pronunciation variation in the form of phone or sub-phone substitutions is possible and can be implemented in a joint optimisation scheme. The natural form of modelling the fixed alignment case is an N-gram type model using a constrained phoneme context. The case of variable alignment allows insertion and deletions for specific combinations of phonemes. The natural extension of the fixed alignment case is the use of multigrams. The latter allows the estimation of joint probabilities for arbitrary length sequences using the E-M algorithm. The necessary modifications and re-estimation formulae for this case have been presented. Apart from the modelling of substitutions, insertion and deletion effects are assumed to be an important factor in pronunciation changes for spontaneous speech. The focus in this thesis is on improved data driven sub-phone modelling of spontaneous speech. Depending on the specific modelling case and perspective, HMS-HMMs can be interpreted as a dynamic model selection technique, as a soft version of phonetically tied phone HMMs, as a maximum likelihood framework for topology construction or as a method for pronunciation modelling.

In Chapter 5 the implementational issues involved in construction of HMS-HMMs were presented. Since data sparsity is a problem at this level specific attention was given to smoothing and clustering techniques of discrete probability distributions with respect to the special requirements for HMS-HMMs. Experimental work is conducted on two different transcription tasks: Resource Management which contains read speech and has very low complexity in terms both in terms of acoustics and semantics; and transcription of conversational telephone speech using a subset of the Switchboard corpus. The latter constitutes a complex task with exceptionally high word error rate results with state of the art modelling approaches. Word error rate results are presented for both corpora, if either consistence or difference between the tasks is of importance.

In case of fixed alignment modelling a range of smoothing techniques well known in language modelling such as discounting and backoff schemes was tested and compared with schemes using interpolation by deleted estimation and clustering. Furthermore the fact that the E-M algorithm does not guarantee the globally optimal model parameters requires an analysis of appropriate initialisation of each set of model parameters involved. Alternative initialisation schemes of both MSM and HMM parameters have been tested. The most successful scheme obtained on both corpora is based on discounting and backoff to less refined discrete probability distributions over HMMs. In this case a reduction in word error rate of 0.81% absolute or 19.7% relative was obtained on RM. On Switchboard data an improvement of 1.7% absolute or 3.7% relative was obtained. In both cases the difference to the baseline systems was statistically significant. Modelling of variable alignment was addressed in two scenarios in order to deal with weaknesses of multigram modelling. On RM the modelling of deletions brought virtually no difference in performance whereas in conjunction with insertions a further reduction in word error rate by 6% relative was obtained. Experiments on the Switchboard corpus revealed a further decrease in word error rate by 0.7% absolute by modelling of deletions. In this case the introduction of potential sub-phone insertions into the MSM had a negative impact on performance.

Experiments conducted to share models between different dictionary phonemes were shown to perform well in the clean speech environment of RM, however surprising results on the Switchboard corpus revealed that pronunciation modelling both by use of pronunciation variants in the dictionary or by use of HMS-HMM for cross-phoneme modelling gives poorer performance than the respective baselines. The use of HMS-HMMs in conjunction with state of the art adaptive techniques such as vocal tract length normalisation or speaker adaptation using maximum likelihood linear regression showed only a small effect on the performance gain obtained with HMS-HMMs.

## 6.2 Suggestions for future work

The experimental evidence presented in this thesis gives indication that the selection of appropriate HMMs or HMM states in the case of modelling of spontaneous speech on the sole basis of a phonemic representation of words is insufficient. Hence improvement of modelling of this data must include different and/or additional information about the speech to be modelled. The generic framework of HMS-HMMs can provide a suitable framework for work in this direction. One option is to exchange the phonemic transcription by more varied and flexible representations such as phonological features or articulator positions. Interdependence of neighbouring units can be modelled with HMS-HMMs and appropriate smoothing techniques. However the data sparsity issues will require both low dimensional context and model spaces. Another natural option is to include additional information such as local estimates for the signal-to-noise ratio or for the speaking rate into the decision process. Since the latter is often modelled by use of separate models operating in parallel HMS-HMMs can provide a suitable framework.

Work on spontaneous speech revealed the considerable diversity in the data assigned to models of a certain set for which the phoneme context appears to be an imperfect predictor. Apart from better discrimination between models by inclusion of additional knowledge another choice is to decrease confusability overall. Recently the use of discriminative training was shown to yield substantial improvements in modelling of data from the Switchboard and CallHome corpora. Conversely a combination of HMS-HMMs with discriminative schemes is desirable, either in training of model parameters or in the use of discriminative information to adjust the sets of jointly used models.

In order to achieve optimal performance on a specific task it is necessary to obtain sufficient speech data for that particular task, both for acoustic and language modelling purposes. However it is practically and commercially highly undesirable to collect and transcribe substantial amounts of speech data in order to be able to train models for that task. Thus a reuse of speech data from existing sources is highly desirable. HMS-HMMs can be used for combination of models from different sources. The appropriate balance between models from an initialisation source and the data from the designated tasks will be adjusted automatically during training.

The computational cost of recognition using HMS-HMMs is substantially larger than for standard HMMs. Therefore suitable methods for efficient compression of model networks are required to

enable the use of HMS-HMMs in single pass decoding of utterances in conjunction with a large vocabulary and complex language models.

### 6.3 Conclusion

Modelling of speech using increased flexibility on a sub-phone level was shown to consistently improve performance on two different tasks and in conjunction with state of the art adaptation techniques. In the case of modelling of spontaneous speech evidence of sub-phone deletion effects was found in addition to substitutions. However experiments suggest that the increased variability observed in spontaneous speech is not solely determined by a phonemic transcription. Modelling of pronunciation variation on spontaneous speech data beyond within-phoneme variation failed in the given setup.

HMS-HMMs compare favourably with phonetic decision trees for clustering and prediction of HMMs or HMM states. In contrast to stringent assumptions about alignments and model complexity only weak theoretic assumptions are required for formulation of the maximum likelihood based optimisation scheme of HMS-HMMs. The novel approach consistently outperforms HMM model sets based on phonetic decision trees. HMS-HMMs provide a sentence model without any additional prior human knowledge and can be simply extended to include standard pronunciation modelling techniques.

---

## *Speech recognition task descriptions*

---

### **A.1 Resource Management**

The Resource Management (RM) corpus consists of read queries on the status of Naval resources and was specifically designed as a training and test corpus for U.S. DARPA contractors (Price et al., 1988). The task is artificial in many aspects such as speech type, range of vocabulary and grammatical constraint. The training set consists of 3990 utterances spoken by 109 speakers corresponding to approximately 3.8 hours of speech. There exist four test sets specified by the date when they served as test sets of speech recognition evaluations conducted by DARPA, namely February 1989 (feb89), October 1989 (oct89), February 1991 (feb91) and September 1992 (sep92). Each of these test sets consists of 300 utterances spoken by 10 speakers. The test sets contain 1.1 hours of speech in total and the complete training and test data is provided with a sample rate of 16kHz and a linear resolution of 16 bit. The test material is completely covered by a word-pair grammar included in the task specification which is used for recognition. The vocabulary size is artificially set to 997. The dictionary used here, i.e. the translation into pronunciation patterns, was produced by CMU (Lee, 1989) and contained one pronunciation per word. The dictionary uses 47 phonemes which is only a small subset of the set of phonetic symbols defined in the International Phonetic Alphabet (IPA). The symbols and examples of their use are contained in Table A.1.

The setup obtained as baseline for this task is based on work presented in (Young et al., 1994) made use of a of standard 3 state left to right model topology. A special topology for modelling the plosives /pd/, /kd/, /td/ and /dd/ employing a potential skip of the final state was used. Since work in this thesis required to split up baseline HMMs a uniform topology for all phone models was desirable. The skip was deleted and no impact on word error rate was found. Table A.2 compares %word error rates obtained by decoding of all four test sets with state clustered triphone models, with or without skip of the final state:

---

<sup>1</sup>Only used in conjunction with td.

Phoneme symbol	Example word	Phoneme symbol	Example word
aa	are	k	can
ae	and	kd	October
ah	one	l	long
ao	four	m	mile
aw	now	n	no
ax	as	ng	long
ay	by	ow	go
b	bay	oy	deployed
ch <sup>1</sup>	each	p	ship
d	day	pd	update
dd	add	r	red
dh	the	s	sea
dx	forty	sh	show
eh	any	t	time
el	level	td	at
en	reasons	th	third
er	after	ts	its
ey	April	uh	full
f	food	uw	cruiser
g	get	v	ever
hh	how	w	way
ih	did	y	year
iy	be	z	as
jh	June		

Table A.1 *Phoneme symbols and example words for the RM dictionary.*

	feb89	oct89	feb91	sep92	Average
Plosives with skips	3.05	3.80	3.46	6.14	4.11
Plosives without skips	3.16	3.80	3.30	6.17	4.11

Table A.2 *%WERs on RM comparing HMMs with different topologies for plosives.*

## A.2 Switchboard

The target of the Switchboard corpus collection was to provide a large speech corpus of conversational telephone speech to be used for speech recognition and speaker identification (Godfrey et al., 1992). The complete Switchboard corpus covers more than 2500 conversations from 500 U.S. speakers. The speakers are unknown to each other and are asked to converse on a certain topic which sometimes is ignored. Due to the telephone channel the data is recorded in stereo mode with a sample rate of 8kHz and is  $\mu$ -law encoded with a resolution of 8 bits per sam-

ple. An echo cancellation algorithm has been consistently applied to this corpus, however with mixed effect. Noise other than in the form of bursts generated by the speaker or other persons close to the conversation is generally not assumed to be a problem. The data was manually segmented and more than 250 hours of speech were transcribed manually by the Linguistic Data Consortium. Due to the inherent difficulty even in manual transcription of this type of data (see Section 3.2) many research sites have created their own segmentations and corrected versions of the training data transcriptions. More recently there has been an effort to consistently re-transcribe and re-segment the data by ISIP at Mississippi State University (Shaffer and Picone, 1998), however work on this labelling was completed only recently. Thus for experiments in this thesis the labelling from another source was required. In 1997 BBN defined a subset of the full corpus called MiniTrain and an associated test set MTtest (Miller and McDonough, 1997). The Switchboard corpus is the major data source for systems participating in the Hub5 LVCSR evaluations held by the U.S. National Institute for Standards and Technology and the particular data sets have been used during the development of the 1998 CU-HTK system (Hain et al., 1999; Hain and Woodland, 1998) and were consistently used in the experimental sections of this thesis for training and testing. The MiniTrain set covers 26904 utterances from 204 speakers and a total of 17.86 hours of speech. The MTtest set contains 701 utterances and 0.54 hours of speech from 14 speakers. In addition to this test set a subset of the test set used in the 1996 JHU Summer Workshop (Weintraub et al., 1996a) was used. The WS96devsub set contains of 767 utterances and 0.52 hours of speech from 11 speakers. The training and test set speakers do not overlap.

The setup used for experiments in this thesis makes use of a backoff trigram language model generated for participation in the 1997 Hub5E evaluations (Woodland et al., 1997b) and trained on approximately 2.3 million words from Switchboard and about 0.2 million words from the related but much smaller CallHome English corpus. The speech data contains hesitations and false starts in significant proportions. Both effects are labelled in the training and test data. Whereas entries in the vocabulary for hesitations are used both in training and test false start are only present in the dictionary used for training. The dictionaries contain multiple pronunciations per word and are based on the LIMSI'93 WSJ dictionary (Gauvain et al., 1994) which uses a set of 45 phonemes (see Table A.3).

Experiments conducted in Appendix B.2 make use of more extended training sets which also include data from the CallHome corpus. This corpus is similar to the Switchboard corpus but removes the constraint of a specific topic and speakers know each other. The results are an even higher word error rate than that observed for Switchboard data. However the amount of CallHome data is small with fewer than 20 hours of speech available for training. The *h5train00* training set consists of 265 hours of speech and covers almost the complete Switchboard corpus. The 68 hour *h5train00sub* set covers almost all the speakers present in *h5train00*. The *eval98* test set is the official 1998 Hub5E evaluation set, whereas the *eval97sub* set is a subset of the 1997 Hub5E evaluation set. For a more detailed information the reader is referred to the description in (Hain et al., 2000).

Phoneme symbol	Example	Phoneme symbol	Example
aa	are	iy	be
ae	and	jh	June
ah	but	k	can
ao	four	l	long
aw	now	m	mile
ax	ago	n	no
axr	were	ng	long
ay	by	ow	go
b	bay	oy	deployed
ch	each	p	ship
d	day	r	red
dh	the	s	seq
eh	any	sh	show
el	level	t	time
em	item	th	third
en	didn't	uh	full
er	bird	uw	cruiser
ey	April	v	ever
f	food	w	way
g	get	y	year
hh	how	z	as
ih	did	zh	asia
ix	aging		

Table A.3 Phoneme symbols and example words for the Switchboard dictionary.

## *Single Pronunciation Dictionaries*

---

The standard setup for a system for the transcription of conversational telephone speech makes use of dictionaries with multiple pronunciation variants per word (see (Strik and Cucchiarini, 1999)). On average 1.1 to 1.2 variants per word are used. These include variants where the change in pronunciation implies a change in meaning, but the variants mostly describes variation in accent. Many attempts have been made to improve pronunciation modelling by an increase in the number of pronunciation variants (e.g. (Byrne et al., 1998; Saraçlar, 2000; Riley et al., 1999)). New experiments in this thesis were conducted to answer the question if the use of single pronunciation dictionaries could improve HMS-HMM performance. In this context dictionaries with a single pronunciation per word had to be constructed. The following section describes the procedure how those dictionaries were created. Since the experimental results on the MiniTrain training set presented in Section 5.4.6 surprisingly suggest that equal performance can be obtained with single pronunciation dictionaries the dictionaries have been put to test on more extensive Hub5E training and test sets (see Appendix A.2). These experiments are described in the subsequent section.

### **B.1 Construction of single pronunciation dictionaries**

In this section the construction of a pronunciation dictionary with one pronunciation per word as used in the following section is described. The procedure assumes that a dictionary with a low number of pronunciation variants exists. Furthermore a model set trained using the multiple pronunciation variant dictionary and a training set with reasonable coverage of the vocabulary is required.

The algorithm is based on the frequency of pronunciation variants obtained during alignment of the training data. This makes the implicit assumption that pronunciation variants are only important for fairly frequently used words. The following sequence of steps was taken:

1. Pronunciation variant frequency

The training data is aligned to determine the pronunciation variant used for each word in the training data.

## 2. Initial dictionary

A dictionary with pronunciation variants sorted according to frequency is created. If a word was observed in the training data all variants which have not been observed are removed.

## 3. Merging of phoneme substitutions

For each word those pairs of variants which can be described by the substitution of a phoneme are merged. Starting from the most frequent variant another variant with phoneme substitutions only is searched. This procedure uses straightforward DP alignment. The variant with the higher frequency is kept but the frequency of the second variant is added. This procedure is continued until only variants exist which differ at least by one insertion or deletion of phonemes. In this stage a list of potential substitution phonemes for each phoneme plus the frequency of occurrence is generated.

## 4. Unseen words in the training data

A similar procedure as in step 3 is taken. If a substitution pair is found a score for the substitution is computed by summing of frequencies of the substitution contained in the lists generated in step 3. This metric is non-symmetric and thus gives a preference for one particular variant. The remaining procedure is identical to step 3.

## 5. Random deletion

If more than one pronunciation variant for a particular word remains the most frequent one is chosen. If the word was not seen in the training data this selection is random.

This algorithm clearly relies on a considerable vocabulary overlap between training and test sets and is far from optimal in other cases. If this would not have been the case a more elaborate selection of variants by estimation of a pronunciation variant probability on the basis of a simple N-gram model would have been chosen.

## B.2 Experiments

Surprisingly initial experiments with single pronunciation dictionaries on the Switchboard corpus using the MiniTrain training set and the MTtest and WS96devsub test sets gave a small improvement in performance. Since this result was unexpected, experiments on larger data sets to verify these results were conducted. This was done by using the infrastructure produced for participation in the March 2000 Hub5E NIST evaluations (Martin et al., 2000). For a detailed explanation of the underlying test and training sets and algorithms the interested reader is referred to (Hain et al., 2000).

Pronunciation variants	eval97sub			eval98		
	Swbd	CHE	Average	Swbd	CHE	Average
multiple	38.7	53.5	46.0	44.0	49.0	46.5
single	38.3	53.5	45.8	43.2	48.0	45.6

Table B.1 %WERs with dictionaries with multiple or single pronunciation variants obtained on a subset of the 1997 evaluation set (*eval97sub*) and the complete 1998 Hub5E evaluation set (*eval98*). The models were trained on 68 hours of speech (*h5train00sub*).

The second set of experiments was focused on the use of larger training and test sets. The 68 hour training set *h5train00sub* was used to train standard HMM model sets with different dictionaries. Two test sets, a 0.89 hour subset of the 1997 Hub5E evaluation set (*eval97sub*) and the complete 3.42 hour 1998 evaluation set (*eval98*) were used for testing. For training of both model sets the phonetic decision trees have been rebuilt and the system was trained with the corresponding dictionary and used the standard CU-HTK methodology of incremental increase of mixture components (Young et al., 1999). Table B.1 shows that for both test sets the word error rate is lower when only one pronunciation per word was used.

Pronunciation variants	eval97sub			eval98		
	Swbd	CHE	Average	Swbd	CHE	Average
multiple	37.7	52.7	45.2	42.7	47.6	45.2
single	38.1	52.3	45.1	42.4	47.3	44.9

Table B.2 %WERs on the *eval97sub* and *eval98* test sets using dictionaries containing pronunciation variant probability estimates. Models are trained on 68 hours of speech (*h5train00sub*).

In (Hain et al., 2000) the use of pronunciation probabilities was shown to bring significant improvements in word error rate. An important proportion of the improvement stems from the inclusion of silence models into the pronunciation variant for probability estimation. One effect of pronunciation probabilities is to limit the effective number of pronunciation variants. This approach may be viewed as a soft decision version of a single pronunciation dictionary. In order to make a fair comparison between a multiple pronunciation dictionary with probabilities and the single pronunciation dictionary the latter has to include probabilities for the silence models on a per word basis. A set of experiments including pronunciation probability estimates is presented in Table B.2. The performance improvement is smaller on both test sets but still the single pronunciation dictionary performs better. Note that the difference in word error rate on *eval98* between multiple and single pronunciation dictionaries as shown in Table B.1 is substantially smaller when using pronunciation probabilities. Nevertheless the use of the single pronunciation dictionary yields lower word error rates on both test sets.

The same set of experiments was conducted on an even large training set consisting of 265 hours of speech (*h5train00*). Table B.3 shows word error rate results with and without the use of pronunciation probabilities. In the case of the *eval97sub* test set the single pronunciation

Pronunciation variants	eval97sub			eval98		
	Swbd	CHE	Average	Swbd	CHE	Average
multiple	36.4	52.5	44.4	42.6	48.6	45.6
single	36.4	52.0	44.1	42.1	46.5	44.3

(a)

Pronunciation variants	eval97sub			eval98		
	Swbd	CHE	Average	Swbd	CHE	Average
multiple	35.8	51.3	43.5	41.2	46.9	44.0
single	36.3	51.4	43.8	41.2	45.9	43.5

(b)

Table B.3 %WERs on the eval97sub and eval98 test sets using dictionaries with multiple or single pronunciation variants (a). Results shown in (b) were obtained using pronunciation probability estimates. Models are trained on 265 hours of speech (h5train00).

dictionary performs better in the standard case but shows a higher word error rate when using pronunciation probabilities. Comparing to the results for eval98 on the smaller training set the difference in word error rate between the single and multiple pronunciation dictionaries is increased, both with and without the use of pronunciation probabilities. The lowest word error rate was obtained using a single pronunciation dictionary with probabilities for the silence variants within each word. Note that the eval97sub test set is substantially smaller than the eval98 test set.

The set of experiments presented in this section confirm the results obtained on a much smaller scale (see Section 5.4.6): the use of specifically constructed dictionaries with only one pronunciation variant per word does not give poorer performance on this type of data. However it must be noted that the reduction can have side effects on the performance of other techniques.

## Interpolation of model distributions

---

The estimate for the probability of a certain model  $\mu$  in a phoneme context  $\rho$  is given by the interpolation of  $I$  probability distributions  $P_i(\mu|g_i(\rho))$  with the interpolation weights  $\lambda(i|h(\rho))$

$$P(\mu|\rho) = \sum_{i=1}^I \lambda(i|h(\rho)) P_i(\mu|g_i(\rho))$$

where  $g_i(\rho)$  and  $h(\rho)$  are context mapping functions which for example map a tri-phoneme context to its associated mono-phoneme context. Note that the inverse  $g_i^{-1}(\bar{\rho})$  denotes the set of all triphones which have a specific mono-phoneme  $\bar{\rho}$  as the functional result. The context  $\rho$  must be the context with the widest span and is further called the *principal context*. All other distribution contexts will be called *secondary or derived contexts*. The interpolation weights may be directly context dependent or may as suggested by (Jelinek and Mercer, 1980) depend on the context frequency  $N(\rho)$ .

In order to maximise the data likelihood the E-M algorithm is used in a similar fashion as for parameter estimation of Gaussian mixture models. The hidden data will be the selector sequence  $\mathbf{K}$  which selects the appropriate distribution at each time instance. Using the formulation of the auxiliary function in Equation 4.4 the complete data is the set of the model, phoneme and the selector sequence  $x = (\mathbf{M}, \mathbf{R}, \mathbf{K})$  whereas the observed data is the pair  $y = (\mathbf{M}, \mathbf{R})$ . The auxiliary function with the initial and re-estimated parameters  $\theta$  and  $\hat{\theta}$  respectively can be formulated in the following way:

$$\begin{aligned} Q(\hat{\theta}, \theta) &= P(\mathbf{M}, \mathbf{R}|\theta) E\{\log P(\mathbf{M}, \mathbf{R}, \mathbf{K}|\hat{\theta})|\mathbf{M}, \mathbf{R}, \theta\} \\ &= \sum_{\mathbf{K}} P(\mathbf{M}, \mathbf{R}, \mathbf{K}|\theta) \log P(\mathbf{M}, \mathbf{R}, \mathbf{K}|\hat{\theta}) \\ &= \sum_{n=1}^{L_M} \sum_{i=1}^I P(i|m_n, \mathbf{r}_n, \theta) \log \left( P(i, m_n|\mathbf{r}_n, \hat{\theta}) \right) \end{aligned}$$

where  $\mathbf{r}_n$  is the context tuple at time  $n$  (e.g. triphone). The above equation used the fact that distributions are independent of each other and that the prior phoneme sequence distribution is constant during optimisation. The probabilities in the above equation can be computed by

$$P(i, \mu | \rho) = \lambda(i|h(\rho))P_i(\mu|g_i(\rho))$$

and

$$P(i|\mu, \rho) = \frac{\lambda(i|h(\rho))P_i(\mu|g_i(\rho))}{\sum_{j=1}^I \lambda(j|h(\rho))P_j(\mu|g_j(\rho))}$$

For optimisation the boundary conditions for the interpolation weights and the probability distributions need to be included by using Lagrange multipliers.

$$\sum_{i=1}^I \lambda(i|\bar{\rho}) = 1 \quad \forall \bar{\rho}$$

$$\sum_{\mu \in \mathcal{M}(\bar{\rho})} P_i(\mu|\bar{\rho}) = 1 \quad \forall (i, \bar{\rho})$$

Note that the phoneme context  $\bar{\rho}$  has exactly the structure required for the corresponding entity, e.g. if interpolation weights are mono-phoneme dependent there are exactly as many boundary constraints for the interpolation weights as there are mono-phonemes. Using the above equations the optimisation function may be written as

$$\begin{aligned} & \sum_{\mu \in \mathcal{M}(\bar{\rho})} \sum_{\rho \in \mathcal{R}} N(\mu, \rho) \cdot \frac{\lambda(i|h(\rho))P_i(\mu|g_i(\rho))}{\sum_{j=1}^I \lambda(j|h(\rho))P_j(\mu|g_j(\rho))} \cdot \log \hat{\lambda}(i|h(\rho)) \hat{P}_i(\mu|g_i(\rho)) - \\ & - \sum_{\forall \bar{\rho} \in \mathcal{H}: \rho = h^{-1}(\bar{\rho})} \eta_{\bar{\rho}} \left( \sum_i \hat{\lambda}(i|\bar{\rho}) - 1 \right) - \sum_{i=1}^I \sum_{\forall \bar{\rho} \in \mathcal{G}_i: \rho = g_i^{-1}(\bar{\rho})} \left( \nu_{\bar{\rho}, i} \left( \sum_{\mu} P_i(\mu|\bar{\rho}) - 1 \right) \right) \end{aligned}$$

The standard procedure, setting the derivative of the above function with respect to the parameters to zero allows to compute the Lagrange auxiliary variables and in succession the re-estimation formulae. The detailed steps are left for the reader. The estimates for the interpolation weights can be computed by

$$\hat{\lambda}(i|\bar{\rho} = h(\rho)) = \frac{1}{N(\cdot, \bar{\rho})} \sum_{\rho \in h^{-1}(\bar{\rho})} \sum_{\mu \in \mathcal{M}(\bar{\rho})} \left( N(\mu, \rho) \cdot \frac{\lambda(i|\bar{\rho})P_i(\mu|\rho)}{\sum_j \lambda(j|\bar{\rho})P_j(\mu|\rho)} \right)$$

and the adjustment of the probability estimates can be computed by

$$\hat{P}_i(\mu|\bar{\rho} = g_i(\rho)) = \frac{\sum_{\rho \in g_i^{-1}(\bar{\rho})} \left( N(\mu, \rho) \frac{\lambda(i|h(\rho))P_i(\mu|\bar{\rho})}{\sum_j \lambda(j|h(\rho))P_j(\mu|\rho)} \right)}{\sum_{\rho \in g_i^{-1}(\bar{\rho})} \sum_{\mu \in \mathcal{M}(\bar{\rho})} \left( N(\mu, \rho) \frac{\lambda(i|\rho)P_i(\mu|\bar{\rho})}{\sum_j \lambda(j|\rho)P_j(\mu|\rho)} \right)}$$

Note that the formulae are similar for both sets of parameters. In both cases the re-estimation can be implemented by accumulation of the posterior probabilities over the supporting set of principal contexts and normalisation to satisfy the sum-to-one constraints. In practice far fewer interpolation weights are computed on a smaller held-out training set. (Jelinek, 1997) suggests that a subset of the training set for estimation of the probability distributions may be used.

---

## Bibliography

---

- Bacchiani, M., Ostendorf, M., Sagisaka, Y., and Paliwal, K. (1996). Design of a speech recognition system based on acoustically derived segmental units. In *Proc. of ICASSP'96*, volume 1, pages 443–446.
- Bahl, L. R., Bellegarda, J. R., de Souza, P. V., Gopalakrishnan, P. S., Nahamoo, D., and Picheny, M. A. (1991). A new class of fenonic Markov models for large vocabulary continuous speech recognition. In *Proc. of ICASSP'91*, volume 1, pages 177–200.
- Bahl, L. R., Brown, P. F., de Souza, P. V., and Mercer, R. L. (1986). Maximum mutual information estimation of hidden Markov model parameters for speech recognition. In *Proc. of ICASSP'86*, volume 1, pages 49–52.
- Bahl, L. R., Brown, P. F., de Souza, P. V., Mercer, R. L., and Picheny, M. A. (1988). Acoustic markov models used in the Tangora speech recognition system. In *Proc. of ICASSP'88*, pages 497–500.
- Bahl, L. R., Jelinek, F., and Mercer, R. L. (1983). A maximum likelihood approach to continuous speech recognition. *IEEE Trans. PAMI*, 5:179–190.
- Baker, J. K. (1975). Stochastic modelling for automatic speech understanding. In Reddy, R., editor, *Speech recognition*, pages 512–542, New York. Academic Press.
- Baum, L. E. and Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model of ecology. *Am. Math. Soc. Bull.*, 73:360–362.
- Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Stat.*, 41(1):164–171.
- Bellegarda, J. R. and Nahamoo, D. (1990). Tied mixture continuous parameter modeling for speech recognition. *IEEE Trans. ASSP*, 38(12):2033–2045.

- Bernstein, E., McAllaster, D., Gillick, L., and Peskin, B. (2000). Recognising call-center speech using models trained from other domains. In *Proc. 2000 NIST Speech Transcription Workshop*, College Park, Maryland.
- Billa, J., Colthurst, T., El-Jaroudi, A., Iyer, R., Ma, K., Matsoukas, S., Quillen, C., Richardson, F., Siu, M., Zavagliagos, G., and Gish, H. (1998). Improvements to Byblos 1998. Presented at the 9<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.
- Bimbot, F., Pieraccini, R., Levin, E., and Atal, B. S. (1995). Variable-length sequence modelling: Multigrams. *IEEE Signal Processing Letters*, 2(6):111–113.
- Boulevard, H. (1995). Towards increasing speech recognition error rates. In *Proc. of EURO-SPEECH'95*, pages 883–894.
- Boulevard, H. and Morgan, N. (1994). *Connectionist speech recognition*. Kluwer Academic Publishers.
- Byrne, W., Finke, M., Khudanpur, S., McDonough, J., Nock, H. J., Riley, M., Saraçlar, M., Wooters, C., and Zavaliagos, G. (1998). Pronunciation modelling using a hand-labelled corpus for conversational speech recognition. In *Proc. of ICASSP'98*, volume 1, pages 313–316.
- Cernocky, J., Baudoin, G., and Chollet, G. (1997). Speech spectrum representation and coding using multigrams with distance. In *Proc. of ICASSP'97*, volume 2, pages 1343–1346.
- Cernocky, J., Baudoin, G., and Chollet, G. (1998). Segmental vocoder – going beyond the phonetic approach. In *Proc. of ICASSP'98*, volume 2, pages 605–608.
- Chen, S. S., Eide, E., Gales, M. J. F., Gopinath, R., Kanevsky, D., and Olsen, P. A. (1999). Recent improvements to IBM's speech recognition system for automatic transcription of Broadcast News. In *Proc. of ICASSP'99*, volume 1, pages 37–40.
- Chen, S. S. and Gopinath, R. A. (1999). Model selection in acoustic modelling. In *Proc. of EURO-SPEECH'99*, volume 3, pages 1087–1090.
- Chou, W., Lee, C.-H., and Juang, B.-H. (1993). Minimum error-rate training based on N-best string models. In *Proc. of ICASSP'93*, volume 2, pages 652–655.
- Cover, T. M. and Thomas, J. A. (1991). *Elements of information theory*. John Wiley & Sons Inc.
- Cremelie, N. and Martens, J.-P. (1995). On the use of pronunciation rules for improved word recognition. In *Proc. of EURO-SPEECH'95*, pages 1747–1750.
- Cremelie, N. and Martens, J.-P. (1999). In search of better pronunciation models for speech recognition. *Speech Communication*, 29(2–4):115–136.
- Davis, S. B. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. ASSP*, 28(4):357–366.

- de Mori, R., editor (1998). *Spoken dialogue with computers*. Academic Press.
- Deligne, S. and Bimbot, F. (1995). Language modelling by variable length sequences: Theoretical formulation and evaluation. In *Proc. of ICASSP'95*, volume 1, pages 169–172.
- Deligne, S. and Bimbot, F. (1997a). Inference of variable-length acoustic units for continuous speech recognition. In *Proc. of ICASSP'97*, volume 3, pages 1731–1734.
- Deligne, S. and Bimbot, F. (1997b). Inference of variable-length linguistic and acoustic units by multigrams. *Speech Communication*, 23:223–241.
- Deligne, S., Yvon, F., and Bimbot, F. (1995). Variable length sequence matching for phonetic transcription using joint multigrams. In *Proc. of EUROSPEECH'95*, volume 3, pages 2243–2246.
- Deller, J. R., Proakis, J. G., and Hansen, J. H. L. (1993). *Discrete-time processing of speech signals*. Macmillan Publishing Company.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39 Ser. B:1–38.
- Deng, L. (1997a). A dynamic feature-based approach to speech modelling and recognition. In *Proc. 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 107–114, Santa Barbara, California.
- Deng, L. (1997b). Speech recognition using the autosegmental representation of phonological units with the interface to the the trended HMM. *Speech Communication*, 23:211–222.
- Digilakis, V. and Murvetts, H. (1994). Genones: Optimising the degree of mixture tying in a large vocabulary hidden Markov model based speech recognizer. In *Proc. of ICASSP'94*, volume 1, pages 537–540.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern classification*. John Wiley & Sons Inc.
- Eide, E. and Gish, H. (1996). A parametric approach to vocal tract length normalisation. In *Proc. of ICASSP'96*, volume 1, pages 346–348.
- Eide, E., Rohlicek, J. R., Gish, H., and Miller, S. (1993). A linguistic feature representation of the speech waveform. In *Proc. of ICASSP'93*, volume 2, pages 483–486.
- Federico, M. and de Mori, R. (1998). Language modelling. In de Mori, R., editor, *Spoken Dialogues with Computers*, pages 199–230. Academic Press.
- Finke, M. and Waibel, A. (1997a). Flexible transcription alignment. In *Proc. 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 34–48, Santa Barbara, California.
- Finke, M. and Waibel, A. (1997b). Speaking mode dependent pronunciation modelling in large vocabulary continuous speech recognition. In *Proc. of EUROSPEECH'97*, volume 5, pages 2379–2382, Rhodes.

- Fisher, W., Doddington, G., and Goudie-Marshall, K. (1986). The DARPA speech recognition research database: Specification and status. In *Proc. 1986 DARPA Speech Recognition Workshop*, pages 93–99.
- Fosler, E., Weintraub, M., Wegmann, S., Kao, Y.-H., Khudanpur, S., Galles, C., and Saraçlar, M. (1996). Automatic learning of word pronunciation from data. In *Proc. of ICSLP'96*.
- Gales, M. J. F. and Olsen, P. A. (1999). Tail distribution modelling using the Richter and power exponential distributions. In *Proc. of EUROSPEECH'99*, volume 4, pages 1507–1510.
- Gales, M. J. F. and Woodland, P. C. (1996). Mean and variance adaptation within the MLLR framework. *Computer Speech and Language*, 10:249–264.
- Gauvain, J.-L., Lamel, L. F., Adda, G., and Adda-Decker, M. (1994). The LIMSI Nov93 WSJ system. In *Proc. 1994 ARPA Spoken Language Technology Workshop*, pages 125–128, Plainsboro, NJ.
- Gillick, L. and Cox, S. (1989). Statistical significance tests for speech recognition algorithms. In *Proc. of ICASSP'89*, volume 1, pages 532–535.
- Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. In *Proc. of ICASSP'92*, volume 1, pages 517–520.
- Gold, B. and Morgan, N. (1999). *Speech and audio signal processing*. John Wiley & Sons Inc., New York.
- Goldberg, R. G. and Riek, L. (2000). *A practical handbook of speech coders*. CRC Press LLC.
- Good, I. J. (1953). The population frequencies of species and the estimation of population parameters. *Biometrika*, 40:237–264.
- Greenberg, S. (1996). The Switchboard transcription project. 1996 LVCSR summer workshop technical reports, Center for Language and Speech Processing, Johns Hopkins University. <http://www.icsi.berkeley.edu/real/stp>.
- Greenberg, S. (1998). Speaking in shorthand - a syllable-centric perspective for understanding pronunciation variation. In *Proc. ESCA Workshop on modelling pronunciation variation for automatic speech recognition*, pages 47–56, Kerkrade, Netherlands.
- Haeb-Umbach, R. and Ney, H. (1992). Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *Proc. of ICASSP'92*, volume 1, pages 13–16.
- Hain, T. and Woodland, P. C. (1998). CU-HTK acoustic modeling experiments. Presented at the 9<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.
- Hain, T. and Woodland, P. C. (1999a). Dynamic HMM selection for continuous speech recognition. In *Proc. of EUROSPEECH'99*, volume 3, pages 1327–1330.

- Hain, T. and Woodland, P. C. (1999b). Hidden model sequences. Presented at the 10<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.
- Hain, T. and Woodland, P. C. (1999c). Recent experiments with the CU-HTK Hub5 system. Presented at the 10<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.
- Hain, T. and Woodland, P. C. (2000). Modelling sub-phone insertions and deletions in continuous speech recognition. In *Proc. of ICSLP 2000*. Paper No. 1192.
- Hain, T., Woodland, P. C., Evermann, G., and Povey, D. (2000). The CU-HTK March 2000 Hub5 transcription system. In *Proc. 2000 NIST Speech Transcription Workshop*, College Park, Maryland.
- Hain, T., Woodland, P. C., Niesler, T. R., and Whittaker, E. W. D. (1999). The 1998 HTK system for transcription of conversational telephone speech. In *Proc. of ICASSP'99*, pages 57–60.
- Hawkins, P. (1988). *Introducing phonology*. Hutchinson Publishing Group (Australia) Pty Ltd.
- Hemphill, C. T., Godfrey, J. J., and Doddington, G. R. (1990). The ATIS spoken language systems pilot corpus. In *Proc. of the DARPA Speech and Natural Language Workshop*.
- Hermansky, H. (1990). Perceptual linear prediction (PLP) analysis of speech. *J. Acoust. Soc. Am.*, 87(4):1738–1752.
- Holmes, W. (2000). Improving the representation of time structure in front-ends for automatic speech recognition. In *Proc. of ICSLP 2000*, Beijing. Paper No. 1059.
- Huang, X., Acero, A., and Hon, H.-W. (2001). *Spoken language processing: A guide to theory, algorithm, and system development*. Prentice Hall.
- Huang, X. D. and Jack, M. A. (1989). Semi-continuous hidden Markov models for speech signals. *Computer Speech and Language*, 3:239–251.
- Huang, X. D., Lee, K.-F., Won, H. W., and Young, M. Y. (1991). Improved acoustic modelling with the SPHINX speech recognition system. In *Proc. of ICASSP'91*, volume 3, pages 345–348.
- Humphries, J. J. (1997). *Accent modelling and adaptation in automatic speech recognition*. PhD thesis, Cambridge University.
- Hwang, M.-Y. and Huang, X. (1992). Subphonetic modelling with Markov states - senone. In *Proc. of ICASSP'92*, volume 1, pages 33–36.
- Hwang, M.-Y., Huang, X., and Alleva, F. (1993). Predicting unseen triphones with senones. In *Proc. of ICASSP'93*, volume 2, pages 311–314.
- Hwang, M.-Y., Rosenfeld, R., Thayer, E., Mosur, R., Chase, L., Weide, R., Huang, X., and Alleva, F. (1994). Improving speech recognition performance via phone-dependent VQ codebooks and adaptive language models in SPHINX-II. In *Proc. of ICASSP'94*, volume 1, pages 549–552.

- Jelinek, F. (1976). Continuous speech recognition by statistical methods. *Proc. IEEE*, 64(4):532–557.
- Jelinek, F. (1997). *Statistical methods for speech recognition*. MIT Press.
- Jelinek, F. (2000). Central issues in the recognition of conversational speech. In *Proc. 2000 NIST Speech Transcription Workshop*, College Park, Maryland.
- Jelinek, F., Bahl, L. R., and Mercer, R. L. (1975). Design of a linguistic statistical decoder for the recognition of continuous speech. *IEEE Trans. Information Theory*, 21(3):250–256.
- Jelinek, F. and Mercer, R. L. (1980). Interpolated estimation of Markov source parameters from sparse data. In *Proc. of the Workshop on Pattern Recognition in Practice*, North Holland, Amsterdam.
- Jurafsky, D. and Martin, J. H. (2000). *Speech and Language Processing*. Prentice Hall.
- Katz, S. M. (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Trans. ASSP*, 35(3):400–401.
- Lee, C.-H., Giachin, E., Rabiner, L. R., Pierachini, R., and Rosenberg, A. E. (1991). Improved acoustic modelling for speaker independent large vocabulary continuous speech recognition. In *Proc. of ICASSP'91*, volume 1, pages 161–164.
- Lee, K.-F. (1989). *Automatic speech recognition: The development of the SPHINX system*. Kluwer Academic Publishers.
- Lee, K.-F. (1990). Context-dependent phonetic hidden Markov models for speaker-independent continuous speech recognition. *IEEE Trans. ASSP*, 38(4):599–609.
- Leggetter, C. J. and Woodland, P. C. (1995). Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. *Computer Speech and Language*, 9(2):171–186.
- Levinson, S. E. (1994). Speech recognition technology: A critique. In Roe, D. B. and Wilpon, J. G., editors, *Voice communication between humans and machines*, pages 159–164. National Academic Press.
- Liporace, L. L. (1982). Maximum likelihood estimation for multivariate observations of Markov sources. *IEEE Trans. Information Theory*, 28(5):729–734.
- Luo, X. (1998). *Balancing model resolution and generalizability in large vocabulary continuous speech recognition*. PhD thesis, Johns Hopkins University.
- Luo, X. and Jelinek, F. (1999). Probabilistic classification of HMM states for large vocabulary continuous speech recognition. In *Proc. of ICASSP'99*, pages 2044–2047.
- Ma, K., Zavaliagos, G., and Iyer, R. (1998). BBN pronunciation modelling. Presented at the 9<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.

- Makhoul, J. and Schwartz, R. (1994). State of the art continuous speech recognition. In Roe, D. B. and Wilpon, J. G., editors, *Voice communication between humans and machines*, pages 165–198. National Academic Press.
- Martin, A. (1995). Statistical significance tests for speech recognition benchmark tests. Technical report, U.S. National Institute for Standards and Technology.
- Martin, A., Przybocki, M., Fiscus, J., and Pallett, D. (2000). The 2000 NIST evaluation for recognition of conversational speech over the telephone. In *Proc. 2000 NIST Speech Transcription Workshop*, College Park, Maryland.
- Merhav, N. and Ephraim, Y. (1991). Hidden Markov modelling using the most likely state sequence. In *Proc. of ICASSP'91*, volume 1, pages 469–472.
- Miller, D. R. H. and McDonough, J. (1997). BBN 1997 acoustic modelling. Presented at Conversational Speech Recognition Workshop DARPA Hub-5E Evaluation.
- Nadas, A. (1985). On Turing's formula for word probabilities. *IEEE Trans. ASSP*, 33(6):1414–1416.
- Ney, H. and Essen, U. (1993). Estimating small probabilities by leaving-one-out. In *3<sup>rd</sup> European Conference on Speech Communication and Technology*, pages 2239–2242, Berlin.
- Ney, H., Essen, U., and Kneser, R. (1995). On the estimation of "small" probabilities by leaving-one-out. *IEEE Trans. PAMI*, 17(12):1202–1212.
- Ney, H., Haeb-Umbach, R., Tran, B.-H., Essen, U., Kneser, R., Oerder, M., Meier, H.-G., Aubert, X., Dugast, C., Geller, D., Höllerbauer, W., and Bartosik, H. (1993). The Philips research system for large-vocabulary continuous speech recognition. In *Proc. of EUROSPEECH'93*, volume 3, pages 2125–2128, Berlin.
- Ney, H., Martin, S., and Wessel, F. (1997). Statistical language modelling using leaving-one-out. In Young, S. J. and Bloothoof, G., editors, *Corpus-Based Methods in Language and Speech Processing*, pages 174–208. Kluwer Academic Publishers.
- Nguyen, L., Makhoul, J., Schwartz, R., Kubala, F., LaPre, C., Yuan, N., Zhao, Y., Anastasakos, T., and Zavagliakos, G. (1995). The 1994 BBN/BYBLOS speech recognition system. In *Proc. of Spoken Language Systems Technology Workshop*, Austin, Texas.
- Niesler, T. R. (1997). *Category-based statistical language models*. PhD thesis, Cambridge University.
- Nock, H. J., Gales, M. J. F., and Young, S. J. (1997). A comparative study of methods for phonetic decision-tree state clustering. In *Proc. of EUROSPEECH'97*, volume 1, pages 111–114.
- Nock, H. J. and Young, S. J. (1998). Detecting and correcting poor pronunciations for multi-word units. In *Proc. ESCA Workshop on modelling pronunciation variation for automatic speech recognition*, pages 85–90, Kerkrade, Netherlands.

- O'Connor, J. D. (1973). *Phonetics*. Penguin Books Ltd., Harmondsworth, Middlesex, England.
- Odell, J. J. (1995). *The use of context in large vocabulary speech recognition*. PhD thesis, Cambridge University.
- O'Shaughnessy, D. (1987). *Speech communication – human and machine*. Addison-Wesley.
- Ostendorf, M. (1999). Moving beyond the "Beads-on-a-String" model of speech. In *Proc. 1999 IEEE Workshop on Automatic Speech Recognition and Understanding*, volume 1, pages 79–83.
- Ostendorf, M., Digalakis, V. V., and Kimball, O. A. (1996). From HMMs to segment models: A unified view of stochastic modelling for speech recognition. *IEEE Trans. SAP*, 4(5):360–378.
- Ostendorf, M. and Singer, H. (1997). HMM topology design using maximum likelihood successive state splitting. *Computer Speech and Language*, 11:17–41.
- Paul, D. B. and Baker, J. M. (1992). The design for the Wall Street Journal-based corpus. In *Proc. of ICSLP'92*, volume 2, pages 899–902.
- Peskin, B., Newmann, M., McAllaster, D., Nagesha, V., Richards, H., Wegmann, S., Hunt, M., and Gillick, L. (1999). Improvements in recognition of conversational telephone speech. In *Proc. of ICASSP'99*, pages 53–56.
- Povey, D. and Woodland, P. C. (1999). Frame discrimination training of HMMs for large vocabulary speech recognition. In *Proc. of ICASSP'99*, pages 333–336.
- Price, P. J., Fisher, W., and Bernstein, J. (1988). A database for continuous speech recognition in a 1000 word domain. In *Proc. of ICASSP'88*, volume 1, pages 651–654.
- Rabiner, L. R. (1993). *Fundamentals of speech recognition*. Prentice Hall Signal Proc. Series.
- Rabiner, L. R. and Schafer, R. W. (1978). *Digital processing of speech signals*. Prentice Hall Signal Proc. Series.
- Richardson, M., Bilmes, J., and Diorio, C. (2000). Hidden-articulator Markov models: Performance improvements and robustness to noise. In *Proc. of ICSLP 2000*. Paper No. 1644.
- Richter, A. G. (1986). Modelling of continuous speech observations. In *Conf. on Advances In Speech Recognition*, IBM Europe Institute, Oberlech, Austria.
- Riley, M., Byrne, W., Finke, M., Khudanpur, S., Lolje, A., McDonough, J., Nock, H. J., Saraçlar, M., Wooters, C., and Zavaliagos, G. (1999). Stochastic pronunciation modelling from hand-labelled phonetic corpora. *Speech Communication*, 29:209–224.
- Robinson, A. J. (1994). An application of recurrent nets to phone probability estimation. *IEEE Trans. Neural Networks*, 5(2):298–305.
- Russell, M. J. (1997). Progress towards speech models that model speech. In *Proc. 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 114–115.

- Saraçlar, M. (2000). *Pronunciation modelling for conversational speech recognition*. PhD thesis, Johns Hopkins University.
- Saraçlar, M. and Khudanpur, S. (2000). Pronunciation ambiguity vs pronunciation variability in speech recognition. In *Proc. of ICASSP'2000*, volume 3, pages 1679–1682, Istanbul.
- Saraçlar, M., Nock, H. J., and Khudanpur, S. (2000). Pronunciation modelling by sharing Gaussian densities across phonetic models. *Computer Speech and Language*, 14:137–160.
- Schwartz, R. (1978). Estimating the dimension of a model. *The Ann. of Statistics*, 6:461–464.
- Seong-Yun and Oh, Y.-H. (1999). Stochastic lexicon modelling for speech recognition. *IEEE Signal Proc. Letters*, 6(2):28–30.
- Shaffer, J. and Picone, J. (1998). *Rule and guidelines for transcription and segmentation of the SWITCHBOARD large vocabulary conversational speech recognition corpus*. ISIP, Mississippi State University.
- Stolcke, A., Bratt, H., Butzberger, J., Franco, H., Gadde, V. R. R., Plauche, M., Richey, C., Shriberg, E., Sonmez, K., Weng, F.-L., and Zhen, J. (2000). The SRI March 2000 Hub-5 conversational speech transcription system. In *Proc. 2000 NIST Speech Transcription Workshop*, College Park, Maryland.
- Strik, H. and Cucchiaroni, C. (1999). Modelling pronunciation variation for ASR: A survey of the literature. *Speech Communication*, 29:225–246.
- Sun, J., Jing, X., and Deng, L. (2000). Data-driven model construction for continuous speech recognition using overlapping articulatory features. In *Proc. of ICSLP 2000*. Paper No. 682.
- Weintraub, M. (1998). Error metrics for speech recognition. Presented at the 9<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.
- Weintraub, M., Fosler, E., Galles, C., Kao, Y.-H., Khudanpur, S., Saraçlar, M., and Wegmann, S. (1996a). WS96 progress report: Automatic learning of word pronunciation from data. Technical report, Center for Language and Speech Processing, Johns Hopkins University.
- Weintraub, M., Taussig, K., Hunicke-Smith, K., and Snodgrass, A. (1996b). Effect of speaking style on LVCSR performance. In *Proc. of ICSLP'96*, pages S16–S19.
- Wilpon, J. G., Lee, C. H., and Rabiner, L. R. (1991). Improvements in connected digit recognition using higher order spectral and energy features. In *Proc. of ICASSP'91*, volume 1, pages 349–352.
- Witten, I. H. and Bell, T. C. (1991). The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Trans. Information Theory*, 37(4):1085–1094.
- Woodland, P. C., Gales, M. J. F., Pye, D., and Young, S. J. (1997a). Broadcast News transcription using HTK. In *Proc. of ICASSP'97*, pages 719–722.

Woodland, P. C., Kapadia, S., Nock, H. J., and Young, S. J. (1997b). CU-HTK March 1997 Hub5E system. Presented at the 8<sup>th</sup> Conversational Speech Recognition Workshop, MITAGS, Linthicum Heights, Maryland.

Woodland, P. C., Odell, J. J., Valtchev, V., and Young, S. J. (1994). Large vocabulary continuous speech recognition using HTK. In *Proc. of ICASSP'94*, volume 2, pages 125–128.

Wooters, C. and Stolcke, A. (1994). Multiple-pronunciation lexical modeling in a speaker-independent speech understanding system. In *Proc. of ICSLP'94*, volume 3, pages 1363–1367.

Young, S. J. (2001). Statistical modelling in continuous speech recognition (CSR). In *Proc. Intl. Conf. on Uncertainty in Artificial Intelligence*, Seattle, WA.

Young, S. J. and Chase, L. L. (1998). Speech recognition evaluation: A review of the U.S. CSR and LVCSR programmes. *Computer Speech and Language*, 12:263–279.

Young, S. J., Kershaw, D., Odell, J. J., Ollason, D., Valtchev, V., and Woodland, P. C. (1999). *The HTK Book (for HTK version 2.2)*. Entropic Ltd., Cambridge, England. <http://htk.eng.cam.ac.uk>.

Young, S. J., Odell, J. J., and Woodland, P. C. (1994). Tree-based state tying for high accuracy acoustic modelling. In *Proc. 1994 ARPA Human Language Technology Workshop*, pages 307–312. Morgan Kaufmann.

Young, S. J., Russell, N. H., and Thornton, J. H. S. (1989). Token passing: A simple conceptual model for connected speech recognition systems. Technical Report CUED/F-INFENG/TR38, Cambridge University Engineering Department.

Young, S. J. and Woodland, P. C. (1993). The use of state-tying in continuous speech recognition. In *Proc. of EUROSPEECH'93*, volume 3, pages 2203–2206.

Young, S. J. and Woodland, P. C. (1994). State clustering in hidden Markov model-based continuous speech recognition. *Computer Speech and Language*, 8:369–383.