Investigations on Discriminative Training Criteria

**Von der Fakultät für Mathematik, Informatik
und Naturwissenschaften
der Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften genehmigte Dissertation**

vorgelegt von

Diplom–Physiker Ralf Schlüter

aus

Rheda, jetzt Rheda-Wiedenbrück

**Berichter: Universitätsprofessor Dr.–Ing. Hermann Ney
Professor Dr. Renato De Mori**

Tag der mündlichen Prüfung: 22. September 2000

Für Hedwig und Leo

Acknowledgements

At this point, I would like to express my thanks to all the people who supported and accompanied me during the progress of this work.

In particular, I would like to thank:

Prof. Dr.-Ing. Hermann Ney for introducing me into the very interesting area of speech recognition research, for the possibility to realize this work at the *Lehrstuhl für Informatik VI* lead by him, for his continuous interest and numerous fruitful and clarifying discussions;

Prof. Renato De Mori, who leads the *Laboratoire d'Informatique* at the *Université d'Avignon et des Pays Vaucluse*, for kindly taking over the task of the co-referee for this work;

Dr. Harald Höge from Siemens AG Munich for supporting this work;

Klaus Beulen, Sven Martin, Stefan Ortmanns, and Lutz Welling for helping me to get this work started;

Wolfgang Macherey and Boris Müller for performing much of the implementations and experiments presented in this work;

Frank Wessel for accompaniment on enlighting paths through word graphs;

Achim Sixtus, Stephan Kanthak, Klaus Beulen, Oliver Bender, and Michael Motter for excellent support with the computing equipment;

All the people at the *Lehrstuhl für Informatik VI* for many constructive discussions and the good atmosphere;

Annette, Claudia, Andreas, Silke, Adel, Linda-Luna, Djassim, Merjan, Silvia, Falk, Sabine, Ludger, Michael, and all the other people for the time we spent together;

And finally my parents Hedwig and Leo for enabling and supporting me with my work and my interests.

Abstract

In this work, a framework for efficient discriminative training and modeling is developed and implemented for both small and large vocabulary continuous speech recognition. Special attention will be directed to the comparison and formalization of varying discriminative training criteria and corresponding optimization methods, discriminative acoustic model evaluation and feature extraction.
A formally unifying approach for a class of discriminative training criteria including *Maximum Mutual Information* (MMI) and *Minimum Classification Error* (MCE) criterion is presented, including the optimization methods gradient descent (GD) and *extended Baum-Welch* (EB) algorithm. Using discriminative criteria, novel approaches to splitting of mixture Gaussian densities and to linear feature transformation are derived. Furthermore, efficient algorithms for the application of discriminative training to speech recognition with both small and large vocabulary are developed. Finally, a novel evaluation method for the stochastic models used in speech recognition is derived using methods related to discriminative training.
Experiments have been carried out on the *TI digit string* corpus for American English continuous digit strings, the *SieTill* corpus for telephone line recorded German continuous digit strings, the *Verbmobil* corpus for German spontaneous speech and the *Wall Street Journal* corpus for American English read speech.

Zusammenfassung

In dieser Arbeit wird ein Rahmen für effizientes diskriminatives Training entwickelt und für kontinuierliche Spracherkennung mit kleinen und großen Vokabularien implementiert. Besondere Aufmerksamkeit wird dabei auf den Vergleich und Formalisierung diverser diskriminativer Trainingskriterien und entsprechender Optimierungsmethoden, diskriminative Bewertung akustischer Modelle und die Merkmalsextraktion gelegt.
Für eine Klasse diskriminativer Trainingskriterien wird ein formal einheitlicher Rahmen eingeführt, der unter anderem das *Maximum Mutual Information* (MMI) und das *Minimum Classification Error* (MCE) Kriterium enthält, und auch die entsprechenden Optimierungsmethoden Gradientenabstieg (GD) und den *erweiterten Baum-Welch* (EB) Algorithmus umfasst. Es werden neue diskriminative Ansätze zum Aufsplitten von Gaußschen Mischverteilungen und zum Training linearer Merkmalstransformationen vorgestellt. Des weiteren werden effiziente Algorithmen für die Anwendung von diskriminativem Training auf die Spracherkennung bei kleinem wie großem Vokabular entwickelt. Schließlich wird ein neuer Ansatz zur Bewertung der in der Spracherkennung verwendeten stochastischen Modelle vorgestellt, der auf Methoden aufbaut, die für das diskriminative Training entwickelt wurden.
Experimente wurden auf dem *TI digit string* Korpus (Ziffernketten in amerikanischem Englisch), dem *SieTill* Korpus (deutsche Ziffernketten, Telefonqualität) durchgeführt, dem *Wall Street Journal* Korpus (gelesenes amerikanisches Englisch), sowie auf dem *Verbmobil* Korpus für deutsche Spontansprache durchgeführt.

Contents

1 Introduction **1**
- 1.1 Statistical Speech Recognition . . . 2
  - 1.1.1 Acoustic Feature Extraction . . . 3
  - 1.1.2 Acoustic Modeling . . . 4
  - 1.1.3 Language Modeling . . . 7
  - 1.1.4 Search . . . 9
- 1.2 Discriminative Training: State of the Art . . . 11
  - 1.2.1 Discriminative Criteria . . . 11
    - 1.2.1.1 Maximum Mutual Information Criterion . . . 12
    - 1.2.1.2 Minimum Classification Error Criterion . . . 13
    - 1.2.1.3 Comparison of Criteria . . . 14
  - 1.2.2 Parameter Optimization for Discriminative Criteria . . . 14
    - 1.2.2.1 Gradient Descent . . . 14
    - 1.2.2.2 Extended *Baum-Welch* Algorithm . . . 14
  - 1.2.3 Efficiency of Discriminative Training . . . 15
    - 1.2.3.1 Time Alignment . . . 15
    - 1.2.3.2 Alternative Word Sequences . . . 16
  - 1.2.4 Discriminative Modeling . . . 17
    - 1.2.4.1 Language Models and Training . . . 17
    - 1.2.4.2 Model Evaluation . . . 17
    - 1.2.4.3 Feature Transformation . . . 18

2 Scientific Goals **21**

3 Discriminative Training Criteria **25**
- 3.1 Unifying View . . . 25
  - 3.1.1 Choice of particular criteria . . . 26
  - 3.1.2 Smoothing . . . 26
- 3.2 Comparison and Other Criteria . . . 27
  - 3.2.1 Minimum Error Based Criteria . . . 27
  - 3.2.2 Criteria related to Maximum Mutual Information . . . 27
  - 3.2.3 Alternative Word Sequences . . . 28
  - 3.2.4 Limiting Cases . . . 28
  - 3.2.5 Interpretation and Relation to Frame Based Methods . . . 28
  - 3.2.6 Relation to Other Discriminative Criteria . . . 29
- 3.3 Asymptotic Behaviour of Discriminative Criteria . . . 29

3.3.1 Error Bounds . . . 29
3.3.1.1 Minimum Error Probability and MMI criterion . . . 30
3.3.1.2 Minimum Error Probability and MCE criterion . . . 31
3.3.2 Model-Free Optimization . . . 33
3.3.2.1 ML and MMI Criterion . . . 33
3.3.2.2 MCE and Related Criteria . . . 33
3.3.3 Discussion of Asymptotic Behaviour . . . 34
3.4 Comparative Results for Discriminative Criteria . . . 34
3.4.1 Criterion Performance Depending on Model Complexities . . . 35
3.4.2 Approximated vs. Complete Criteria . . . 36

4 The Parameter Optimization Problem 37
4.1 Discriminative Averages . . . 37
4.1.1 Continuous Mixture Densities . . . 37
4.1.2 Conventional Forward-Backward Probability . . . 38
4.1.3 Generalized Forward-Backward Probability . . . 38
4.1.4 Formal Differentiation of the Unified Criterion . . . 38
4.1.5 Definition of Discriminative Averages . . . 39
4.1.5.1 Further Decomposition of Discriminative Averages . . . 39
4.1.6 Discriminative Averages in Viterbi Approximation . . . 40
4.1.6.1 Decomposed Discriminative Averages in Viterbi Approximation . . . 40
4.2 Extended *Baum* Algorithm . . . 41
4.2.1 Optimization of the MMI Criterion . . . 41
4.2.2 Extension to the Unified Criterion . . . 41
4.2.3 Derivation of Reestimation Formulae . . . 42
4.2.4 Reestimation Formulae for Gaussian Mixtures . . . 42
4.2.4.1 Smoothed Reestimation of Mixture Weights . . . 43
4.3 Gradient Descent . . . 44
4.3.1 Derivation of Reestimation Formulae . . . 44
4.4 Comparison of Optimization Methods . . . 45
4.4.1 Interdependence between EB and GD . . . 45
4.4.2 Comparative Results for Parameter Optimization . . . 46
4.5 Convergence Control . . . 48
4.5.1 Derivation of Iteration Constants . . . 48
4.5.1.1 Iteration Constants for the Means and Variances . . . 48
4.5.1.2 Iteration Constants for the Mixture Weights . . . 50
4.5.2 Global Convergence Heuristics . . . 51
4.5.3 Convergence Results for Small Vocabulary . . . 51
4.5.3.1 Overall Convergence . . . 51
4.5.3.2 Non-Monotonic Behaviour . . . 51
4.5.3.3 Choice of Parameter Sets . . . 53
4.5.4 Convergence for Large Vocabulary . . . 54
4.5.4.1 Overall Convergence . . . 55
4.5.4.2 Choice of Parameter Sets . . . 55

5 Discriminative Training for Speech Recognition 57
5.1 Discriminative Training Algorithm . . . 57
5.1.1 Training Procedure and Complexity for Small Vocabulary . . . 58
5.1.2 Training Procedure and Complexity for Large Vocabulary . . . 58
5.2 Alternative Word Sequences . . . 59
5.2.1 Representation of Alternative Word Sequences . . . 59
5.3 Recognition for Discriminative Training . . . 59
5.3.1 MMI Training using Word Graphs with Fixed Word Boundaries . . 59
5.3.2 Constrained Recognition for Discriminative Training . . . 60
5.3.3 Evaluation of Constrained Recognition . . . 62
5.4 Choice of Language Models . . . 62
5.4.1 Possible Effects of Language Models on Discriminative Training . . 62
5.4.2 Experiments with Varying Language Models for MMI Training . . . 63
5.4.3 Interdependence between Language Models for Recognition and MMI Training . . . 64
5.5 Spontaneous Speech . . . 64

6 Efficient Estimation of Discriminative Statistics 67
6.1 Word Sequence Based HMM-State Posterior Probabilities . . . 68
6.1.1 ML Training: *Viterbi* Approx. vs. State Summation . . . 69
6.2 Word Posterior Probabilities . . . 70
6.2.1 Definition of Word Probabilities . . . 70
6.2.2 Decomposition into Forward and Backward Word Probabilities . . . 70
6.2.3 Forward and Backward Word Recursions . . . 71
6.2.4 Disadvantages of N-Best Lists . . . 71
6.3 Word Graph based HMM-State Posterior Probabilities . . . 72
6.3.1 Definition of State Probabilities . . . 72
6.3.2 Handling of the Smoothing Exponent . . . 73
6.3.3 Decomposition into Forward and Backward State Probabilities . . . 73
6.3.4 Forward and Backward State Recursions . . . 75
6.3.5 MMI Training: *Viterbi* Approx. vs. State Summation . . . 76
6.4 MCE Training using Word Graphs . . . 77
6.4.1 Competing-Word Probabilities for MCE Training . . . 77

7 Refined Scoring Approaches 79
7.1 Decompositions of the Posterior and Joint Probabilities . . . 79
7.1.1 Standard Model for the Joint Probability . . . 79
7.1.2 Summation over Word Boundaries . . . 80
7.1.3 Alternative Decomposition for the Posterior Probability . . . 83
7.2 Search Using Forward and Backward Word Probabilities . . . 83
7.2.1 Normalization Properties of Word Posterior Probabilities . . . 84
7.2.2 Experimental Results for Search using Word Posterior Probabilities 86
7.2.2.1 Word Posterior Probabilities Without Context . . . 87
7.2.2.2 Word Posterior Probabilities With Single Word Context . 88
7.3 Efficient Search involving HMM-State Summation . . . 88

8 Discriminative Criterion Based Acoustic Modeling **91**
8.1 Mixture Density Splitting . . . 91
8.1.1 Interdependence of Discriminative Training and Model Complexity 91
8.1.2 Discriminative Splitting Choice . . . 92
8.1.3 Alternative Derivation of Discriminative Splitting . . . 92
8.1.4 Determination of Parameters for Splitted Densities . . . 93
8.1.5 Hybrid MMI/ML Mixture Density Splitting Algorithm . . . 93
8.1.6 Experiments for Hybrid MMI/ML Splitting . . . 93
8.1.6.1 Convergence . . . 95
8.1.6.2 Distribution of Densities . . . 95
8.2 Linear Feature Transformation . . . 96
8.2.1 Dimension Reducing Linear Feature Transformations . . . 96
8.2.2 Normalization . . . 97
8.2.3 Discriminative Linear Feature Transformation . . . 97
8.2.4 Limitations and Smoothing . . . 98
8.2.5 Linear MMI Analysis (LMA) . . . 99
8.2.6 Experiments using LMA . . . 99
8.2.6.1 Determination of the Smoothing Parameter . . . 99
8.2.6.2 Comparison of LMA and LDA . . . 100

9 Scientific Contributions **103**

10 Outlook **107**

A Speech Corpora and Recognition Systems **109**
A.1 Continuous Digit Strings . . . 109
A.1.1 Digit Corpora . . . 109
A.1.2 Continuous Digit Recognition Systems . . . 109
A.2 Spontaneous Speech . . . 110
A.2.1 *Verbmobil I* . . . 111
A.2.2 *Verbmobil II* . . . 111
A.2.3 *Arise* . . . 112
A.2.4 *Broadcast News* Transcription . . . 112
A.3 Read Speech . . . 113
A.3.1 *Wall Street Journal* (WSJ) 5k . . . 113
A.3.2 *North American Business* (NAB) corpus . . . 114

B Symbols and Acronyms **115**
B.1 Mathematical Symbols . . . 115
B.2 Acronyms . . . 119

C Detailed Calculations **121**
C.1 Derivation of Minimal Iteration Constant Ensuring Positive Variance . . . 121
C.2 Derivative of the Acoustic Emission HMM . . . 123

Bibliography **125**

List of Tables

3.1 Several discriminative criteria, which are contained in the unified formulation defined in Eq. (3.1). 26
3.2 Comparison of recognition results for several discriminative training criteria on the *SieTill* corpus for telephone line recorded, continuously spoken German digits. Results are given for low (single densities) and optimal model complexity. The number of densities per mixture is denoted by 'dns'. 35

4.1 Recognition results for the *TI digit string* corpus. Word (WER) and sentence error rates (SER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with state specific diagonal covariances. 47
4.2 Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with state specific diagonal covariances. 47
4.4 Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Gaussian mixture densities with 4 densities per state, one pooled diagonal covariance and LDA. 47
4.3 Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with one pooled diagonal covariance. 48

5.1 Comparison of rescoring and constrained recognition using word graphs for the iterative determination of alternative word sequences during discriminative training. Results on the *Wall Street Journal* corpus (5k), training and recognition with bigram language model. 60
5.2 Performance of constrained recognition compared to full recognition. 62
5.3 Language model perplexities: ARPA WSJ0 training and testing corpora. The notations "bi-phr" and "tri-phr" refer to language models containing phrases/multiwords. 63
5.4 Comparison of several language models for MMI training and recognition. Results on the *Wall Street Journal* corpus (5k). 64

6.1 ML training using the *Viterbi* approximation or the *Baum-Welch* algorithm. Results for *Verbmobil I*, evaluation corpus 1996 (cf. Appendix A.2.1). 69

6.2 MMI and ML training using the *Viterbi* approximation and exact state summation. Results for the *SieTill* corpus. 77

7.1 Experimental details for the five different testing corpora. WGD denotes the word graph density, NGD the node graph density, BGD the boundary graph density, GER the word graph error rate in [%] and WER the word error rate in [%]. For details on the word graph measures see [Ortmanns$^+$ 1997a]. 87

7.2 Experimental results for five different testing corpora. Standard recognition is compared to recognition with word posterior probabilities without using context and using single word context. In all cases a trigram language model has been used. For further experimental details cf. Table 7.1. WER denotes the word error rate in [%]. 88

8.1 Comparison of the discriminative and conventional mixture density splitting. Results on the *SieTill* corpus. In the column 'dns' the average number of densities per mixture is given. 94

8.2 Comparison of feature transformation using *linear discriminant analysis* (LDA) and linear MMI analysis (LMA). Recognition results on the *SieTill* corpus. 101

A.1 Corpus statistics: Digit corpora. 109

List of Figures

1.1 Architecture of an automatic speech recognition system [Ney 1990] 2
1.2 HMM in *Bakis* topology. 5

4.1 CT criterion as a function of the iteration index for single Gaussian densities (*TI digit string* training corpus). 52
4.2 Word error rate as a function of the iteration index for CT using single Gaussian densities (male portion of the *TI digit string* corpus). 52
4.3 CT criterion as a function of the iteration index for single and mixture Gaussian acoustic models (*SieTill* training corpus). 53
4.4 Word error rates on the training corpus as a function of the iteration index for Corrective Training (CT) using single and mixture Gaussian acoustic models (*SieTill* training corpus). 53
4.5 MCE, MMI and CT criterion as a function of the iteration index for mixture Gaussian acoustic models with 32 densities per state (*SieTill* training corpus). Note that the MCE and MMI training iterations begin after 10 iterations of Corrective Training (CT). The values of the MCE criterion were scaled by a factor of 20. 54
4.6 Word error rates on the training corpus as a function of the iteration index for the MCE, MMI and CT criterion using mixture Gaussian acoustic models with 32 densities per state (*SieTill* training corpus). Note that the MCE and MMI training iterations begin after 10 iterations of Corrective Training (CT). 54
4.7 MMI criterion as a function of the iteration index for the WSJ0 training corpus. 55
4.8 Word error rates as a function of the iteration index of an MMI training for the WSJ0 training corpus and the WSJ0 Nov. '92 development test set. 55

5.1 Constrained recognition: words of the word graph, which are allowed to be started at time τ. 61
5.2 Lexical prefix tree for constrained recognition. Allowed words are marked with black squares. 61

6.1 Time-state space for calculation of *forward-backward* (FB) probability $\gamma_{rt}(s|W_r)$. Highlighted by thick arrows is a possible *Viterbi* alignment path. 69

6.2 Time-state space for calculation of generalized *forward-backward* (FB) probability $\gamma_{rt}(s|W_r)$. Highlighted by thick arrows are possible *Viterbi* alignment paths for the individual words, as they would contribute to the word posterior probabilities. 75

7.1 A simplified word graph for illustrational purposes. The solid edges in the graph above represent word and silence hypotheses. We assume that no language model is used and that all acoustic probabilities are equal. For each edge $[w;\tau,t]$ the posterior hypothesis probability $p([w;\tau,t]|x_1^T)$ is specified. As can be seen, these probabilities sum up to unity for any point in time. 85

8.1 Evolution of word error rates on the *SieTill* test corpus for the proposed hybrid MMI/ML splitting approach and for conventional splitting with ML and MMI training. 94
8.2 Comparison of the average log-likelihood from ML training against number of Gaussian densities for both splitting approaches considered here (female portion of the *SieTill* corpus). 95
8.3 Distribution of the number of densities per mixture obtained by applying the proposed hybrid MMI/ML splitting approach to the male portion of the *SieTill* corpus. 96
8.4 Preliminary optimization of the smoothing parameter η for the LMA. Recognition results on the *SieTill* training corpus. 100

Chapter 1

Introduction

By now, speech processing in general, and automatic speech recognition in particular, have diversified into a wide range of closely interrelated application areas and research directions. Speech is one of the most natural means of human communication and automatic speech recognition therefore represents a convenient basis for the development of human-machine interfaces, telecommunication services, and multimedia tools. Human speech production together with the excellent human ability to recognize speech even under demanding conditions gives rise to a wide range of scientific and technical challenges and leads to important research topics, such as statistics, pattern recognition, signal processing, phonetics, psycho-acoustics, neuro-physiology, and linguistics, to name but a few.

Throughout this work, automatic speech recognition is investigated within the statistical framework of *Bayes*' decision rule [Duda & Hart 1973]. In this approach, the acoustic signal of a given speech utterance first is preprocessed into a stream of acoustic feature vectors. In a global search procedure, stochastic models of speech are then used to determine that word sequence with the highest posterior probability, given the corresponding acoustic features. On the basis of spoken and text corpora, which are representative for the envisioned speech recognition task and using prior knowledge such as pronunciation lexical, stochastic models of the acoustic and linguistic properties of speech, i.e. the *acoustic model* and the *language model*, are built in preliminary training phases.

Automatic speech recognition systems usually are aimed at obtaining optimal word or sentence error rate. Standard training criteria for speech recognition are designed to determine the parameters of the acoustic models such that the training data is optimally described. These criteria are optimal in the sense of recognition performance, provided the model assumptions are correct and sufficient training data is available. Since these conditions usually are *not* met, in this work, discriminative training criteria are investigated, which are more closely related to the aim of optimal recognition performance. Discriminative training criteria directly aim at the optimization of the separability between the spoken (correct) word sequences, and all the alternative (competing) word sequences.

1.1 Statistical Speech Recognition

In the field of automatic speech recognition the statistical approach by now has been established as *de-facto* standard. As illustrated in Fig. 1.1,

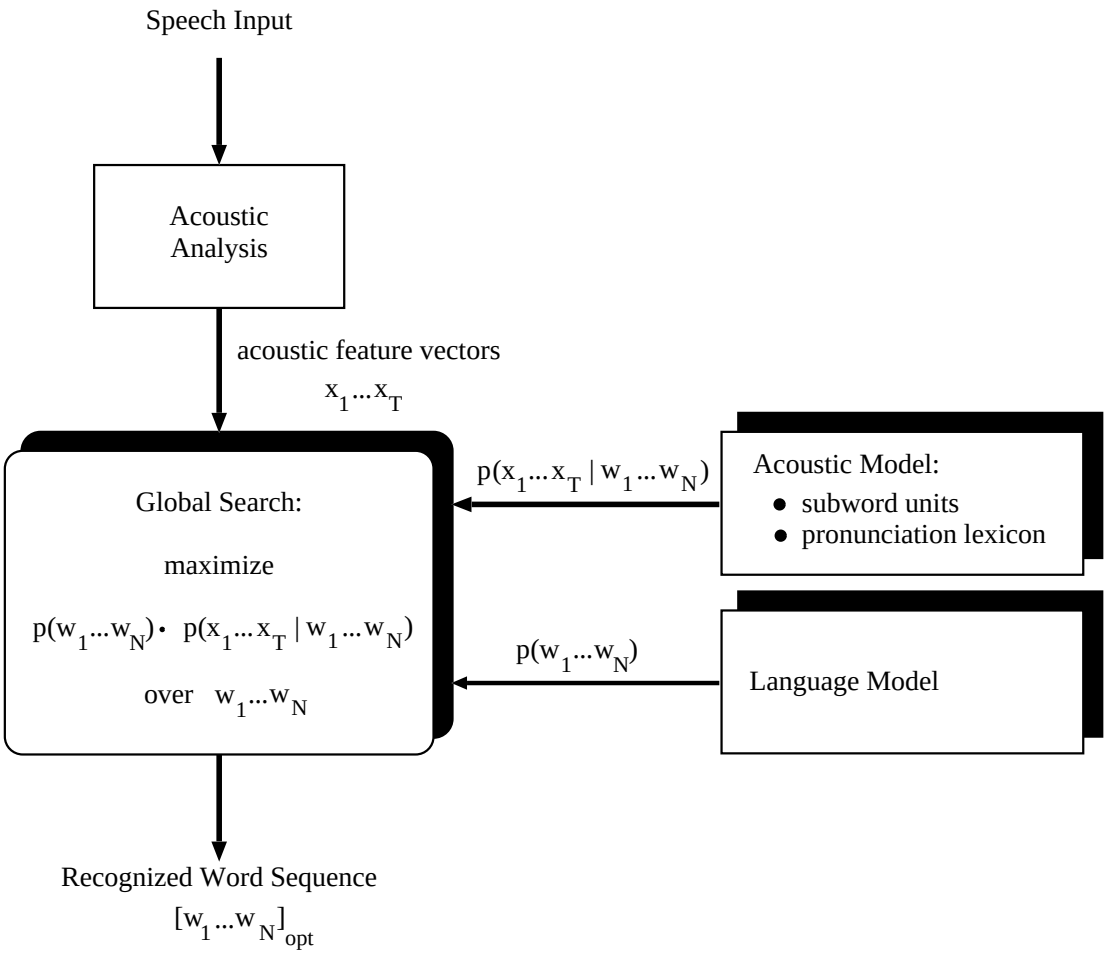

Figure 1.1: Architecture of an automatic speech recognition system [Ney 1990]

the architecture of an automatic speech recognition system in the framework of *Bayes'* decision rule consists of the following four main components:

1. The extraction of the *acoustic features*, i.e. the parameterization of the analog speech signal.

2. The *acoustic model*. For a given word sequence this model gives the probability to observe a corresponding temporal sequence of acoustic features. The acoustic model could be divided into two knowledge sources:

   (a) acoustic models for the smallest sub-word units, which are to be distinguished; i.e. phoneme, syllable or even whole word models, and

   (b) the pronunciation lexicon, which defines the decomposition of the words into the subword units.

3. The *language model*, a statistical model of the syntax, semantics, and pragmatics of the speech to be recognized.

4. The *search* procedure of the word sequence with the highest *a-posteriori* probability for a given speech signal.

These components will be described and standard procedures for the determination of the corresponding stochastic models will be outlined in the following.

1.1.1 Acoustic Feature Extraction

The aim of the acoustic feature extraction is to provide the automatic speech recognition system with a parameterization of the analog speech signal. From the point of view of optimal performance of the speech recognition system, this parameterization should fulfill the following requirements:

- The acoustic features are to enable optimal distinction of the basic speech units to be modeled by the acoustic models.

- The dimension of the acoustic features should at the same time be small enough to enable reliable estimation of the corresponding statistical models.

- The acoustic features are to contain only the speech information. Dependencies on speaker, acoustic channel, or the acoustic environment including other acoustic sources should not be contained as far as possible.

For the optimization of the feature extraction, todays state of the art speech recognizers do not explicitly take into account the structure of the subsequent recognizer modules. Usually, the feature extraction is optimized empirically with respect to the recognition performance. There have been attempts to use discriminative training criteria for data-driven optimization of the parameters of the feature extraction [Biem & Katagiri 1997], but so far these approaches did not lead to significant improvements in error rate.

Feature extraction methods of high performance speech recognizers today mostly are based on either Fourier transformation or linear prediction, usually combined with cepstral analysis. Important representatives of these signal analysis methods are the *mel frequency cepstral coefficients* (MFCC) [Davis & Mermelstein 1980], and the *perceptual linear prediction* (PLP) [Hermansky 1990].
It could be shown that several methods of linear transformation could further improve the quality of the acoustic features. A standard method is to include dynamic features, i.e. the first and second order discrete derivatives or regression coefficients of the basic acoustic features [Picone 1993]. More generally, *linear discriminant analysis* (LDA) provides the most important features based on a class separation measure [Hunt & Lefèbvre 1989, Haeb-Umbach & Ney 1992, Beulen$^+$ 1995].
For a comprehensive overview of signal analysis methods for speech recognition see [Picone 1993]. More elaborate investigations on auditory model based feature extraction could, among others, be found in [Blomberg$^+$ 1984, Ghitza 1986,

Hunt & Lefèbvre 1987, Cohen 1989, Gao$^+$ 1992, Ohshima & Stern 1994].

The experiments reported in this work were performed with the feature extraction module of the RWTH speech recognition system, which is based on MFCCs [Davis & Mermelstein 1980]. Details on the short time spectral analysis, the subsequent normalization steps, the extraction of dynamic features, and the *linear discriminant analysis* (LDA) could be found in [Welling 1999] for sampling frequencies of both 16kHz and 8kHz (telephone bandwidth).

As part of this work, a generalization of the LDA will be presented, which is based on discriminative training criteria.

1.1.2 Acoustic Modeling

The aim of acoustic modeling is to provide a stochastic model of elementary speech units, such as words, subwords, or phonemes. These models are to capture both the acoustic and temporal characteristics of human speech. Acoustic models of individual words then are built by concatenating the models of subwords according to a pronunciation lexicon.

Depending on the task and the availability of training data, an inventory of subword units has to be defined, which could consist of words itself (the usual choice for digit recognition), partial words, syllables, phonemes, or phonemes in context. The standard choice for large vocabulary speech recognition are the so-called triphones, i.e. phonemes in right and left context. The advantage of introducing context is its potential to model specifically effects of coarticulation between adjacent speech units. In the RWTH continuous digit recognizer the subword units are the digits itself, whereas the subword units for the RWTH large vocabulary continuous speech recognizer are triphones.

In order to allow for proper modeling of the variation in speaking rate, *Hidden Markov Models* (HMM) have been established as *de-facto* standard in speech recognition [Baker 1975b, Rabiner 1989]. HMMs are stochastic finite automata that consist of a network of *states*, each of which models the characteristics of the acoustic features of a certain part of a subword unit.

For the example of the German word “sieben”, Fig. 1.2 illustrates the topology of the HMMs, as they are used in this work [Ney 1990]. The topology is strictly left-to-right, and subsequent states of the HMM are identical in pairs as indicated. Thus, a triphone HMM is given by six states, of which only three are different [Schwartz$^+$ 1985, Ney & Noll 1988]. The temporal characteristics of speech are captured by the transitions between states of the HMM. In the *Bakis* model [Bakis 1976] used here, only transitions from a state to itself (*loop*), to the next state (*forward*), and to the state following the next (*skip*) are allowed. With a frame shift of 10ms, the standard length of such a triphone HMM is 60ms, which approximately corresponds to the average length of a phoneme. Correspondingly, the minimum length, which the above HMM topology could model, is 30ms.
Using *Bayes*' rule [van Kampen 1992] and making the assumption that the temporal sequence of acoustic feature vectors follows a first order *Markov* process [van Kampen 1992],

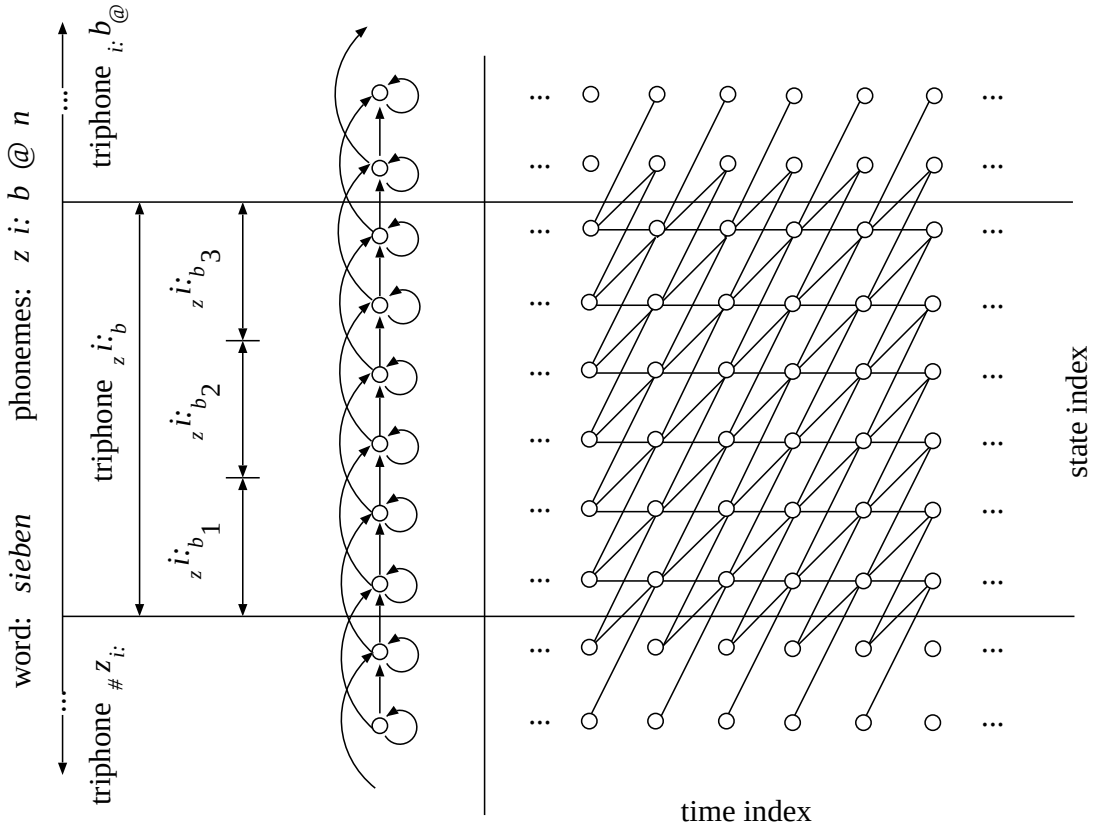

Figure 1.2: HMM in *Bakis* topology.

the acoustic emission probability $p(x_1^T|w_1^N)$ for a sequence of acoustic feature vectors, $x_1, \ldots, x_T$, given the word sequence $w_1, \ldots, w_N$ could be decomposed in the following way:

$$
\begin{aligned}
p(x_1^T|w_1^N) &\overset{\text{HMM}}{=} \sum_{s_1^T:w_1^N} p(x_1^T, s_1^T) \\
&\overset{\textit{Bayes}}{=} \sum_{s_1^T:w_1^N} \prod_{t=1}^{T} p(x_t|x_1^{t-1}, s_1^t) \cdot p(s_t|x_1^{t-1}, s_1^{t-1}) \\
&\overset{\text{1. order } \textit{Markov} \text{ process}}{=} \sum_{s_1^T:w_1^N} \prod_{t=1}^{T} p(x_t|s_t) \cdot p(s_t|s_{t-1}) \\
&\overset{\textit{Viterbi}}{\approx} \max_{s_1^T:w_1^N} \prod_{t=1}^{T} p(x_t|s_t) \cdot p(s_t|s_{t-1}). \qquad (1.1)
\end{aligned}
$$

The acoustic emission probability $p(x_t|s_t)$ denotes the probability to observe the acoustic feature vector x_t given state s_t, and the probability for the transition from state s_{t-1} to state s_t is denoted by $p(s_t|s_{t-1})$. As indicated in the last part of Eq. (1.1), the sum over all possible state sequences is often approximated by a corresponding maximization, which is commonly called the *Viterbi* approximation [Ney 1990].

Both the summation and the maximization could be efficiently evaluated using *forward-backward* (FB) calculations [Baum 1972, Rabiner & Juang 1986] or dynamic programming [Bellman 1957, Viterbi 1967, Ney 1984].

The acoustic emission probabilities could be modeled with discrete probabilities [Jelinek 1976, Liporace 1982], semi-continuous probability distributions [Huang & Jack 1989, Huang$^+$ 1990], or continuous probability distributions [Levinson$^+$ 1983, Ney & Noll 1988]. For the case of continuous probability distributions a wide range of distribution functions exist, e.g. Kernel densities, mixture densities, or even neural networks [Robinson & Fallside 1991]. Mixture densities are given by a weighted sum of continuous densities in the following way:

$$p(x_t|s_t) = \sum_l c_{sl} \cdot p(x_t|s_t, l),$$

where c_{sl} denotes the normalized mixture weight, with which the component density $p(x_t|s_t, l)$ contributes to the mixture of densities. Usually, the acoustic models are initialized by one single component density per state. Subsequently, the mixture densities are then iteratively splitted up in the training process.

Typical choices for the component densities are Gaussian or Laplacian densities; a systematic optimization of this choice could be found in [Chen$^+$ 1999]. Throughout this work, only mixture Gaussian densities will be considered.

In contrast to the case of digit recognition, where each digit is observed with sufficient frequency in the training corpus, the distribution of context dependent phonemes in a training corpus is very heterogeneous. For large and very large vocabulary speech recognition a large number of context dependent phonemes even is unobserved in the training corpus. Therefore the parameters of the acoustic model have to be tied together, in order to be able to obtain reliable estimations of these parameters [Young 1992]. One strategy for parameter tying is given by the bottom-up cluster algorithms [Dugast$^+$ 1995]. A disadvantage of bottom-up clustering is that those triphones, which were not observed in training, have to be considered separately. This is not the case for parameter tying approaches based on top-down clustering, where each triphone could be associated to an HMM state. In [Hwang$^+$ 1992, Young$^+$ 1994, Beulen$^+$ 1996, Beulen$^+$ 1997, Beulen & Ney 1998] top-down clustering of context dependent phonemes is realized by phonetic classification and regression trees. For the experiments on large vocabulary speech recognition presented in this work, only generalized triphones obtained by tree-based phonetic clustering were used.

In most speech recognition systems including the RWTH system, at least initially, the estimation of the parameters of the acoustic model is performed by the *Maximum Likelihood* (ML) principle in combination with the *Expectation Maximization* (EM) algorithm [Dempster$^+$ 1977]. The EM algorithm is an iterative procedure, that guarantees local optimality, given an initial parameter set.

An efficient implementation of the EM algorithm for speech recognition either applies the FB algorithm [Baum 1972, Rabiner & Juang 1986], or the *Viterbi* algorithm [Viterbi 1967, Ney 1984], if the maximum approximation is used for the evaluation of the HMMs.

The ML principle optimizes the acoustic models on the training data given the spoken word sequences. This does not explicitly take into account the decision boundaries between competing word sequences. In contrast to ML training, discriminative training criteria do take into account the decision boundaries, and are thus able to outperform ML training, especially if the model assumptions do not apply [Brown 1987, Normandin$^+$ 1994a].

Discriminative training and related topics will be the main focus of this work. Especially the optimal choice of the particular discriminative criterion and the corresponding optimization methods will be investigated. Moreover, discriminative training criteria will be used to optimize the structure of the acoustic models itself, by defining a new discriminative procedure for mixture density splitting.

1.1.3 Language Modeling

The aim of language modeling is to provide a statistical model of the syntax, semantics, and pragmatics of the speech that is to be recognized. Especially for large and very large vocabularies the language model contributes significantly to the performance of a speech recognition system.

In general, the language model is given by the prior probability $p(w_1^N)$ for word sequences w_1^N. Due to the combinatorial diversity of speech, this probability could not be estimated without further model assumptions. Using *Bayes*' rule, and, subsequently assuming that a word sequence follows an $(m-1)$-th order *Markov* process [van Kampen 1992], the prior probability of a word sequence could be decomposed as follows:

$$p(w_1^N) \overset{Bayes}{=} \prod_{n=1}^{N} p(w_n|w_1^{n-1}) \tag{1.2}$$

$$\overset{\text{(m-1). order } Markov \text{ process}}{=} \prod_{n=1}^{N} p(w_n|w_{n-m+1}^{n-1}). \tag{1.3}$$

For simplicity, define $w_{n-m+1}^{n-1} = w_1^{n-1}$ for $n < m$, as well as $w_{1-m+1}^{0} = \emptyset$. In language modeling, the order $(m-1)$ of the *Markov* process also is called the length of the history of a word; the corresponding language models are called m-gram language models [Bahl$^+$ 1983].

Similar to the ML principle for the estimation of the parameters of the acoustic model, the *perplexity* (PP) is defined as an evaluation measure and training criterion for language models [Bahl$^+$ 1983]. The perplexity of a test sample, i.e. of a word sequence w_1^N, is

defined as the geometric mean of the corresponding sequence of conditional language model probabilities:

$$\begin{aligned} PP &= \left[p(w_1^N)\right]^{-1/N} \\ &= \left[\prod_{n=1}^{N} p(w_n|w_{n-m+1}^{n-1})\right]^{-1/N}. \end{aligned}$$

For the estimation of an m-gram language model, the perplexity has to be minimized on the training corpus. Doing this, the estimation of the conditional language model probabilities could be performed on the basis of absolute frequencies of the corresponding sequences of m words, the m-grams. The number of possible m-grams increases exponentially with m; i.e. for a vocabulary of W words, the number of possible m-grams is W^m. Hence, especially for large vocabularies, a large number of m-grams will be unobserved. Therefore, smoothing methods have to be applied. These methods depend on combinations of the *discounting* and the *backing-off* principle [Katz 1987, Ney$^+$ 1994, Generet$^+$ 1995, Ney$^+$ 1997]. For *discounting*, probability mass is subtracted from observed m-grams, and is shifted to m-grams, which are unobserved in the training corpus. In *backing-off*, unobserved m-grams are substituted by generalized models with shorter word history. Better results are obtained, if *leaving-one-out*, a special variant of cross-validation, is used to estimate the smoothing parameters [Ney$^+$ 1994]. A systematic comparison of smoothing methods for language modeling could be found in [Martin$^+$ 1999].

Further improvements have been obtained using the following additional modeling schemes:

- *Cache* [Kuhn & de Mori 1990, Generet$^+$ 1995, Martin$^+$ 1997]: for certain words the probability of occurrence in a history of given length is taken into account.
- *Phrases (multi-words)* [Jelinek 1991, Klakow 1998]: frequent groups of consecutive words are modeled as single words.
- *Word classes* [Brown$^+$ 1992, Kneser & Ney 1993, Jardino 1996, Martin$^+$ 1998]: instead of modeling each word separately, word classes are used.
- *Distant m-grams* [Rosenfeld 1994, Martin$^+$ 1999]: language models are built by combining m-grams of varying history length m as well as m-grams with gaps.

A description of the implementation of the language models in the RWTH recognition system, as it is used in this work, is found in [Wessel$^+$ 1997].

Although improvements in the perplexity of a language model do fairly well correspond to improvements in word error rate for speech recognition, it has been found that this is not always true. Therefore, recognition experiments are gradually established as a further, and more reliable evaluation measure for language models.

Although the estimation of the language models itself is not considered in this work, language models need to be used for discriminative training of the acoustic models. Therefore, the *choice* of language models for discriminative training is investigated as part of this work.

1.1.4 Search

The aim of the search procedure is to apply *Bayes*' decision rule, and, as illustrated in Fig. 1.1, find the word sequence that maximizes the *a-posteriori* probability for a given sequence of acoustic feature vectors:

$$\begin{aligned}\left[w_1^N\right]_{\text{opt}} &= \underset{w_1^N}{\operatorname{argmax}}\, p(w_1^N|x_1^T) \\ &= \underset{w_1^N}{\operatorname{argmax}} \left[p(w_1^N) \cdot p(x_1^T|w_1^N)\right]\end{aligned}$$

For a vocabulary of W words and a maximal sentence length of N, the number of possible word sequences is exponential in N:

$$1 + W + W^2 + W^3 + \ldots + W^N = \frac{W^{N+1} - 1}{W - 1}.$$

In addition to the search problem in the space of possible word sequences, for each word sequence the acoustic and the language model have to be evaluated. Substituting the acoustic emission probability from Eq. (1.1), and the language model probability from Eq. (1.2) into the decision rule, the following optimization problem is obtained:

$$\begin{aligned}\left[w_1^N\right]_{\text{opt}} \quad &= \quad \underset{w_1^N}{\operatorname{argmax}} \left\{ \left[\prod_{n=1}^{N} p(w_n|w_1^{n-1}) \right] \cdot \left[\sum_{s_1^T : w_1^N} \prod_{t=1}^{T} p(x_t|s_t) \cdot p(s_t|s_{t-1}) \right] \right\} \\ &\overset{\textit{Viterbi}}{\approx} \quad \underset{w_1^N}{\operatorname{argmax}} \left\{ \left[\prod_{n=1}^{N} p(w_n|w_1^{n-1}) \right] \cdot \left[\max_{s_1^T : w_1^N} \prod_{t=1}^{T} p(x_t|s_t) \cdot p(s_t|s_{t-1}) \right] \right\}. \quad (1.4)\end{aligned}$$

The complexity of this search problem could be largely reduced by applying the *Viterbi* approximation, and by using dynamic programming. In dynamic programming, the mathematical structure of the search problem is used to split the global search problem into successive local search problems.

There currently exist two different algorithms to solve the search problem by dynamic programming, the *Viterbi*-search, and the A^*-search (or *stack-decoding*). In A^*-search, a time-*asynchronous* expansion procedure of states and words is applied, which depends on a heuristic measure that produces strictly overestimated probabilities for the remaining, unexpanded part of a hypothesis [Jelinek 1969, Paul 1991]. The performance

of A^*-search heavily depends on the quality of this heuristic. On the other hand, in *Viterbi*-search, the process of evaluating and expanding state hypotheses is performed time-*synchronous* [Vintsyuk 1971, Baker 1975a, Sakoe 1979, Ney 1984]. Since the probabilities of different hypotheses up to a certain time frame could be compared with each other, intermediate pruning of unlikely hypotheses is possible, in order to further reduce the search space.

The following methods could be applied to reduce the computational complexity of the time-synchronous *Viterbi* search, especially for large vocabulary applications:

- *Lexical prefix tree* [Ney$^+$ 1992, Ney 1993, Ortmanns$^+$ 1997b]: the pronunciation lexicon is organized into a tree structure. Since a considerable amount of the search effort is due to the first phoneme-models of the words, this leads to a large reduction in search space.

- *Beam-search* [Lowerre 1976, Ney$^+$ 1987, Ortmanns & Ney 1995]: the search space is reduced to the most likely hypotheses at certain intervals in the procedure of expanding and evaluating hypotheses (usually every time frame). This procedure clearly is sub-optimal, but the pruning parameters could be adjusted such that no significant search errors do occur.

- *Look-ahead*: the beam-search procedure could be further optimized by calculating approximate estimates of future hypotheses. Using a lexical prefix tree organization, the identity of the hypothesized words is generally not known until a word end is reached. Hence, for the purpose of *language model look-ahead*, upper estimates of the language model probabilities are propagated backwards in the lexical prefix tree [Steinbiss$^+$ 1994, Odell$^+$ 1994, Alleva$^+$ 1996, Ortmanns$^+$ 1996a]. Similarly, for the purpose of *phoneme look-ahead*, a number of successor phoneme hypotheses are evaluated approximately [Ney$^+$ 1992, Haeb-Umbach & Ney 1994, Ortmanns$^+$ 1996b].

- *Fast likelihood computation*: high-performance speech recognizers usually use mixture densities with a very high number of densities, which leads to a high additional search effort. This could be significantly reduced by structuring the corresponding search space [Ramasubramansian & Paliwal 1992, Fritsch 1997], by quantization of the corresponding feature vectors [Bocchieri 1993, Ortmanns$^+$ 1997c], or by a partitioning of the corresponding feature space [Nene & Nayar 1996], which even might utilize the importance of the particular features [Ortmanns$^+$ 1997c].

The aim of multiple-pass search architectures, as opposed to integrated or one-pass search architectures, is, to apply more simple acoustic and language models for the first pass, in order to reduce the computational complexity of the search, and store the most likely recognition results for further rescoring with more accurate models. The two main strategies to represent and store the most likely word sequence hypotheses of an intermediate recognition pass for further processing are *N-best lists* and *word graphs*. In N-best lists, the N word sequences with the highest *a-posteriori* probabilities are stored (e.g. [Schwartz & Chow 1990, Schwartz & Austin 1991, Oerder & Ney 1993,

Woodland$^+$ 1995]), whereas in word graphs the intermediate word sequences are stored in a directed, acyclic graph, where the arcs are identified with word hypotheses (e.g. [Schwartz & Austin 1991, Ney & Aubert 1994, Aubert & Ney 1995, Woodland$^+$ 1995, Ortmanns$^+$ 1997a]).

A comprehensive overview of efficient search approaches for large vocabulary continuous speech recognition could be found in [Ortmanns 1998].

During discriminative training, alternative word sequences have to be determined, which is usually done by a corresponding recognition pass on the training data. For large vocabulary discriminative training, the resulting alternative word sequences could efficiently be represented, and further processed using word graphs [Valtchev$^+$ 1996, Valtchev$^+$ 1997].

During each discriminative training iteration, new sets of alternative word sequences have to determined. As part of this work, a fast algorithm for large vocabulary speech recognition constrained to word graphs is presented for the special purpose of efficient discriminative training.

Eq. (1.4) represents the usual way, the *a-posteriori* probability is decomposed for the application of dynamic programming based recognition schemes. In this work, an alternative decomposition of the *a-posteriori* probability will be presented, which further improves the recognition performance, without altering the underlying models.

1.2 Discriminative Training: State of the Art

In the past years of intense research in the field of statistical speech recognition, the *Maximum Likelihood* (ML) principle has evolved as the state of the art training criterion most speech recognition systems are trained with at least initially. Given training observations with their corresponding correct class assignments, i.e. for speech recognition the training utterances, consisting of acoustic observations and the corresponding spoken word sequences, the ML criterion estimates the model parameters such that the class conditional probability of the training data is maximized. Hence, for a given training observation ML training only considers the corresponding correct classes. In general, depending on the underlying class conditional probability model, ML training does not explicitly take into account the overlap of the class boundaries.

1.2.1 Discriminative Criteria

Provided the statistical models are correct and enough training data is available, it could easily be shown that ML training leads to class conditional probability models that are equal to the correct distributions. Under realistic conditions, the amount of training data available is usually limited. In addition, even if correct model assumptions could be found, the model complexity is limited by the amount of training data.

There is some empirical evidence [Brown 1987, Normandin$^+$ 1994a] that, discriminative training criteria, as opposed to ML training, are in a better position to handle the case of incorrect model assumptions. Discriminative criteria not only try to maximize the class conditional probability of the training data given the correct classes, but also try to minimize the corresponding class conditional probabilities of the alternative classes. Thus discriminative training optimizes class separability or recognition performance respectively.

1.2.1.1 Maximum Mutual Information Criterion

Some of the first experiments with discriminative training for HMM based speech recognizers were presented in [Bahl$^+$ 1986]. Therein, the *Maximum Mutual Information* (MMI) criterion was applied to a speaker dependent isolated word recognition system with a vocabulary of 2000 words. Acoustic emission probabilities were modeled by discrete distribution HMMs. In comparison to ML training a relative improvement in word error rate of 18% was reported.

In [Brown 1987], both discrete and continuous acoustic emission distributions were trained on a speech recognition task for isolated letters, the English *E*-set. Several approaches for acoustic modeling were compared using ML and MMI training. Starting with the best ML result, a relative improvement of nearly 18% in recognition rate was obtained using MMI training.

Similar results were obtained for MMI training of discrete HMMs for speaker dependent phoneme recognition in [Merialdo 1988]. Relative improvements of about 23% in phoneme error rate were obtained in comparison to ML training.

As shown in [Valtchev 1995], the MMI criterion is equivalent to the Conditional Maximum Likelihood criterion presented in [Nádas$^+$ 1988], if the language model is supposed to be given.

In [Chow 1990], MMI was used to train discrete HMMs and codebook exponents for continuous speech recognition on the *Resource Management* corpus with a vocabulary of 1000 words, but only minor improvements in error rate were obtained.

In [Normandin 1991], the MMI criterion was applied both to discrete and continuous HMMs. For the case of continuous digit string recognition on the *TI digit string* corpus relative improvements of up to nearly 50% in string error rate in comparison to ML training were obtained depending on the complexity of the underlying acoustic model. The same methods were applied to large vocabulary speech recognition (*Air Travel Information System*) with a vocabulary of about 2000 words, but no significant improvements in word error rate were obtained [Normandin$^+$ 1994b].

Later, Valtchev *et. al.* [Valtchev$^+$ 1996, Valtchev$^+$ 1997] presented results applying MMI training to a continuous mixture HMM speech recognition system with a vocabulary of 64k words on the WSJ database and obtained relative improvements in word error rate

of 5–10% in comparison to ML training.

In [Bahl$^+$ 1996, Povey & Woodland 1999], modified discriminative approaches have been introduced. The computationally expensive discriminative model of alternative or competing word hypotheses was replaced with a model on a frame by frame basis. The modified frame based MMI criterion lead to results comparable with the word based MMI criterion, but with significantly reduced computational complexity.

1.2.1.2 Minimum Classification Error Criterion

An alternative class of discriminative training criteria, represented by the *Minimum Classification Error* (MCE) principle was introduced in [Juang & Katagiri 1992]. As the name suggests, the MCE criterion is intended to minimize the error rate on the training data for a given pattern recognition system. As a training criterion, the actual error rate is a discontinuous function of the parameters of the underlying recognizer model. Therefore, in order to allow for parameter optimization using gradient descent based methods, the MCE criterion is a smoothed version of the empirical error rate on the training data, in order to obtain a continuous and differentiable objective function.

The kind of error rate, which is represented by the MCE criterion, depends on the task in question. For continuous phoneme recognition on the *TIMIT* task, relative improvements in phoneme error rate between 5% and 14% were reported in [McDermott & Katagiri 1997] for phoneme level MCE in comparison to ML training. For continuous digit string recognition on the *TI digit string* corpus, relative improvements in string error rate of more than 25% were obtained in [Chou$^+$ 1993, Chou$^+$ 1994] for string level based MCE training in comparison to ML training. In [Saul & Rahim 2000], MCE training on small and medium-sized vocabulary tasks lead to relative improvements in word error rate of about 10%.

In another application, a *Minimum* Verification *Error* (MVE) criterion was used to train and adapt speaker verification models. In this task the false acceptance and false rejection rates as well as the equal error rates were reduced by more than 25% for MVE training in comparison to ML training and by about 50% for MVE adaptation in comparison to *Maximum A-Posteriori* adaptation when beginning with ML trained initial models.

In speech recognition, due to the additional word boundary optimization and the HMM-state alignment problem, the MCE criterion usually represents the smoothed empirical *sentence* error rate. It could be argued that this is not optimal, if the objective is to minimize the *word* error rate for a given task. Correspondingly, experiments showed that, minimizing the word error rate on the training data for an isolated word recognition task gave better word error rates on the test data than, a minimization of the HMM-state error rate on the training data [Bauer 1998]. Although *Minimum* state *error* training lead to minor improvements in word error rate only, word based MCE training lead to a 50% reduction in word error rate on the test data. Interestingly, in case of the word based MCE criterion, the state error rate on the training data even increased in

comparison to ML training [Bauer 1998].

Correspondingly, in another experiment it was shown that phoneme based MCE training performed better than phoneme string level MCE if the objective is to optimize the phoneme recognition rate [McDermott & Katagiri 1997].

1.2.1.3 Comparison of Criteria

In most studies on discriminative training criteria, the performance of discriminative training usually is compared to ML training, which is the standard initialization for a discriminative training procedure. On the other hand, information on direct comparisons between discriminative criteria is very limited.

One approach to compare MMI and phoneme string level MCE training was presented in [Reichl & Ruske 1995] for the case of continuous phoneme recognition on the *Phondat "Diphon"* corpus for continuous German speech. In this investigation, MMI training gave a relative improvement of 7% in phoneme error rate compared to ML training. In contrast to this, MCE training gave a relative improvement of 13% in comparison to ML training. Hence, on this task MCE training outperformed MMI training by about 7% relatively.

1.2.2 Parameter Optimization for Discriminative Criteria

To the authors knowledge, no optimization methods proved to converge for discriminative training criteria in practical conditions are known up to now. Methods used up to now, for which convergence has been shown empirically, are gradient descent based methods, and an extended version of the *Baum-Welch* algorithm.

1.2.2.1 Gradient Descent

For the MCE criterion, gradient descent usually is the optimization method of choice, e.g. [Chou$^+$ 1992, Juang & Katagiri 1992, Paliwal$^+$ 1995]. In order to get fast and reliable convergence, some care is needed when choosing appropriate step sizes [Chou$^+$ 1992].

For the MMI criterion, first studies also made use of gradient descent optimization [Bahl$^+$ 1986]. Furthermore, in [Merialdo 1988], approximations to the gradients involved in order to improve reduced convergence caused by low valued probabilities were introduced for the first time.

In [Valtchev 1995], a series of first and second order gradient based optimization methods and learning rate update rules are discussed.

1.2.2.2 Extended *Baum-Welch* Algorithm

A few years after the first applications of MMI training for speech recognition were published, the *extended Baum-Welch* (EB) algorithm, an extension to the standard *Baum-Welch* algorithm designed for optimization of the MMI criterion was

introduced. Initially, EB was developed for discriminative training of discrete probabilities [Normandin & Morgera 1991, Cardin$^+$ 1993, Normandin$^+$ 1994a]. For this case a proof of convergence was presented in [Gopalakrishnan$^+$ 1991], although the corresponding iteration constants lead to impractically low convergence rates.

Nevertheless, empirical expressions for the iteration constants were found, which lead to fast and reliable convergence [Gopalakrishnan$^+$ 1991, Normandin$^+$ 1994a]. For discrete probabilities, as proposed for gradient descent in [Merialdo 1988], Normandin [Normandin 1991] suggested similar changes to the EB reestimation formulas in order to prevent reduced convergence caused by small valued probabilities.

The EB algorithm was extended to the optimization of continuous probabilities in [Normandin 1991]. For the continuous case no proof of convergence is known up to now. The iteration constants, for which convergence could be proven in the discrete case map to infinity in the continuous case [Normandin 1991].

In [Kapadia$^+$ 1993] comparative experiments for phoneme recognition on the TIMIT database were performed, for which the EB algorithm performed better than gradient based methods. Consequently, later the EB algorithm was also chosen for MMI training on the *Wall Street Journal* task with a recognition vocabulary of 65k words [Valtchev$^+$ 1996, Valtchev$^+$ 1997].

1.2.3 Efficiency of Discriminative Training

When performing discriminative training a crucial problem is the computation and processing of the alternative word hypotheses defining the competing models in contrast to the correct models based on the spoken word sequences. In every iteration step of a discriminative training procedure, the discriminative model has to be newly estimated based on the current model parameters.

1.2.3.1 Time Alignment

Without using any further approximations, for each iteration of discriminative training a recognition pass on the complete training data needs to be performed. This includes *forward-backward* paths for every recognized and statistically relevant word sequence coming out of this recognition pass, in order to calculate the corresponding HMMs for the competing model. For digit recognition, this was done in [Normandin 1995, Normandin 1999].

As shown in [Bridle 1990], there exists a certain interpretation of HMMs as *Recurrent Neural Networks* (RNN). Strong relations were indicated between backpropagation training, and the backward pass of the *Baum-Welch* optimization of the ML criterion, which is also part of the EB algorithm.

For the application of discriminative training to large vocabulary speech recognition, the combinatorial variety of possible word sequences becomes so high that for this case the

Viterbi-approximation [Ney 1990] was applied in [Valtchev$^+$ 1997].

To the authors knowledge, results showing the impact of the *Viterbi*-approximation have been published neither for ML nor for MMI training up to now.

1.2.3.2 Alternative Word Sequences

For discriminative training of phoneme and small vocabulary speech recognition systems, the alternative phoneme or word sequences are usually determined by a full recognition pass [Chou$^+$ 1992, Merialdo 1988, Normandin 1996].

For large vocabulary speech recognition this clearly becomes unfeasible. Therefore, in [Valtchev$^+$ 1997], a full recognition pass was performed only once before discriminative training to produce initial word lattices. After each iteration step, these were used to perform an acoustic rescoring, where the word boundary times from the initial lattices were not used [Valtchev 1999].

In order to reduce the computational complexity of the corresponding time-alignment or *forward-backward* calculations, many approaches only use one competing word sequence per utterance. For the MCE criterion, this means that only the best *incorrectly* recognized word sequence is used as competition [Juang & Katagiri 1992]. In the case of MMI training it means that only the best recognized word sequence is used as competition against the spoken one, which is called *Corrective Training* (CT) [Normandin 1996]. Performing CT, only misrecognized training utterances make contributions to reestimation [Normandin 1996].

In [Chow 1990], as a further step, N-best lists were used for discriminative training with a 1000 word vocabulary. Provided the number of words to be considered as alternatives for a spoken word is given, the number of items needed in the corresponding N-best list clearly would increase exponentially with the number of spoken words in one utterance. In order to overcome this, looped lattices were used instead of N-best lists in [Normandin$^+$ 1994b]. Moreover, a *forward-backward* algorithm working on the hypotheses of word lattices in order to efficiently process the alternative word hypotheses for large vocabulary discriminative training was presented in [Valtchev$^+$ 1996].

A different objective for optimizing the training time was followed in [Bahl$^+$ 1996, Povey & Woodland 1999]. Instead of finding efficient algorithms for determining and processing the alternative word hypotheses, the marginal distribution of the acoustic observations itself is approximated. In this approximation the recognition model is replaced by a frame based model incorporating only those emission probabilities, which are most important for discrimination of a particular frame. The improvements compared to ML training are similar to the standard word based MMI approach, whereas the computational efficiency is largely improved.

1.2.4 Discriminative Modeling

Discriminative training introduces certain aspects according to the underlying statistical models, which are not relevant to or not part of ML training.

1.2.4.1 Language Models and Training

As one new aspect, discriminative criteria in several perspectives introduce language models to training. Firstly, the language model for the - at least initial - recognition pass are to be chosen, which have impact on the word sequences to be considered as alternatives. No investigations related to this are known to the author so far. Secondly, the choice of language models for discriminative training itself should have impact on the resulting models. Finally the question arises, in how far training results obtained with a particular language model generalize to the recognition with other language models on a given task.

In [Chou$^+$ 1993], for a 1000-word vocabulary task it was found that, using no language model in MCE training gave better results than using a word pair grammar in training, where in both cases a word pair grammar was used for evaluation.

In [Valtchev$^+$ 1997], a bigram language model was used to discriminatively train a large vocabulary speech recognizer. It could be seen that, for a given language model improvements in comparison to the baseline ML results diminish when using language models with increasing context length.

1.2.4.2 Model Evaluation

An important observation is that the improvements obtained by discriminative training methods in comparison to conventional Maximum Likelihood (ML) training increase with decreasing model complexity [Valtchev$^+$ 1996]. Therefore one would expect discriminative training criteria to be a good candidate to give an estimation of the ability of an acoustic model to describe the same data. For example, it is not possible to train codebook exponents for coexisting acoustic models using ML training, since ML would simply choose the best scoring acoustic model while neglecting all others. On the other hand it was shown in [Chow 1990, Normandin 1996] how discriminative training of codebook exponents could both improve ML and discriminatively trained models.

The standard approach to acoustic modeling in speech recognition uses HMMs in combination with continuous mixture densities. A discriminatively based modeling approach could therefore try to balance the need for high acoustic resolution with the limited amount of training data by evaluating the models with respect to their corresponding parameter numbers.

In [Bahl & Padmanabhan 1998], a discriminative measure for model complexity evaluation was introduced. For each HMM mixture model this measure was used to choose the optimal model from ML trained models with different number of densities. It was shown that nearly the same recognition results could be obtained with significantly lower

numbers of parameters.

Furthermore, in [Normandin 1995], an HMM mixture splitting algorithm fully based on the MMI criterion is introduced. Therein, a discriminative measure derived from the MMI criterion is used to choose those densities, which are to be splitted up. This was then combined with MMI training of the corresponding models. For connected digit recognition, significant improvements in sentence error rate were observed with this approach when compared to both ML, and subsequently MMI trained models of even higher complexity. A similar approach was chosen in [Valtchev$^+$ 1997] for large vocabulary speech recognition with similar success, although the approach was not pursued up to optimal model complexities.

1.2.4.3 Feature Transformation

The determination of feature transformations includes both the training of transformation parameters and the choice of the resulting appropriate features. Therefore it builds a connection between optimization and model evaluation. Several approaches based on varying degrees of discrimination have been proposed to determine linear and neural network based feature transformations.

Linear discriminant analysis (LDA) has been established as a standard feature extraction method for speech recognition [Hunt & Lefèbvre 1989, Haeb-Umbach & Ney 1992, Beulen$^+$ 1995]. LDA is based on a class separability measure and the corresponding optimization problem could be solved explicitly for certain cases. For the application of LDA to a specific task, the appropriate class definition has to be optimized individually [Haeb-Umbach & Ney 1992]. Using LDA, usually relative improvements around 10% could be obtained in comparison to using untransformed acoustic features [Beulen$^+$ 1995].

Major variations to the LDA approach investigated so far are to either replace the linear transformation by a neural network and/or replace the class separability criterion by a mutual information or smoothed error rate based criterion.

In [Bengio$^+$ 1992], a hybrid phoneme recognition system combining neural networks with HMMs has been introduced, where the outputs of an artificial neural network (ANN) constitute the acoustic feature vectors for the HMM. On the *TIMIT* task a relative reduction in phoneme error rate of 30% has been obtained compared to a HMM system without using a neural network.

In [Rigoll & Willett 1998], multi layer perceptrons and recurrent neural networks were trained to optimize the mutual information between acoustic observations and HMM states on basis of a precomputed optimal state alignment. In applying this approach to the *Resource Management* corpus with a vocabulary of 1000 words, relative improvements of about 10% in word error rate were obtained.

A different approach was followed in [Reichl$^+$ 1995], where a class discrimination measure was used to train a multi layer-perceptron, whose output nodes were interpreted as *a-*

posteriori probabilities of the corresponding states, i.e. the HMM states of the underlying phoneme recognition task. The dimension of the resulting features was then reduced to the dimension of the original features using *Principal Component Analysis*. Relative improvements of up to 7% in phoneme recognition rate were obtained in comparison to the standard LDA approach.

Since LDA is defined on a per-frame basis, it is not clear how word discrimination could be included into the approach. In order to find a criterion based on word error rate, in [Ayer$^+$ 1993] an extension to LDA based on the MCE criterion was proposed. On a British English E-set task relative improvements in error rate of nearly 50% were reported, although several hundred training iterations were needed to obtain this result.

In another approach, the MMI criterion was applied to jointly optimize HMM and recurrent feature transformation parameters on an E-set recognition task [Valtchev$^+$ 1993]. Probably due to under-training, no improvements were obtained in comparison to the best MMI trained models. Consequently, significant improvements were reported for reduced model complexities, showing the capabilities of the proposed method.

In a similar approach, the MCE criterion was used to jointly train HMM model parameters and a multi layer perceptron for feature transformation [Rahim & Lee 1996]. On a telephone based digit recognition task, this approach lead to relative improvements of 16% in comparison to the best system without feature transformation. In a further development, the approach was extended to incorporate both linear and non-linear feature transformations, which subsequently were combined linearly [Rahim$^+$ 1997]. On a digit recognition task, in comparison to MCE training without feature transformation, relative improvements in word error rate of 5% were obtained using linear transformation alone and an HMM set with mixture densities and 4 densities per mixture. Combining a linear transformation with a neural network for feature transformation did not further increase this improvement. Using transformations for each digit and silence model gave increased relative improvements of 13% for linear transformation alone, and 22% for the combination of linear and non-linear feature transformation.

Chapter 2

Scientific Goals

An automatic speech recognizer consists of three main building blocks: the acoustic model, the language model and the search algorithm. Of these, the acoustic model ensures the connection between the acoustic level and the word level of speech, and its quality is essential so as to obtain high recognition performance. Discriminative training criteria, as opposed to the standard maximum likelihood approach, directly take into account the connection between the underlying models and the recognition performance of a speech recognizer. Therefore discriminative training represents an important alternative to standard training methods.

The following insights on the current state of the art of discriminative training can be derived from the literature:

1. Several criteria, especially the *Maximum Mutual Information* (MMI) and the *Minimum Classification Error* (MCE) criterion have been investigated, but as yet no systematic comparisons have been performed.

2. No parameter optimization method is known so far, which guarantees convergence in all cases under realistic conditions. Methods used are the *extended Baum* (EB) algorithm for the MMI criterion, and gradient descent, but no standard optimization procedure has been established so far.

3. Due to the determination and processing of competing word hypotheses, discriminative criteria still need considerable higher training times than standard ML training, especially for large vocabulary applications.

4. Discriminative training usually involves the use of language models for training, i.e. for the recognition of competing word hypotheses and the discriminative training criterion itself. No investigations have been made so far to clarify the effect of the choice of language models for training in relation to the language models chosen for recognition.

5. There is evidence that discriminative training criteria are especially well suited to train models of low complexity or models, for which model mismatch occurs. Therefore discriminative training criteria should be better suited for comparative model evaluation than the *Maximum Likelihood* (ML) criterion.

6. It has been shown that for infinite amounts of training data both ML and MMI training will lead to the correct distributions.

The aim of this work is to build up a framework for efficient discriminative training and modeling so as to enhance both small and large vocabulary continuous speech recognition. Especially, the following novel approaches to discriminative training for speech recognition will be presented:

I. *Development of a unified framework for a class of discriminative training criteria and optimization methods:*

   In a first step, a class of discriminative training criteria will be merged into a unifying formulation. On this basis several discriminative criteria will be compared analytically and experimentally. In addition, close similarities between the parameter optimization methods gradient descent and *extended Baum* (EB) algorithm will be derived analytically and proved experimentally.

II. *Asymptotic behaviour of MCE and related criteria:*

   The relation between the true *Bayes* error rate and the MCE criterion will be investigated and a closed form solution for the optimization of the MCE criterion will be presented in the asymptotic case of infinite training data. The same analysis will be made for a novel discriminative training criterion, the *generalized Gini* (GG) criterion, which is introduced in this work.

III. *Evaluation of several levels of approximation for discriminative training:*

   Several levels of approximation will be evaluated by comparison to the corresponding exact realizations: *Viterbi* alignment vs. *forward-backward* algorithm for ML and MMI training; determination of alternative word sequences by rescoring vs. constrained recognition using word graphs; and the effect of *Corrective Training* respectively.

IV. *Investigation of the interdependence between language models and discriminative training:*

   For discriminative training of large vocabulary speech recognizers, the effect of the choice of language models for training and its correlation to the language model choice for recognition will be investigated. Therefore extensive experiments using several language models for training and recognition will be performed with special attention to the choice of context length.

V. *Development of efficient modeling approaches for mixture density splitting and linear feature transformation using discriminative training criteria:*

   A hybrid mixture density splitting algorithm will be presented, which combines the model evaluation abilities of discriminative training criteria with the computational efficiency of ML training. Furthermore, a discriminative algorithm for feature transformation with closed reestimation formula will be derived and evaluated.

VI. *Introduction of an alternative scoring algorithm for speech recognition:*

   Finally, inspired by discriminative training methods using word graphs, a novel evaluation method for the stochastic models used in speech recognition is suggested. The algorithm is based on an alternative decomposition of *Bayes*' decision rule, and takes advantage of summations over word boundaries.

The remaining part of this work will be organized as follows. In Chapter 3, a unified formulation of a class of discriminative training criteria will be presented together with qualitative and quantitative comparisons, as well as analytical investigations of several training criteria. This is followed up by a discussion of optimization methods in Chapter 4. In Chapter 5, topics of discriminative training specific to speech recognition will be addressed. Furthermore, efficiency of discriminative training is discussed in Chapter 6. Based on methods derived for discriminative training, in Chapter 7 an alternative decomposition of *Bayes*' decision rule is presented. Finally efficient approaches to acoustic modeling, i.e. discriminative splitting and feature transformation are treated in Chapter 8. This work is finished by a summary of the scientific contributions and an outlook in Chapters 9 and 10.

For structural clarity, experimental results, which are significant to particular topics always are presented in the corresponding sections, even though most results depend on findings presented in more than one section. Details on the particular speech corpora and the corresponding baseline speech recognition systems used in this work are summarized in Appendix A.

Chapter 3

Discriminative Training Criteria

In an increasing number of applications discriminative training criteria such as *Maximum Mutual Information* (MMI)[1] [Ben-Bassat 1982, pp. 785], [Devijver & Kittler 1982, pp. 262], [Bahl$^+$ 1986] and *Minimum Classification Error* (MCE) [Juang & Katagiri 1992, Chou$^+$ 1993] have been used. In MCE training, an approximation to the error rate on the training data is optimized, whereas MMI optimizes the *a posteriori* probability of the training utterances and hence the class separability. Here, a formally unifying approach for a class of discriminative training criteria including the MMI and the MCE criterion will be presented [Schlüter & Macherey 1998, Schlüter$^+$ 2000], thus extending a comparison done in [Reichl & Ruske 1995].

3.1 Unifying View

The training data shall be given by training utterances r with $r = 1, ..., R$, each consisting of a sequence X_r of acoustic observation vectors $x_{r1}, ..., x_{rt}, ..., x_{rT_r}$ and the corresponding sequence $W_r = w_{r1}, ..., w_{rt}, ..., w_{rN_r}$ of N_r spoken words. The *a posteriori* probability for the word sequence W_r given the acoustic observation vectors X_r shall be denoted by $p_\theta(W_r|X_r)$. Similarly, $p_\theta(X_r|W_r)$ and $p(W_r)$ represent the corresponding emission and language model probabilities respectively. In the following, the language model probabilities are supposed to be given. Hence the parameter θ represents the set of all parameters of the emission probabilities $p_\theta(X_r|W_r)$. Finally, let $\mathcal{M}_r$ denote the set of word sequences, which are considered for discrimination in utterance r. A class of discriminative training criteria $\mathcal{F}$ including MMI and MCE could then be defined by the expression:

$$\mathcal{F}(\theta; f, \alpha, \{\mathcal{M}\}) = \frac{1}{R} \sum_{r=1}^{R} f\Big(\log \frac{p_\theta^\alpha(X_r|W_r) \cdot p^\alpha(W_r)}{\sum_{W \in \mathcal{M}_r} p_\theta^\alpha(X_r|W) \cdot p^\alpha(W)} \Big) \tag{3.1}$$

The choice of the set of alternative word sequences, together with the optional smoothing function f and the optional weighting exponent α determine the choice of the particular criterion.

[1] The MMI criterion, as it is usually used in speech recognition, should correctly be referred to as the *Equivocation* criterion. Both criteria are equivalent only in the case of given class priors.

3.1.1 Choice of particular criteria

In Table 3.1, examples for the choice of $\mathcal{M}_r$, f, and α are listed for the MMI and the MCE criterion as well as the corrective training (CT) and the falsifying training (FT) criterion. The ML criterion is also contained in the unified approach and therefore is listed, too. Moreover, the novel discriminative *generalized Gini* (GG) criterion is added, which is introduced in Section 3.3.

Table 3.1: Several discriminative criteria, which are contained in the unified formulation defined in Eq. (3.1).

criterion	smoothing function $f(z)$	alternative word sequences in $\mathcal{M}_r$	exponent α
ML	identity	–	–
MMI	identity	all (recognized)	1
CT		best (recognized)	(∞)
MCE	$-\frac{1}{1+e^{2\varrho z}}$	all *without* W_r	*free*
FT		best (recognized) $\neq W_r$	(∞)
GG	$-(1-e^{z})$	all	*free*

It should be noted that the sign of the smoothing function of the MCE, FT and GG is chosen to be negative without loss of generality, since the unified criterion is chosen to be *maximized* for parameter optimization.

In contrast to the ML criterion, all discriminative criteria have in common that the correct models are not only optimized by themselves, but at the expense of some discriminative models. This means that the joint probabilities of the acoustic observations and the *spoken* (correct) word sequences of the training utterances are "discriminated" against the sum over the corresponding joint probabilities of the set of alternative word sequences $\mathcal{M}_r$ defined for each training utterance r. $\mathcal{M}_r$ is often called the discriminative model of a training utterance.

3.1.2 Smoothing

In general the smoothing function f leads to a weighting of the contribution of whole training utterances. This utterance weighting depends only on the discriminative distance measure $\log \rho_\theta(W_r|X_r)$ between the spoken and the set of alternative word sequences,

$$\log \rho_\theta(W_r|X_r) = \log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}. \tag{3.2}$$

It should be noted that in the case of $\alpha = 1$ the discriminative distance measure $\log \rho_\theta(W_r|X_r)$ is equal to the logarithm of the corresponding *a-posteriori* probability for

the spoken word sequence, provided that the set $\mathcal{M}_r$ of alternative word sequences consists of all essential alternative word sequences including the spoken word sequence W_r itself.

3.2 Comparison and Other Criteria

3.2.1 Minimum Error Based Criteria

As indicated in Table 3.1, using the sigmoid function,

$$f(z) = -\frac{1}{1+e^{2\varrho z}},$$

yields an equivalent variant of the MCE criterion, which is to be maximized (cf. Table 3.1),

$$\mathcal{F}_{MCE}(\theta) = -\frac{1}{R}\sum_{r=1}^{R} \frac{1}{1+\Big[\dfrac{p_\theta^\alpha(X_r|W_r)\cdot p^\alpha(W_r)}{\displaystyle\sum_{W\neq W_r} p_\theta^\alpha(X_r|W)\cdot p^\alpha(W)}\Big]^{2\varrho}}.$$

Ideally, the MCE and FT criterion would represent the sentence error rate on the training data. Especially for FT, the argument of the smoothing function is the score difference of the spoken word sequence and the best recognized word sequence different from the spoken word sequence. Therefore, if f would be a step function, the FT criterion would represent the sentence error rate on the training data, since the score difference is lower than zero for correctly recognized utterances and greater zero otherwise. In order to obtain a criterion, which is differentiable with respect to the acoustic parameters θ, a smoothed version of the step function is chosen for f instead, i.e. f is usually given by a sigmoid function. Moreover, for the MCE criterion not only the best recognized, but all (recognized) word sequences excluding the spoken word sequence are chosen for training, in order to obtain a further smoothing effect between the alternative word sequences with scores near to the best recognized word sequence.

3.2.2 Criteria related to Maximum Mutual Information

On the other hand, choosing $\alpha = 1$ and $f(z) = z$ yields the MMI criterion,

$$\begin{aligned}\mathcal{F}_{MMI}(\theta) &= \frac{1}{R}\sum_{r=1}^{R} \log p_\theta(W_r|X_r) \\ &= \frac{1}{R}\sum_{r=1}^{R} \log \frac{p_\theta(X_r|W_r)\cdot p(W_r)}{\displaystyle\sum_{W\in\mathcal{M}_r} p_\theta(X_r|W)\cdot p(W)}.\end{aligned}$$

The MMI criterion is given by the sum over the *a posteriori* probabilities of the spoken word sequences W_r on the training data, given the corresponding acoustic observations X_r. No smoothing function is needed for the MMI criterion, i.e. the smoothing function is given by the identity function.

3.2.3 Alternative Word Sequences

Ideally, in the case of the MMI criterion the set $\mathcal{M}_r$ would contain all possible word sequences. In practice, $\mathcal{M}_r$ is obtained through a recognition pass and is represented by N-best lists or word graphs. For MCE the spoken word sequence has to be excluded from this set. The contribution of each alternative sentence to reestimation is controlled by the exponent α. Very large values of α asymptotically lead to a maximum approximation. For the MMI criterion the latter is called *Corrective Training* (CT), where only the best recognized word sequences are used for discrimination. The smoothing function f leads to an optional weighting on the level of whole training utterances, as will be shown in the derivation of the reestimation equations for the case of Gaussian mixture densities in Chapter 4.

3.2.4 Limiting Cases

In the limit of infinite α, the MMI criterion becomes the *Corrective Training* (CT) approximation, and MCE becomes the *Falsifying Training* (FT) approximation. For CT, as well as FT, either the determination of the set of alternative word sequences, $\mathcal{M}_r$, or the definition of the weighting exponent $\alpha = \infty$ is redundant, since their choice for both CT and FT is mutually dependent. Therefore the choice of the weighting exponent is given in brackets in these cases. In both approximations, only one word sequence is included into the set of alternative word sequences. For the case of CT, this is the best recognized word sequence, with the consequence that only those training utterances contribute to CT, which are incorrectly recognized. For the case of FT, only the best recognized word sequence, which is *not* the recognized one, is used for training. Thus FT, like MCE, still represents a smoothed version of the error rate on the training data.

3.2.5 Interpretation and Relation to Frame Based Methods

All discriminative training criteria included in the unified approach represent sums over logarithmic, optionally smoothed, likelihood ratios. In other words, the objective of all discriminative training methods discussed here is, to optimize the likelihood of the spoken word sequence at the expense of some alternative or competing model, which here is defined by sums over alternative word sequences. If the sums over alternative word sequences were extended to include sums over any models, which are not restricted to represent word sequences, then even methods like frame discriminative training [Bahl$^+$ 1996, Povey & Woodland 1999] could be represented by the unified approach discussed here.

Under the assumption that the smoothing function f is increasing, the unified discriminative criterion is to be maximized according to the acoustic parameters θ. An optimization

of the unified criterion therefore tries to simultaneously maximize the emission probabilities of the spoken word sequence and minimize a weighted sum over the emission probabilities for each alternative word sequence given the acoustic observation sequence for each training utterance. The weights in the sum over the alternative word sequences are given by the language model probabilities relative to the spoken word sequence. Thus the unified discriminative criterion optimizes the class separability according to the words under consideration of the language model.

3.2.6 Relation to Other Discriminative Criteria

It should be noted that the *Minimum Squared Error* (MSE) criterion, which is often used for neural network training, is not contained in the unified formulation given here. The reason is that the MSE criterion involves a sum over all classes, i.e. (for speech recognition) all alternative word sequences of a training utterance, which could not be efficiently decomposed into a *forward-backward* like summation, as is the case for the MMI and MCE criterion. But such an efficient algorithm is necessary, in order to make discriminative training feasible for speech recognition, especially for large vocabulary applications.

3.3 Asymptotic Behaviour of Discriminative Criteria

In this section, proofs for the following properties for the MCE criterion and the *generalized Gini* criterion will be given [Schlüter & Ney 2000]:

- The MCE criterion represents an upper bound to the true (optimal) *Bayes*' error rate independent of the underlying model distribution.
- Model-free optimization of the MCE criterion with sufficient training data leads to a closed form solution. The corresponding model distribution leads to the true *Bayes*' error rate.
- A new discriminative training criterion similar to the MCE criterion, the *generalized Gini* (GG) criterion is defined. The GG criterion gives a closer upper bound to the *Bayes*' error rate than the MCE criterion. Model-free optimization of the GG criterion with infinite training data leads to the same optimal model distribution than the MCE criterion.

The model distributions resulting from model-free optimization of the MCE and the GG criterion differ considerably from the true distributions. This result suggests modifications to the structure of the model distributions applied in MCE training.
It should be noted that in this section we strictly distinguish between true distributions (representing the training data) and the corresponding model distributions. This strict distinction is usually not found in literature.

3.3.1 Error Bounds

The probability of error for a pattern classification problem is minimized by *Bayes'* decision rule [Duda & Hart 1973]. In case of continuous speech recognition, the probability

of error refers to the *sentence* error rate.

True distributions are denoted by p, e.g. $p(X, W)$ and $p(W|X)$. The true distributions are distinguished from the corresponding model distributions by the parameter θ, e.g. $p_\theta(X, W)$ and $p_\theta(W|X)$. For both the true distributions and the model distributions, the usual normalization constraints are assumed. Then the expectation of the true *Bayes* error rate $p_{Bayes}(e)$ is given by:

$$\begin{aligned} p_{Bayes}(e) &= \lim_{R\to\infty} \frac{1}{R} \cdot \sum_{r=1}^{R} \Big[1 - \max_W p(W|X_r)\Big] \\ &= \int dX \cdot p(X) \cdot \underbrace{\Big[1 - \max_W p(W|X)\Big]}_{:= p_{Bayes}(e|X)} \end{aligned} \tag{3.3}$$

with the local *Bayes* error $p_{Bayes}(e|X)$.

In the following sections it will be shown that, the MMI and the MCE criterion represent upper limits to the minimum sentence error rate.

3.3.1.1 Minimum Error Probability and MMI criterion

The model based posterior probability for word sequence W given the observation sequence X will be denoted by $p_\theta(W|X)$. In the limit of an infinite amount of training data the (model based) MMI criterion then is defined by:

$$\begin{aligned} \mathcal{F}_{MMI} &= \lim_{R\to\infty} \frac{1}{R} \cdot \sum_{r=1}^{R} \log p_\theta(W_r|X_r) \\ &= \int dX \cdot \sum_W p(X, W) \cdot \log p_\theta(W|X) \\ &= \int dX \cdot p(X) \cdot \underbrace{\sum_W p(W|X) \cdot \log p_\theta(W|X)}_{=: f_{MMI}(e|X)}, \end{aligned}$$

where $f_{MMI}(e|X)$ denotes the local MMI score. The notation here reflects the fact that $f_{MMI}(e|X)$ gives an upper bound to the local true *Bayes'* error rate. Utilizing the inequality $\log y \leq y - 1$ for the natural logarithm, we obtain the following inequality:

$$
\begin{aligned}
-f_{MMI}(e|X) &= -\sum_W p(W|X) \log p_\theta(W|X) \\
&\geq \sum_W p(W|X) \cdot (1 - p_\theta(W|X)) \\
&= 1 - \sum_W p(W|X) \cdot p_\theta(W|X) \\
&\geq 1 - \max_W p(W|X) \\
&= p_{Bayes}(e|X).
\end{aligned}
\tag{3.4}
$$

Hence, except for the negative sign, the MMI criterion is an upper bound to the true *Bayes* error rate independent of the model $p_\theta(W|X)$ used in the MMI criterion. However, since the above estimation of the natural logarithm is not very tight, the MMI criterion gives only a broad upper limit for the minimum error probability.

3.3.1.2 Minimum Error Probability and MCE criterion

The model distribution for word sequence W and the observation sequence X will be denoted by $p_\theta(W|X)$. In the limit of an infinite amount of training data, the MCE criterion [Juang & Katagiri 1992] then is given by[2]:

$$
\begin{aligned}
\mathcal{F}_{MCE} &= \lim_{R\to\infty} \frac{1}{R} \cdot \sum_{r=1}^{R} \frac{1}{1 + \left(\dfrac{p_\theta(W_r|X_r)}{\sqrt[\alpha]{\displaystyle\sum_{W \neq W_r} p_\theta^\alpha(W|X_r)}} \right)^{2\varrho\alpha}} \\
&= \sum_W \int dX \cdot p(X,W) \cdot \frac{1}{1 + \left(\dfrac{p_\theta(W|X)}{\sqrt[\alpha]{\displaystyle\sum_{W' \neq W} p_\theta^\alpha(W'|X)}} \right)^{2\varrho\alpha}} \\
&= \int dX \cdot p(X) \cdot \underbrace{\sum_W \frac{p(W|X)}{1 + \left(\dfrac{p_\theta(W|X)}{\sqrt[\alpha]{\displaystyle\sum_{W' \neq W} p_\theta^\alpha(W'|X)}} \right)^{2\varrho\alpha}}}_{=f_{MCE}(e|X)}
\end{aligned}
$$

where $f_{MCE}(e|X)$ denotes the local MCE error. To obtain a link between the local MCE criterion and the *Bayes* error, we need the following inequality:

[2]It should be noted that the MCE criterion here is defined to be positive as usual, whereas in all other parts of this work it is referred to with a negative sign due to the definition of the unified criterion, cf. Section 3.2.1.

$$\frac{\sum\limits_{W'\neq W} p_\theta^{2\varrho\alpha}(W'|X)}{\Big[\sum\limits_{W''\neq W} p_\theta^{\alpha}(W''|X)\Big]^{2\varrho}} = \sum_{W'\neq W}\Bigg[\frac{p_\theta^{\alpha}(W'|X)}{\sum\limits_{W''\neq W} p_\theta^{\alpha}(W''|X)}\Bigg]^{2\varrho} \leq 1 \qquad \text{for} \quad 2\varrho \geq 1. \tag{3.5}$$

By using the fact that, for $2\varrho = 1$, the equality is true, it is easy to prove the inequality. Using this inequality we obtain:

$$\begin{aligned}
f_{MCE}(e|X) &= \sum_W \frac{p(W|X)}{1+\dfrac{p_\theta^{2\varrho\alpha}(W|X)}{\Big[\sum\limits_{W'\neq W} p_\theta^{\alpha}(W'|X)\Big]^{2\varrho}}} \\
&\geq \sum_W \frac{p(W|X)}{1+\dfrac{p_\theta^{2\varrho\alpha}(W|X)}{\sum\limits_{W'\neq W} p_\theta^{2\varrho\alpha}(W'|X)}} \\
&= \sum_W p(W|X)\cdot \frac{\sum\limits_{W'\neq W} p_\theta^{2\varrho\alpha}(W'|X)}{\sum\limits_{W'} p_\theta^{2\varrho\alpha}(W'|X)} \\
&= \sum_W p(W|X)\cdot\Bigg[1-\frac{p_\theta^{2\varrho\alpha}(W|X)}{\sum\limits_{W'} p_\theta^{2\varrho\alpha}(W'|X)}\Bigg] \\
&= \underbrace{1-\sum_W p(W|X)\cdot\frac{p_\theta^{2\varrho\alpha}(W|X)}{\sum\limits_{W'} p_\theta^{2\varrho\alpha}(W'|X)}}_{:=\, f_{GG}(e|X)} \\
&\geq 1-\max_V p(V|X)\cdot\sum_W \frac{p_\theta^{2\varrho\alpha}(W|X)}{\sum\limits_{W'} p_\theta^{2\varrho\alpha}(W'|X)} \\
&= 1-\max_W p(W|X) \\
&= p_{Bayes}(e|X),
\end{aligned} \tag{3.6}$$

where the "intermediate" local error $f_{GG}(e|X)$ will be used to define a new discriminative criterion, which will be discussed in Section 3.3.2.2. The above result shows that the MCE criterion gives an upper bound to the true *Bayes* error rate independent of the discrimination function model used in the MCE criterion.
In the limit of infinite α, the MCE criterion approaches the true error rate for the probability model chosen. Therefore, if the probability models were exact, the upper limit would approach the equality in this limit. Nevertheless, the upper limit derived here is valid for *any* probability model $p_\theta(W|X)$, and *any* choice of model parameters θ.

3.3.2 Model-Free Optimization

3.3.2.1 ML and MMI Criterion

For the ML and the MMI criterion, model-free optimization in the asymptotic case of infinite training data leads to closed form solutions. Under consideration of the normalization constraints for probability models, the model-free optimization of the ML and the MMI criterion leads to the true distributions:

$$\begin{aligned} p_\theta(X|W) &= \frac{p(X,W)}{p(W)} = p(X|W) \qquad \text{(ML criterion)}, \\ p_\theta(W|X) &= \frac{p(X,W)}{p(X)} = p(W|X) \qquad \text{(MMI criterion)}. \end{aligned}$$

Therefore, under the assumption that the true *a-priori* probability $p(W)$ is known, both the ML and the MMI criterion lead to *Bayes'* decision rule for the optimization of the sentence error rate.

3.3.2.2 MCE and Related Criteria

For the asymptotic case of infinite training data, in this section we will derive a closed-form solution for the model-free optimization of both the original MCE criterion as well as for the *generalized Gini* (GG) criterion. The GG criterion is derived from the intermediate local error $f_{GG}(e|X)$ and gives even a closer upper bound to the true *Bayes'* error than the MCE criterion itself, cf. Eq. (3.6). Because of its similarity to the *Gini* criterion [Breiman 1984, pp. 38], the new criterion will be called the *generalized Gini* (GG) criterion:

$$\mathcal{F}_{GG} = \int dX \cdot p(X) \cdot \sum_W p(W|X) \cdot \big[1 - \frac{p_\theta^{2\varrho\alpha}(W|X)}{\sum_{W'} p_\theta^{2\varrho\alpha}(W'|X)}\big].$$

In general, the exponent $2\varrho\alpha$ could be replaced by a single exponent. However, here the above notation is kept for reasons of comparison between MCE and GG.

In Ineq. (3.6), a system of inequalities between the local MCE error $f_{MCE}(e|X)$, the local GG error $f_{GG}(e|X)$, and the true local *Bayes*' error rate $p_{Bayes}(e|X)$ is given. The equality, i.e. optimality for both the MCE and the GG criterion is obtained if the model distribution $p_\theta(W|X)$ is set to:

$$\hat{p}_\theta(W|X) = \begin{cases} 1 & \text{iff} \quad W = \underset{W'}{\operatorname{argmax}}\, p(W'|x) \\ 0 & \text{otherwise.} \end{cases}$$

When we replace the true distribution $p(W|X)$ in *Bayes'* decision rule by this model distribution $\hat{p}_\theta(W|X)$, we obtain the important result that the decision about the unknown class remains the same:

$$\underset{W}{\operatorname{argmax}}\, \hat{p}_\theta(W|X) = \underset{W}{\operatorname{argmax}}\, p(W|X).$$

Therefore both the MCE criterion and the GG criterion lead to optimal (sentence) error rate in the asymptotic case of infinite training data, although the structure of the corresponding model distributions differs considerably from the structure of the true distributions.

3.3.3 Discussion of Asymptotic Behaviour

Despite the fact that the MCE criterion only *approximates* the empirical error rate on the training data, in Section 3.3.1.2 it is shown that the MCE criterion gives an upper bound to the true *Bayes'* error rate. This result is independent of the discrimination function model used in the MCE criterion.

An optimization of the MCE criterion aims at improving the empirical error rate. Moreover, in the limit of $\varrho\alpha \to \infty$ the MCE criterion *equals* the empirical error rate on the training data, and any discrimination function minimizing the empirical error rate gives a possible solution in this case. Nevertheless, for finite $\varrho\alpha$, the *optimal* outcome of an MCE training might very well depend on the model choice. In this paper, for the case of infinite training data, a model-free optimization is performed for the MCE criterion, which leads to a closed form solution, which leads to the true *Bayes'* error rate. The same result holds for the GG criterion defined in this section, which is shown to be similar to the MCE criterion, and which even gives a closer upper bound to the true *Bayes'* error rate, than the MCE criterion itself.

Although being a function of the true distributions, the model distribution resulting from the model-free optimization of both the MCE and the GG criterion differs considerably from the corresponding true distribution and from the structure of the model distributions usually used for statistical pattern recognition. This result for the MCE criterion suggests that the structure of the models usually used for ML and MMI training might not be optimal for MCE training. Since these analytical results have only been found at the end of this work, no experimental investigations on this have been performed so far.

3.4 Comparative Results for Discriminative Criteria

In Table 3.2 results for several discriminative training criteria, MMI, CT, MCE and FT in comparison to ML training are presented for the recognition of telephone line recorded German continuous digit strings on the *SieTill* corpus.

Table 3.2: Comparison of recognition results for several discriminative training criteria on the *SieTill* corpus for telephone line recorded, continuously spoken German digits. Results are given for low (single densities) and optimal model complexity. The number of densities per mixture is denoted by 'dns'.

dns	criterion	del - ins - sub	WER[%]	SER[%]
1	ML	304-270-1053	3.78	9.74
	CT	329-201- 695	2.85	7.27
	MMI	349-175- 684	2.81	7.13
	FT	281-274- 650	2.80	7.27
	MCE	315-176- 657	2.60	6.73
32	ML	198-201- 450	1.97	5.31
64		198-162- 419	1.81	4.93
32	CT	223-128- 433	1.82	4.97
	MMI	182-161- 407	1.74	4.80
	FT	176-158- 409	1.67	4.50
64	MCE	183-148- 398	1.69	4.64

3.4.1 Criterion Performance Depending on Model Complexities

A first observation is that the results for the comparison between discriminative criteria differ qualitatively for low and optimal complexity of the underlying acoustic models. For single densities, the word error rate for ML training amounts to 3.78%. Using MCE training, this result was reduced to 2.60%. This is a relative reduction of nearly 1/3. On the other hand, the word error rate using the optimal model complexity, i.e. using 64 Gaussian densities per mixture amounted to 1.81% for ML training. The best result for optimal model complexity was obtained using FT or MCE training giving word error rates of 1.67% and 1.69% respectively – a relative reduction of about 7%, i.e., the relative reduction in word error rate by discriminative training in comparison to ML training is considerably less for high acoustic model complexity than for low model complexity.

In a similar way, the difference between the MMI and the MCE criterion diminishes for increasing model complexity: for single Gaussian densities MCE training gives a relative reduction of 7.5% in word error rate compared to MMI training. For optimal model complexity this advantage for MCE training decreases to a relative reduction in word error rate of 3% only.

All in all, discriminative training consistently gave better results than ML training, especially for low model complexities. Independent of the model complexity in question, MCE training consistently gave better results than MMI training, which might be due to the fact that the MCE criterion by definition works "closer to the error rate", than the MMI criterion.

It should be noted that, the same approach to MMI training, as it is presented here,

has been successfully extended to statistical image object recognition using Gaussian mixtures [Dahmen$^+$ 1999].

3.4.2 Approximated vs. Complete Criteria

Different to the comparison of MMI vs. MCE training, the comparison between the approximations to the MMI and the MCE criterion, CT and FT respectively leads to some further understanding of the effect of discriminative training. Although in different ways, CT and FT are somewhat concentrated versions of MMI and MCE training. In contrast to MMI, CT only considers those training utterances for training, which are incorrectly recognized. Similarly, FT for each training utterance only considers that alternative incorrect word sequence, which is next to the decision boundary.

For low model complexity, MMI and CT give approximately the same results, whereas MCE performs significantly better than FT. On the other hand, for optimal model complexities, MMI performs slightly better than CT, and both FT and MCE better than MMI. One reason for the good performance of MCE training for low model complexity is that outliers are ignored, since these do not have good chances to be modeled by coarse models. In contrast to this, MMI and more so CT do try to correct outliers. Furthermore, by using more than a single alternative word sequence, MCE – in contrast to FT – introduces a smoothing, which facilitates the process of finding and optimizing most of those parts of a coarse model, which are possible to lead to improvements.

For more detailed models, the situation is different: FT performs as good as MCE, since concentration on particular model parts is possible for detailed models. For detailed models MMI is still subject to outliers and even more CT suffers from over-training. In the extreme case of no recognition errors on the training corpus, CT would not result in any effect, although the distance between correct and incorrect models might still be small.

Chapter 4

The Parameter Optimization Problem

Since there does not exist any discriminative training method guaranteed to converge under practical conditions, much effort has been made to develop parameter optimization techniques with fast and reliable convergence. The commonly used parameter optimization techniques for discriminative training are the *extended Baum* (EB) algorithm, and the gradient descent (GD) method. EB is an extension to the standard *Baum-Welch* algorithm designed for optimization of the MMI criterion. EB was first developed for discriminative training of discrete probabilities [Cardin$^+$ 1993, Gopalakrishnan$^+$ 1991, Normandin & Morgera 1991, Normandin$^+$ 1994a], but was later extended to continuous densities [Normandin 1991, Normandin 1996]. On the other hand, optimization of the MCE criterion is usually performed in combination with GD. In this section, it will be shown that EB and GD in fact are very similar and give similar recognition results in the case of the MMI criterion. Furthermore, using the unifying approach for discriminative training criteria presented here, the EB algorithm will be extended to the optimization of the MCE criterion.

4.1 Discriminative Averages

4.1.1 Continuous Mixture Densities

In the experiments presented here, continuous mixture density HMMs are applied for acoustic modeling. The probability density for a state s is defined by

$$p_\theta(x_{rt}|s) = \sum_l c_{sl} \cdot p(x_{rt}|\mu_{sl}, \Sigma_{sl}),$$

where in the following state indices are identified with their corresponding mixture indices. Each index l represents a Gaussian mixture probability density $p(x_{rt}|\mu_{sl}, \Sigma_{sl})$ with parameters $\theta_{sl} = \{c_{sl}, \mu_{sl}, \Sigma_{sl}\}$, i.e. the mixture weights c_{sl}, the mean vectors μ_{sl} and the pooled, state or density specific covariance matrices Σ_{sl}.

4.1.2 Conventional Forward-Backward Probability

In addition, the *forward-backward* (FB) probability $\gamma_{rt}(s;W)$ for being in mixture s at time t is defined, given a word sequence W and the acoustic observation sequence X_r of a training utterance r. In *Viterbi* approximation [Ney 1990] the FB probability equals one for the states of the best alignment path $s_t(X_r, W)$ and zero otherwise,

$$\begin{aligned}
\gamma_{rt}(s;W) &= p_\theta(s_t = s|X_r, W) && (4.1)\\
&= \frac{p_\theta(s_t = s, X_r|W)}{p_\theta(X_r|W)} && (4.2)\\
&\overset{\text{Viterbi}}{\approx} \delta(s, s_t(X_r, W)), && (4.3)
\end{aligned}$$

with the *Kronecker* delta function δ.

4.1.3 Generalized Forward-Backward Probability

Similarly, the generalized FB probability $\gamma_{rt}(s)$ for being in mixture s at time t is defined, given the acoustic observation sequence X_r of a training utterance r, accumulated over the set of alternative word sequences,

$$\begin{aligned}
\gamma_{rt}(s) &= \sum_{W\in\mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum_{V\in\mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)} \cdot \gamma_{rt}(s;W) && (4.4)\\
&\overset{\text{Viterbi}}{\approx} \sum_{W\in\mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum_{V\in\mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)} \cdot \delta(s, s_t(X_r, W)).
\end{aligned}$$

4.1.4 Formal Differentiation of the Unified Criterion

Formal differentiation of the unified discriminative criterion with respect to density specific parameters θ_{sl} of the acoustic emission probabilities leads to the following expression (for further details on the derivation see Appendix C.2),

$$\begin{aligned}
\frac{\partial\mathcal{F}}{\partial\theta_{sl}} =& \frac{1}{R}\sum_{r=1}^{R} \alpha\, f'\Big(\log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\Big)\\
&\cdot \sum_{t=1}^{T_r} \left[\gamma_{tr}(s;W_r) - \gamma_{tr}(s)\right] \frac{c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl})}{\sum_k c_{sk}p(x_{rt}|\mu_{sk},\Sigma_{sk})} \qquad (4.5)\\
&\cdot \frac{\partial}{\partial\theta_{sl}} \log c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl}).
\end{aligned}$$

4.1.5 Definition of Discriminative Averages

Since expressions like Eq. (4.5) occur frequently in the following, discriminative averages for functions $g(x)$ of the acoustic observations x are defined:

$$\Gamma_{sl}\left(g(x)\right) = \frac{1}{R}\sum_{r=1}^{R} f'\Big(\log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\Big) \cdot \sum_{t=1}^{T_r}\left[\gamma_{rt}(s;W_r) - \gamma_{rt}(s)\right] \cdot \frac{c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl})}{\sum\limits_k c_{sk}p(x_{rt}|\mu_{sk},\Sigma_{sk})} \cdot g(x_{rt}).$$

with usually polynomial functions g of the acoustic features. Using this definition, the formal differentiation of the unified discriminative criterion could be reduced to

$$\frac{\partial\mathcal{F}}{\partial\theta_{sl}} = \alpha\cdot\Gamma_{sl}\left(\frac{\partial}{\partial\theta_{sl}}\log c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl})\right). \tag{4.6}$$

A similar expression holds for the case of state specific parameters θ_s. In order to simplify later expressions, state specific discriminative averages are defined also,

$$\Gamma_s\left(g(x)\right) = \sum_l \Gamma_{sl}\left(g(x)\right).$$

4.1.5.1 Further Decomposition of Discriminative Averages

For the derivation of smoothed reestimation equations for the mixture weights and for the derivation of discriminative training of linear transforms the following decomposition of the discriminative averages into the following averages, on the spoken word sequences (spk), and the sets of alternative word sequences represented by the generalized FB probability (gen) respectively, will be defined:

$$\Gamma_{sl}^{spk}\left(g(x)\right) = \frac{1}{R}\sum_{r=1}^{R} f'\Big(\log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\Big) \cdot \sum_{t=1}^{T_r}\gamma_{rt}(s;W_r) \cdot \frac{c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl})}{\sum\limits_k c_{sk}p(x_{rt}|\mu_{sk},\Sigma_{sk})} \cdot g(x_{rt}),$$

$$\Gamma_{sl}^{gen}\left(g(x)\right) = \frac{1}{R}\sum_{r=1}^{R} f'\Big(\log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\Big) \cdot \sum_{t=1}^{T_r}\gamma_{rt}(s) \cdot \frac{c_{sl}p(x_{rt}|\mu_{sl},\Sigma_{sl})}{\sum\limits_k c_{sk}p(x_{rt}|\mu_{sk},\Sigma_{sk})} \cdot g(x_{rt}).$$

with:

$$\Gamma_{sl}(g(x)) \;=\; \Gamma_{sl}^{spk}(g(x)) - \Gamma_{sl}^{gen}(g(x)). \tag{4.7}$$

Summing the above equations over all densities l results in the corresponding state specific expressions:

$$\begin{aligned}\Gamma_{s}^{spk}(g(x)) &= \sum_{l} \Gamma_{sl}^{spk}(g(x)),\\ \Gamma_{s}^{gen}(g(x)) &= \sum_{l} \Gamma_{sl}^{gen}(g(x)),\end{aligned}$$

and

$$\begin{aligned}\Gamma_{s}(g(x)) &= \sum_{l} \Gamma_{sl}(g(x))\\ &= \Gamma_{s}^{spk}(g(x)) - \Gamma_{s}^{gen}(g(x)).\end{aligned}$$

4.1.6 Discriminative Averages in Viterbi Approximation

Like the *Viterbi* approximation for the case of state time-alignment, the maximum approximation is also applied to the calculation of mixture densities, where the best density index, given observation x and state s, shall be denoted by $l(x,s)$. Using the *Viterbi* approximation and maximum approximation at the mixture level, the discriminative averages could be simplified to:

$$\begin{aligned}\Gamma_{sl}\left(g(x)\right) \overset{\text{Viterbi}}{\approx} \frac{1}{R}\sum_{r=1}^{R} f'\big(\log \frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\big)\cdot\\ \cdot\sum_{t=1}^{T_r}\left[\delta(s,s_t(X_r,W_r)) - \gamma_{rt}(s)\right]\cdot\delta(l,l(x_{rt},s))\cdot g(x_{rt}).\end{aligned}$$

4.1.6.1 Decomposed Discriminative Averages in Viterbi Approximation

Similarly, the decomposition of the discriminative averages in *Viterbi* approximation and maximum approximation at the mixture level simplifies to the following expressions:

$$\Gamma_{sl}^{spk}\left(g(x)\right) \overset{\text{Viterbi}}{\approx} \frac{1}{R}\sum_{r=1}^{R} f'\big(\log\frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\big)\cdot \cdot\sum_{t=1}^{T_r}\delta(s,s_t(X_r,W_r))\cdot\delta(l,l(x_{rt},s))\cdot g(x_{rt}), \tag{4.8}$$

$$\Gamma_{sl}^{gen}\left(g(x)\right) \overset{\text{Viterbi}}{\approx} \frac{1}{R}\sum_{r=1}^{R} f'\big(\log\frac{p_\theta^\alpha(X_r|W_r)p^\alpha(W_r)}{\sum\limits_{W\in\mathcal{M}_r} p_\theta^\alpha(X_r|W)p^\alpha(W)}\big)\cdot \cdot\sum_{t=1}^{T_r}\gamma_{rt}(s)\cdot\delta(l,l(x_{rt},s))\cdot g(x_{rt}). \tag{4.9}$$

4.2 Extended *Baum* Algorithm

4.2.1 Optimization of the MMI Criterion

Discriminative training with the MMI criterion usually applies an extended version of *Baum-Welch* training, the EB algorithm [Gopalakrishnan$^+$ 1991, Normandin 1991, Normandin$^+$ 1994b, Normandin 1996]. For the case of continuous emission probabilities, the MMI criterion is maximized via the following auxiliary function (derived from [Normandin 1991], p.100):

$$\begin{aligned}\mathcal{S}(\theta,\hat\theta) = &\sum_s \frac{1}{R}\sum_{r=1}^{R} f'\Big(\frac{p^\alpha(X_r,W_r)}{\sum\limits_{W\in\mathcal{M}_r} p^\alpha(X_r,W)}\Big)\cdot\sum_{t=1}^{T_r}\left[\gamma_{rt}(s;W_r)-\gamma_{rt}(s)\right]\cdot\log p(x_{rt}|\hat\theta_s)\\ &+\sum_s D_s\int dx\, p(x|\theta_s)\cdot\log p(x|\hat\theta_s),\end{aligned} \tag{4.10}$$

which is to be optimized iteratively.

4.2.2 Extension to the Unified Criterion

It should be noted that the auxiliary function in Eq. (4.10) originally was derived for the MMI criterion, for which convergence has been proven in case of discrete probability models [Gopalakrishnan$^+$ 1991]. Later, the approach has even been generalized to cover objective functions, which are not necessarily rational with respect to the probability models [Kanevsky 1995], like the unified criterion presented here. Nevertheless, here discriminative training will be discussed with respect to continuous probability models. For the MMI criterion, it has been shown how to extend the EB algorithm to the continuous case [Normandin 1991], which could equally well be done for the generalization presented in [Kanevsky 1995]. However, the corresponding iteration constants D_s needed

to guarantee convergence are infinite in the continuous case [Normandin 1991], which means that convergence could not be guaranteed under realistic conditions. However, this extension has been performed in this work, in order to transfer the method for choosing step sizes from the EB algorithm to gradient descent.

One motivation for this was the fact that EB has been reported to perform better than GD [Kapadia$^+$ 1993]. Another motivation was to provide a common optimization framework which could equally well be applied to all criteria included in the unified approach. It should be noted that proportionalities found for the step sizes for GD optimization by comparison to the EB algorithm are in agreement with the results from independent theoretical considerations for GD step sizes for MCE training presented in [Chou$^+$ 1992].

4.2.3 Derivation of Reestimation Formulae

Applying the maximum approximation to the mixture density calculation, the differentiation of the above auxiliary function with respect to the new iterated parameters $\hat{\theta}_s$ leads to the following expression, from which reestimation formulae can be derived:

$$\begin{aligned}\frac{\partial \mathcal{S}(\theta,\hat{\theta})}{\partial \hat{\theta}_{sl}} = \quad & \Gamma_{sl}\left(\frac{\partial}{\partial \theta_{sl}}\log \hat{c}_{sl}p(x_{rt}|\hat{\theta}_{sl})\right) \\ & + D_s c_{sl}\int dx\, p(x|\theta_{sl})\frac{\partial \log \hat{c}_{sl}p(x|\hat{\theta}_{sl})}{\partial \hat{\theta}_{sl}}.\end{aligned}$$

Analogous to the case of reestimating discrete probabilities [Normandin 1991], here the integral term enables convergence, by smoothing the discriminative averages with the corresponding parameters of the previous iteration. The constants D_s control the convergence rate.

4.2.4 Reestimation Formulae for Gaussian Mixtures

For reestimation of Gaussian mixture densities with state specific diagonal covariance, the parameter θ_{sl} represents the initial parameter set of a density l of state s consisting of the mixture weight c_{sl}, the Gaussian mean vector μ_{sl} and pooled variance σ^2, or state specific variance σ_s^2, or density specific variance σ_{sl}^2. Using the EB algorithm, the following reestimation equations for the new parameters are obtained:

$$\hat{\mu}_{sl,\,\mathrm{EB}} = \frac{\Gamma_{sl}(x) + D_s c_{sl} \mu_{sl}}{\Gamma_{sl}(1) + D_s c_{sl}} \tag{4.11}$$

$$\hat{\sigma}^2_{\mathrm{EB}} = \frac{\Gamma(x^2) + \sum_s D(\sigma^2 + \sum_l c_{sl}\mu^2_{sl})}{\Gamma(1) + \sum_s D} - \sum_{s,l} \frac{\Gamma_{sl}(1) + D c_{sl}}{\Gamma(1) + \sum_s D} \hat{\mu}^2_{sl} \tag{4.12a}$$

$$\hat{\sigma}^2_{s,\,\mathrm{EB}} = \frac{\Gamma_s(x^2) + D_s(\sigma^2_s + \sum_l c_{sl}\mu^2_{sl})}{\Gamma_s(1) + D_s} - \sum_{l} \frac{\Gamma_{sl}(1) + D_s c_{sl}}{\Gamma_s(1) + D_s} \hat{\mu}^2_{sl} \tag{4.12b}$$

$$\hat{\sigma}^2_{sl,\,\mathrm{EB}} = \frac{\Gamma_{sl}(x^2) + D_s c_{sl}(\sigma^2_{sl} + \mu^2_{sl})}{\Gamma_{sl}(1) + D_s c_{sl}} - \hat{\mu}^2_{sl} \tag{4.12c}$$

$$\hat{c}_{sl,\,\mathrm{EB}} = \frac{\partial \mathcal{F}(\theta)/\partial c_{sl} + D_s}{\sum_k c_{sk} \partial \mathcal{F}(\theta)/\partial c_{sk} + D_s} c_{sl}, \tag{4.13}$$

where in the pooled case always $D_s \equiv D$ is assumed, i.e., the iteration constant is pooled, too. In consequence, this pooling scheme applies to the reestimation equations of the means as well.

4.2.4.1 Smoothed Reestimation of Mixture Weights

It should be noted that the reestimation Eq. (4.13) for the mixture weights is not used with the exact derivatives of the criterion F, but with smoothed versions as proposed in [Normandin 1996]:

$$\partial \mathcal{F}(\theta)/\partial c_{sl} \approx \frac{\Gamma^{spk}_{sl}(1)}{\Gamma^{spk}_{s}(1)} - \frac{\Gamma^{gen}_{sl}(1)}{\Gamma^{gen}_{s}(1)}.$$

This leads to the following smoothed reestimation equation:

$$\hat{c}_{sl,\,\mathrm{EB}} = \frac{\dfrac{\Gamma_{sl}^{spk}(1)}{\Gamma_{s}^{spk}(1)} - \dfrac{\Gamma_{sl}^{gen}(1)}{\Gamma_{s}^{gen}(1)} + C_s}{\displaystyle\sum_{l'} c_{sl'} \left[\frac{\Gamma_{sl'}^{spk}(1)}{\Gamma_{s}^{spk}(1)} - \frac{\Gamma_{sl'}^{gen}(1)}{\Gamma_{s}^{gen}(1)} \right] + C_s} c_{sl},$$

which also requires new iteration constants C_s, since the magnitude of the smoothed terms differs from the corresponding terms for the means and variances.

4.3 Gradient Descent

4.3.1 Derivation of Reestimation Formulae

Performing gradient descent for parameter optimization, the following iterative reestimation equation is applied for parameters $\theta_{s(l)}$,

$$\hat{\theta}_{s(l)} = \theta_{s(l)} + \Delta\theta_{s(l)} \cdot \frac{\partial \mathcal{F}(\theta)}{\partial \theta_{s(l)}}.$$

For the case of gradient descent, the following reestimation equations are obtained:

$$\hat{\mu}_{sl,\,\mathrm{GD}} = \mu_{sl} + \Delta\mu_{sl} \frac{\partial \mathcal{F}(\theta)}{\partial \mu_{sl}} \tag{4.14}$$

$$\hat{\sigma}^2_{\mathrm{GD}} = \sigma^2 + \Delta\sigma^2 \frac{\partial \mathcal{F}(\theta)}{\partial \sigma^2} = \sigma^2 - \Delta\sigma^{-2} \frac{\partial \mathcal{F}(\theta)}{\partial \sigma^{-2}} \tag{4.15a}$$

$$\hat{\sigma}^2_{s,\,\mathrm{GD}} = \sigma^2_{s} + \Delta\sigma^2_{s} \frac{\partial \mathcal{F}(\theta)}{\partial \sigma^2_{s}} = \sigma^2_{s} - \Delta\sigma_{s}^{-2} \frac{\partial \mathcal{F}(\theta)}{\partial \sigma_{s}^{-2}} \tag{4.15b}$$

$$\hat{\sigma}^2_{sl,\,\mathrm{GD}} = \sigma^2_{sl} + \Delta\sigma^2_{sl} \frac{\partial \mathcal{F}(\theta)}{\partial \sigma^2_{sl}} = \sigma^2_{sl} - \Delta\sigma_{sl}^{-2} \frac{\partial \mathcal{F}(\theta)}{\partial \sigma_{sl}^{-2}} \tag{4.15c}$$

$$\hat{c}_{sl,\,\mathrm{GD}} = \frac{c_{sl} + \Delta c_{sl} \dfrac{\partial \mathcal{F}(\theta)}{\partial c_{sl}}}{1 + \displaystyle\sum_{k} \Delta c_{sk} \frac{\partial \mathcal{F}(\theta)}{\partial c_{sk}}}. \tag{4.16}$$

As for the case of the EB algorithm, the derivatives for reestimation of the mixture weights are replaced by smoothed versions according to [Normandin 1996].

4.4 Comparison of Optimization Methods

Here, the comparison of optimization methods will be discussed for the example of Gaussian mixture densities with pooled, state specific, and density specific diagonal variance. Similar formulae hold for full covariances.

4.4.1 Interdependence between EB and GD

Comparing EB and GD optimization, the following special step sizes could be obtained for GD, which lead to close resemblance between the EB and GD reestimation equations [Schlüter$^+$ 1997a, Schlüter$^+$ 1997b, Schlüter$^+$ 2000]:

$$\Delta\mu_{sl} = \frac{\sigma^2}{\Gamma_{sl}(1) + Dc_{sl}} \tag{4.17a}$$

$$\Delta\mu_{sl} = \frac{\sigma_s^2}{\Gamma_{sl}(1) + D_s c_{sl}} \tag{4.17b}$$

$$\Delta\mu_{sl} = \frac{\sigma_{sl}^2}{\Gamma_{sl}(1) + D_s c_{sl}} \tag{4.17c}$$

$$\Delta\sigma^{-2} = \frac{2}{\Gamma(1) + \sum_s D} \tag{4.18a}$$

$$\Delta\sigma_s^{-2} = \frac{2}{\Gamma_s(1) + D_s} \tag{4.18b}$$

$$\Delta\sigma_{sl}^{-2} = \frac{2}{\Gamma_{sl}(1) + D_s c_{sl}} \tag{4.18c}$$

$$\Delta c_{sl} = \frac{\partial\mathcal{F}(\theta)/\partial c_{sl} + D_s - 1}{\partial\mathcal{F}(\theta)/\partial c_{sl}}\, c_{sl}. \tag{4.19}$$

Using the above step sizes for gradient descent, the following relations between the reestimated parameters of GD and EB are obtained, provided the initial parameters are equal:

$$\hat{\mu}_{sl,\,\mathrm{GD}} = \hat{\mu}_{sl,\,\mathrm{EB}} \tag{4.20}$$

$$\hat{\sigma}^2_{\mathrm{GD}} = \hat{\sigma}^2_{\mathrm{EB}} + \sum_s \sum_l \frac{\Gamma_{sl}(1) + Dc_{sl}}{\Gamma(1) + \sum_s D} \left(\mu_{sl} - \hat{\mu}_{sl,\,\mathrm{EB}}\right)^2 \tag{4.21a}$$

$$\hat{\sigma}^2_{s,\,\mathrm{GD}} = \hat{\sigma}^2_{s,\,\mathrm{EB}} + \sum_l \frac{\Gamma_{sl}(1) + D_s c_{sl}}{\Gamma_s(1) + D_s} \left(\mu_{sl} - \hat{\mu}_{sl,\,\mathrm{EB}}\right)^2 \tag{4.21b}$$

$$\hat{\sigma}^2_{sl,\,\mathrm{GD}} = \hat{\sigma}^2_{sl,\,\mathrm{EB}} + \left(\mu_{sl} - \hat{\mu}_{sl,\,\mathrm{EB}}\right)^2 \tag{4.21c}$$

$$\hat{c}_{sl,\,\mathrm{GD}} = \hat{c}_{sl,\,\mathrm{EB}}. \tag{4.22}$$

Clearly, the means and mixture weights are equally reestimated by GD and EB, whereas the variances differ only by the squared step sizes of the corresponding means, which in case of pooled and state specific variances are averaged. Since the only dependence on the particular criterion applied is contained in the discriminative averages, the above resemblance of GD and EB holds for all criteria contained in the unifying approach for discriminative criteria discussed here. Hence, the methods of optimal choice of iteration constants in the case of EB optimization of the MMI criterion could equally well be applied to the MCE criterion.

Looking at the reestimation formula for the mean vectors, the step sizes for GD are found to be proportional to the corresponding variance. This result is inherited in the EB algorithm. By the above comparison it is transfered to GD without additional assumptions. The variance factor in the step sizes for GD reestimation of Gaussian mean vectors (cf. Eq. (4.20)) was introduced independently in [Chou$^+$ 1992] by theoretical arguments.

4.4.2 Comparative Results for Parameter Optimization

In Tables 4.1-4.4, results for discriminative training comparing extended *Baum* (EB) and gradient descent (GD) optimization are given.

Table 4.1: Recognition results for the *TI digit string* corpus. Word (WER) and sentence error rates (SER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with state specific diagonal covariances.

corpus	criterion	opt.	del/ins[%]	WER[%]	SER[%]
train	ML	–	0.28/0.04	0.56	1.69
	CT	EB	0.00/0.00	0.00	0.00
		GD	0.01/0.01	0.02	0.06
test	ML	–	0.20/0.11	0.72	2.00
	CT	EB	0.12/0.08	0.50	1.38
		GD	0.13/0.08	0.47	1.32

Table 4.2: Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with state specific diagonal covariances.

corpus	criterion	opt.	del/ins[%]	WER[%]
train	ML	–	0.28/1.24	4.11
	CT	EB	0.14/0.03	0.50
		GD	0.14/0.07	0.59
test	ML	–	0.48/1.59	5.00
	CT	EB	0.86/0.37	3.07
		GD	0.72/0.33	2.96

Table 4.4: Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Gaussian mixture densities with 4 densities per state, one pooled diagonal covariance and LDA.

corpus	criterion	opt.	del/ins[%]	WER[%]
train	ML	–	0.22/0.67	2.29
	CT	EB	0.21/0.16	0.67
		GD	0.21/0.15	0.69
test	ML	–	0.35/1.04	3.05
	CT	EB	0.62/0.52	2.59
		GD	0.63/0.48	2.53

As was expected analytically, no consistent differences between results using GD and EB reestimation could be observed for both the *TI digit string* and the *SieTill* corpus

Table 4.3: Recognition results for the *SieTill* corpus. Word error rates (WER) for Maximum Likelihood (ML) and Corrective Training (CT) using extended *Baum* (EB) and gradient descent (GD) optimization. Single Gaussian densities with one pooled diagonal covariance.

corpus	criterion	opt.	del/ins[%]	WER[%]
train	ML	–	0.44/0.86	4.59
	CT	EB	0.21/0.21	1.08
		GD	0.21/0.24	1.19
test	ML	–	0.59/1.26	5.79
	CT	EB	0.59/0.62	3.32
		GD	0.57/0.63	3.45

employing several kinds of acoustic modeling. Note that these comparative results on the *SieTill* corpus are not the best reported in this work, since they were produced while still optimizing the baseline recognition system, i.e. especially feature extraction and word modeling.

As a consequence of the comparison of EB and GD, GD optimization was arbitrarily chosen for all following experiments. For MCE training, also GD was applied for parameter optimization, and the formalism for finding optimal step sizes was used, as obtained from the comparison of GD and EB in the case of the CT and MMI criteria.

4.5 Convergence Control

Proofs of convergence do exist for both GD [Chou$^+$ 1992] and (in the case of discrete probabilities) for EB [Baum & Eagon 1967, Gopalakrishnan$^+$ 1991]. In the case of EB reestimation of continuous emission probabilities, convergence even has only been proven for infinitesimal step sizes [Normandin 1991]. In practice, reasonable fast convergence is achieved in the EB case, if the iteration constants D_s are chosen such that the corresponding variances are kept positive [Normandin 1996]. In addition, all denominators in the reestimation equations are ensured to remain non-singular.

4.5.1 Derivation of Iteration Constants

4.5.1.1 Iteration Constants for the Means and Variances

For density specific variances, the condition of positive variances leads to inequations which are quadratic in the iteration constants and could be solved explicitly to give the lowest iteration constant ensuring positive variance [Valtchev$^+$ 1997]. On the other hand, it is not possible to find an explicit formula for the lowest iteration constant ensuring the condition of positive variances for the cases of pooled or state specific variances. This is due to the second term in Eqs. (4.12a) and (4.12b), which prevents an explicit solution,

since, through $\hat{\mu}_{sl}^2$, D_s occurs in the denominator within the summation over the densities. In order to find the smallest iteration constants ensuring positive variances in the cases of state specific or pooled variances, the following inequalities are required,

$$\sigma_{s,\,\mathrm{EB}}^2 \;\geq\; \sigma_{\min}, \tag{4.23}$$

$$\Gamma_{sl}(1) + c_{sl} D_s \;\geq\; \frac{1}{\beta_s}, \tag{4.24}$$

with positive constants $\sigma_{\min} > 0$. The value of $\sigma_{\min}$ provides a lower limit for the variances and therefore depends on the magnitude of the acoustic features. A value of 1 has been found to be appropriate, which has been approximately 10^4 times lower than the usual magnitude of the variances observed in the experiments reported in this work. The value of the lower limit to the denominators, β_s, is determined according to the magnitude of the counts $\Gamma_{sl}^{spk}(1)$ and $\Gamma_{sl}^{gen}(1)$ and the corresponding difference $\Gamma_{sl}(1) = \Gamma_{sl}^{spk}(1) - \Gamma_{sl}^{gen}(1)$. In preliminary experiments, the following heuristic formula for the calculation of β_s were developed, in order to obtain optimal training convergence:

$$\frac{1}{\beta_s} = 1 + \left(|\Gamma_{s\eta_s}(1)| - 1\right) \frac{|\Gamma_{s\eta_s}(1)|}{\Gamma_{s\eta_s}^{max}} \tag{4.25}$$

with:

$$\begin{aligned} \eta_s &= \operatorname*{argmax}_l |\Gamma_{sl}(1)| \\ \Gamma_{s\eta_s}^{max} &= \max\{\Gamma_{s\eta_s}^{spk}(1), \Gamma_{s\eta_s}^{gen}(1)\}. \end{aligned}$$

The idea behind this formula is to choose $1/\beta_s$ according to the magnitude of $\Gamma_{s\eta_s}(1)$, as far as the ratio $|\Gamma_{s\eta_s}(1)|/\Gamma_{s\eta_s}^{max}$ is not too low. Otherwise, if the ratio is low, the contributions of $\Gamma_{s\eta_s}^{spk}(1)$ and $\Gamma_{s\eta_s}^{gen}(1)$ nearly cancel, which requires that β_s approaches a fixed limit. Otherwise the iteration constants would become very large, which leads to low convergence rates, as follows from Eqs. (4.26a) and (4.26b) below.

Using the reestimation equations for the pooled and state specific variances in the EB case (Eqs. (4.12a) and (4.12b)), the following estimations of the minimal iteration constants $D_{\min}$ and $D_{s,\min}$ respectively could be calculated, which fulfill the constraint of positive variances for each acoustic feature component:

$$\begin{aligned} D_{\min} = \frac{1}{\sum_s \sigma^2 - \sigma_{\min}} \cdot \Big[&- \Gamma(x^2) + \sigma_{\min}\Gamma(1) + \sum_{s,l} \left[2\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]\mu_{sl} \\ &+ \beta_s \sum_{s,l} \left[\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]^2 \Big], \end{aligned} \tag{4.26a}$$

$$D_{s,\min} = \frac{1}{\sigma_s^2 - \sigma_{\min}} \cdot \Big[-\Gamma_s(x^2) + \sigma_{\min}\Gamma_s(1) + \sum_l \left[2\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]\mu_{sl} + \beta_s \sum_l \left[\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]^2 \Big]. \tag{4.26b}$$

Finally, for reestimation with pooled variances, choose:

$$D = h \cdot \max\left\{ D_{\min}, \max_{s,l} \frac{1}{c_{sl}} \left[\frac{1}{\beta_s} - \Gamma_s(1) \right] \right\}, \tag{4.27a}$$

and, for reestimation with state specific variances, choose:

$$D_s = h \cdot \max\left\{ D_{s,\min}, \max_{l} \frac{1}{c_{sl}} \left[\frac{1}{\beta_s} - \Gamma_s(1) \right] \right\}. \tag{4.27b}$$

The terms in the maximization make sure that both the constraint on the denominators and on the variances are fulfilled, cf. Eqs. (4.23) and (4.24) respectively. The global factor $h > 1$ controls the convergence of the iteration process, high values leading to low step sizes.

Substituting the above choice of the iteration constant D_s for the case of state specific variance from Eq. (4.27b) into Eq. (4.17b), it could be realized that the constraint on the denominators in the EB case implies an upper bound of $\beta_s\sigma_s^2/h$ to the resulting step size of GD. This upper bound for the step sizes is reached only, if the step size estimated from the constraint of positive variance becomes too high. The same applies for the case of pooled variance.

4.5.1.2 Iteration Constants for the Mixture Weights

For the iteration constants of the mixture weights, C_s, an expression similar to that for the reestimation of the means and variances is applied, which makes sure that the denominators of the reestimation equations of the mixture weights are positive, non-singular (case $\epsilon_s = 0$), and their magnitude is near to the differences of the relative counts of the corresponding reestimation equations:

$$C_s = h \cdot \left[\max\left\{ -\max_l \left[\frac{\Gamma_{sl}^{spk}(1)}{\Gamma_s^{spk}(1)} - \frac{\Gamma_{sl}^{gen}(1)}{\Gamma_s^{gen}(1)} \right], 0 \right\} + \left\{ \begin{array}{ll} 1 & \text{iff } \epsilon_s = 0 \\ \epsilon_s & \text{else,} \end{array} \right\} \right],$$

with:

$$\epsilon_s = \max_l \left| \frac{\Gamma_{sl}^{spk}(1)}{\Gamma_s^{spk}(1)} - \frac{\Gamma_{sl}^{gen}(1)}{\Gamma_s^{gen}(1)} \right|.$$

4.5.2 Global Convergence Heuristics

In the experiments, factors of $h = 2$ for the *TI digit string* corpus and $h = 1.1$ for the *SieTill* and WSJ0 corpus were found to be optimal by means of convergence rate and recognition results on the training corpus.

In general convergence can only be guaranteed, if the step sizes used are sufficiently small. Therefore, in addition to the above step size estimates, the training procedure was based on the following method. In each iteration step it is checked, whether the word error rate on the training data was more than doubled. If this occured, the corresponding iteration step was removed by a fall-back to the previous reestimation, for which statistics were temporarily saved, and the reestimation was repeated with an iteration constant of $h = 5$. For the large vocabulary applications, this special step size control never became active – the error rate on the training data strictly decreased using $h = 1.1$ throughout all iterations.

4.5.3 Convergence Results for Small Vocabulary

Experiments for the convergence properties of discriminative training were performed for Corrective Training (CT) applying both the EB and GD optimization methods.

4.5.3.1 Overall Convergence

Using iteration factors $h = 1.1$ for pooled variances (*SieTill*) and $h = 2$ for state specific variances (*TI digit string*, cf. Figs. 4.1 and 4.3), relatively steady convergence was found for both GD and EB. Similar results could be observed for the word error rates on test and training data, as shown in Fig. 4.2 for the male portion of the *TI digit string* corpus.

Clearly, convergence on test and training data is comparable and thus the convergence of the error rate on the training data was used as criterion to stop an iteration.

It should be noted that, the optimization approach presented here has been successfully applied to MMI training of Gaussian mixture models for image object recognition [Dahmen$^+$ 1999], where similar convergence characteristics could be observed.

4.5.3.2 Non-Monotonic Behaviour

Although overall convergence could be observed, the CT criterion (Fig. 4.3) and the corresponding word error rates (Fig. 4.4) on the *SieTill* training corpus show non-monotonic behaviour in the course of the training iterations for both single and mixture Gaussian densities. Preliminary experiments using different iteration factors showed that despite the occurring increases the choice of $h = 1.1$ was optimal according to the convergence rate.

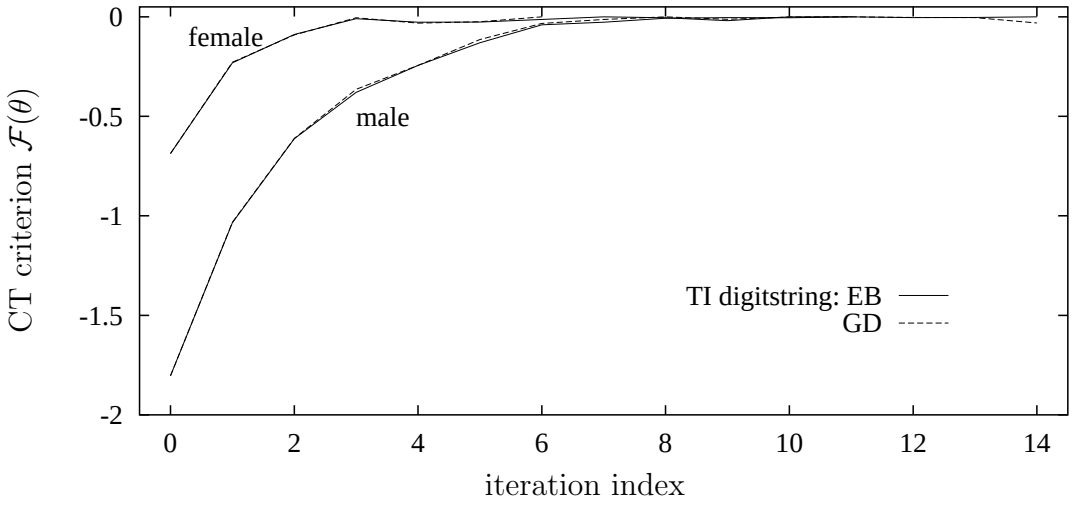

Figure 4.1: CT criterion as a function of the iteration index for single Gaussian densities (*TI digit string* training corpus).

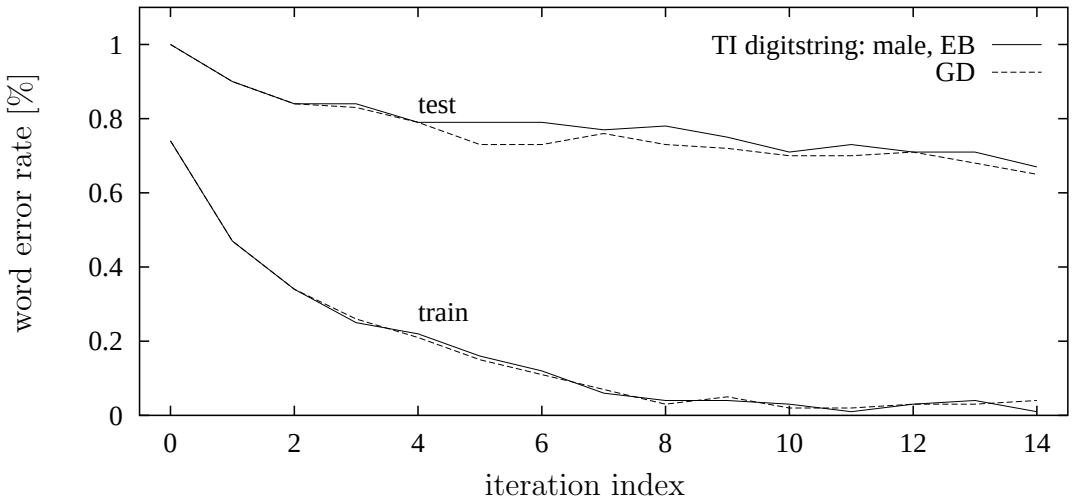

Figure 4.2: Word error rate as a function of the iteration index for CT using single Gaussian densities (male portion of the *TI digit string* corpus).

The same was observed when using the MCE and the MMI criterion, Figs. 4.5 and 4.6 showing a comparison of convergence for the MCE, MMI and CT criteria and the evolution of the corresponding error rates for the *SieTill* training corpus using 32 densities per state. Note that MCE and MMI were applied after 10 iterations of CT. Since the magnitude of the criteria differ[1], no direct comparison of convergence according to the criteria is possible. Even the recognition results on the training corpus give no clue to how the performance of the different criteria on unseen data would differ.

[1]Note that the MCE criterion was multiplied by a factor of 20 in Fig. 4.5

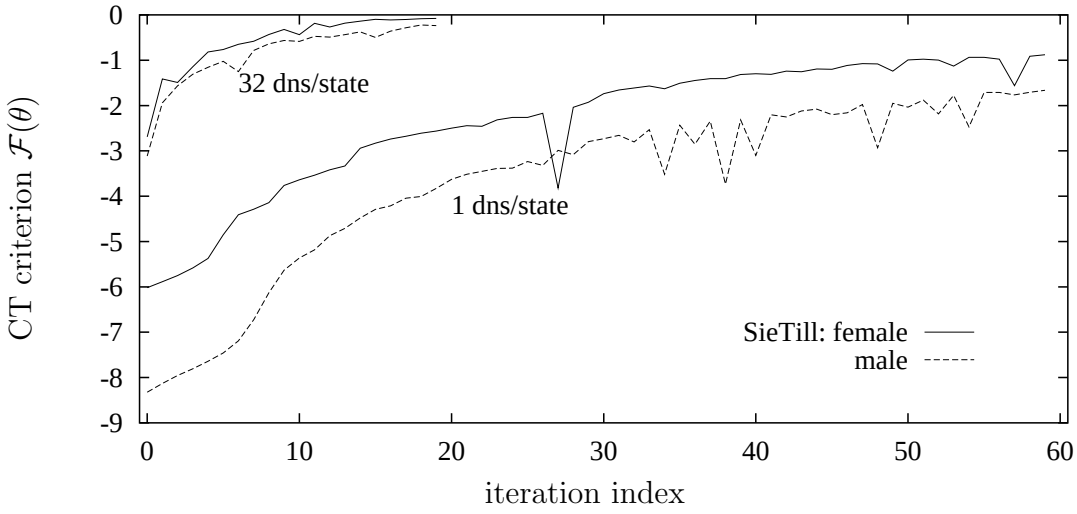

Figure 4.3: CT criterion as a function of the iteration index for single and mixture Gaussian acoustic models (*SieTill* training corpus).

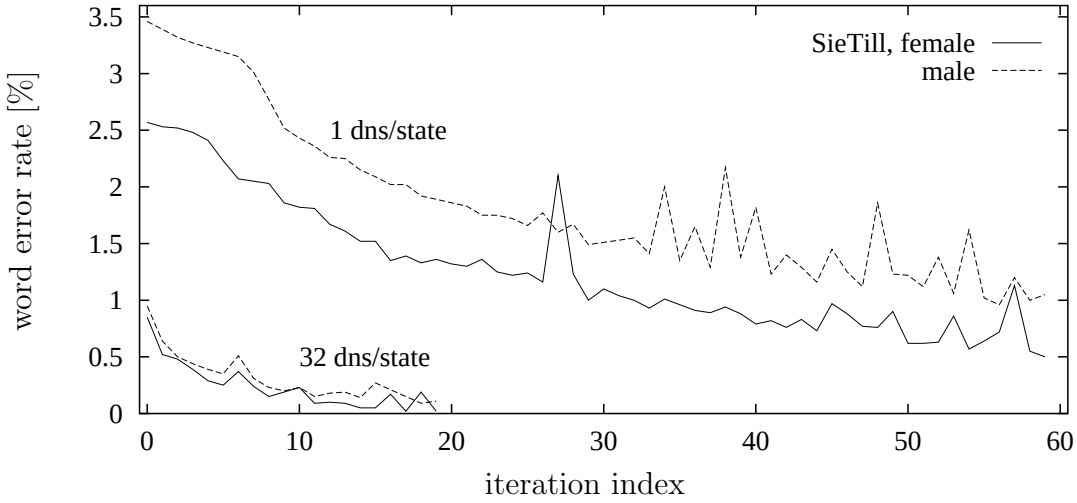

Figure 4.4: Word error rates on the training corpus as a function of the iteration index for Corrective Training (CT) using single and mixture Gaussian acoustic models (*SieTill* training corpus).

4.5.3.3 Choice of Parameter Sets

Because of the non-monotonic behaviour of the criteria in the training iterations, the parameter sets from discriminative training to be used for digit string recognition were chosen according to the best recognition results on the training corpus.

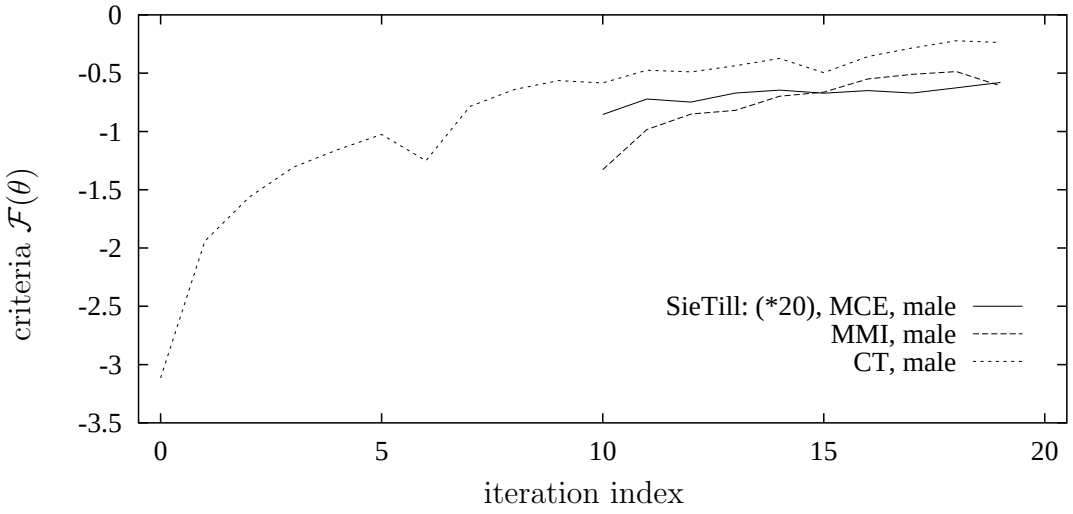

Figure 4.5: MCE, MMI and CT criterion as a function of the iteration index for mixture Gaussian acoustic models with 32 densities per state (*SieTill* training corpus). Note that the MCE and MMI training iterations begin after 10 iterations of Corrective Training (CT). The values of the MCE criterion were scaled by a factor of 20.

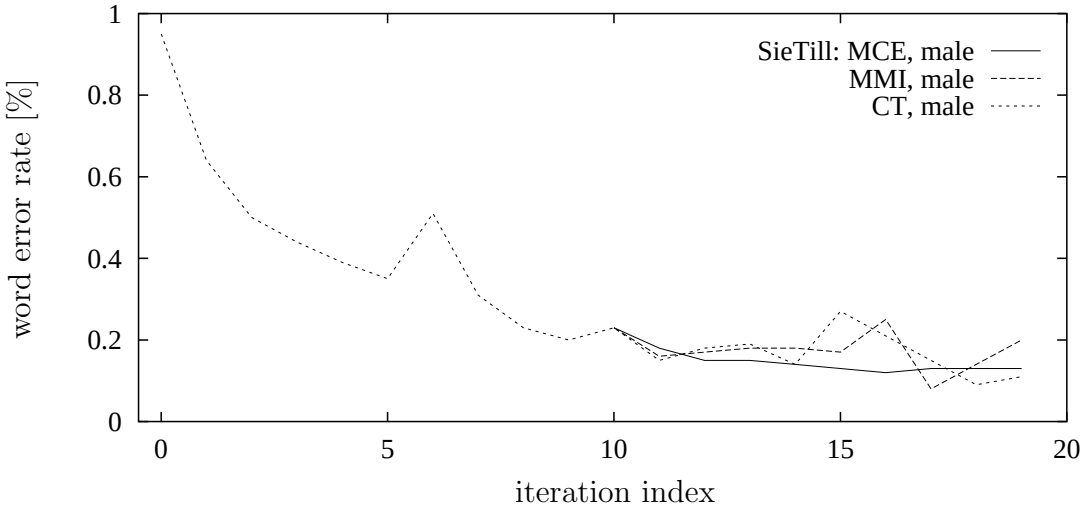

Figure 4.6: Word error rates on the training corpus as a function of the iteration index for the MCE, MMI and CT criterion using mixture Gaussian acoustic models with 32 densities per state (*SieTill* training corpus). Note that the MCE and MMI training iterations begin after 10 iterations of Corrective Training (CT).

4.5.4 Convergence for Large Vocabulary

When changing to discriminative training on large vocabulary tasks, the optimization methods developed for small vocabulary whole word recognizers were transferred.

4.5.4.1 Overall Convergence

Similar to the small vocabulary applications, for MMI training on the WSJ0 training corpus, good overall convergence could be observed for both the MMI criterion itself (Fig. 4.7) and very smoothly for the word error rate on the training data (Fig. 4.8). As for the case of discriminative training for small vocabulary speech recognition, local increases in the discriminative criterion where tolerated in order to high convergence rates.

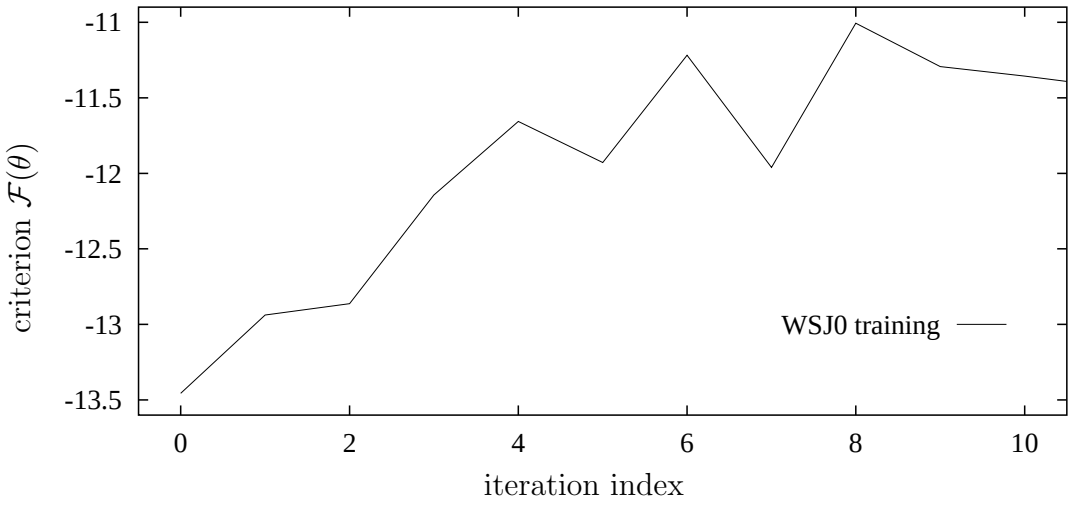

Figure 4.7: MMI criterion as a function of the iteration index for the WSJ0 training corpus.

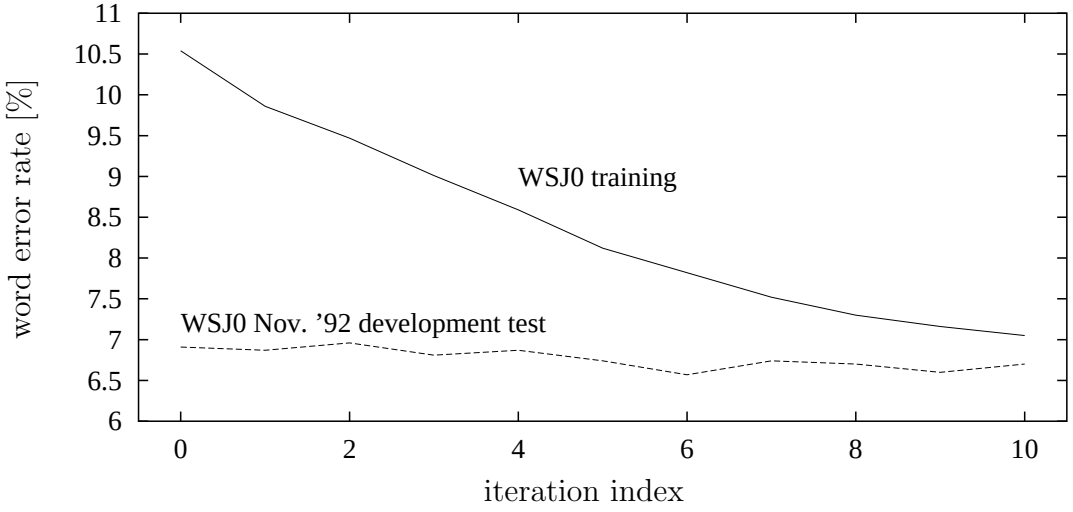

Figure 4.8: Word error rates as a function of the iteration index of an MMI training for the WSJ0 training corpus and the WSJ0 Nov. '92 development test set.

4.5.4.2 Choice of Parameter Sets

Note that, in contrast to the small vocabulary application (Fig. 4.2), convergence of the word error rates on the training corpus and on the development test set is not similar

(Fig. 4.8). Consequently, in case of large vocabulary, the choice of references for evaluation was made according to the best recognition results on the development test set.

Chapter 5

Discriminative Training for Speech Recognition

The aspects of discriminative training criteria and optimization methods considered so far in this work were fairly independent of the underlying pattern recognition applications in question. In this section, specific aspects of applying discriminative training to continuous speech recognition will be discussed.

5.1 Discriminative Training Algorithm

The course of discriminative training for speech recognition involves several steps. First, for each discriminative training iteration, sets of alternative word sequences have to be determined for each training utterance, cf. Section 3.1. For small vocabulary applications, this is usually done by a full recognition pass, whereas for large vocabulary applications a full recognition pass is performed only once before discriminative training in order to produce word graphs, which subsequently are used to restrict the recognition passes in each discriminative training iteration.

Next, discriminative statistics are determined. For this, the FB state probabilities for the spoken word sequences and the generalized FB state probabilities for the sets of alternative word sequences are evaluated and the results are used to accumulate the discriminative averages, cf. Section 4.1. Except for the experiments presented in Subsections 6.1.1 and 6.3.5, all experiments have been performed using the *Viterbi* approximation. In this case, the determination of the conventional FB state probability reduces to a time alignment of the spoken word sequences, and the determination of the generalized FB probabilities reduce to the calculation of word posterior probabilities (cf. Section 6.2) and time alignments of each alternative word in its word boundaries as determined by the recognition pass. In order to prevent redundant computations, the time alignments of the alternative words from the recognition pass are stored.

Finally, the parameters of the acoustic model are reestimated using the reestimation formulae derived in Sections 4.2 and 4.3.

5.1.1 Training Procedure and Complexity for Small Vocabulary

All discriminative trainings for small vocabulary were initialized with a standard ML training, which consist of a number of up to ten Expectation-Maximization (EM) iterations in *Viterbi* approximation followed by one mixture density splitting step. For the highest number of 64 densities per mixture presented here, one iteration of *Viterbi* training took about 2.5 hours for each gender, resulting in a real time factor (RTF) of about 0.4 on an ALPHA 5000 PC.

The discriminative training procedures following ML training were as follows. In order to speed up training times, several of the experiments were performed in an additive fashion, i.e. CT followed by MMI, MCE or further CT, as indicated in detail below. For single densities, each discriminative training was initialized with 20 iterations of CT, which served as common starting point for MMI and MCE training as well as further CT with 10 iterations each. For FT training of single density models, 30 iterations were performed following ML training. For mixture Gaussian densities with 32 densities per state, each discriminative training was initialized with 10 iterations of CT, which served as common starting point for MMI and MCE training respectively as well as further CT with 10 iterations each. For FT training of models with 32 densities per mixture 10 iterations were performed following ML training. For MMI and MCE training of models with 64 densities per mixture 15 iterations were performed following the initialization with ML training.

The acoustic models used for discriminative training were exactly the same than those used for ML training, i.e., for a given number of densities per mixture, the numbers of trained parameters all are the same for each training method considered.

In terms of computational complexity, the discriminative training methods discussed here are dominated by the recognition on the training data. Therefore the training times for MMI, CT, MCE and FT show only minor variations. One iteration of discriminative training took slightly more than 9 hours resulting in an RTF of about 1.5 on an ALPHA 5000 PC, which is about 3-4 times the time needed for one ML iteration.

5.1.2 Training Procedure and Complexity for Large Vocabulary

Discriminative training for large vocabulary was initialized with a standard ML training, which consist of a number of six Expectation-Maximization (EM) iterations in *Viterbi* approximation followed by one mixture density splitting step. For the optimal number of 96k mixture densities presented here, one iteration of *Viterbi* training took about 3.5 hours on the WSJ0 training corpus, resulting in a real time factor (RTF) of about 0.2 on an ALPHA 5000 PC.

Discriminative training used exactly the same structure of the acoustic models than ML training, with the same number of 96k mixture densities, resulting in a number of about 3.2 million free parameters.

For large vocabulary tasks, discriminative training methods become computationally very extensive. Most of the training time is needed for determination and calculation of the discriminative part of the criterion and the discriminative averages. Performing unconstrained recognition, initial word graphs were obtained for the approx. 15 hours of WSJ0 training data with a word graph density of 29. The word graphs took about 150 MB of disk space without compression.

Including the constrained recognition (cf. Section 5.3.2), the calculation of word probabilities and the reestimation process, a single iteration step of MMI training on the ARPA WSJ0 training corpus took about 1.5 days resulting in an RTF of about 2.3 on an ALPHA 5000 PC.

5.2 Alternative Word Sequences

The class of discriminative training criteria treated here always involves the determination of sets of competing word sequences for each training utterance. Because of the combinatorial complexity, it is certainly unrealistic to consider all possible word sequences with all possible word boundaries, especially for large vocabulary applications. Therefore alternative word sequences usually are determined by a full recognition pass, at least once.

5.2.1 Representation of Alternative Word Sequences

The alternative word sequences resulting from a recognition pass could either be represented by N-best lists or word graphs. Due to computational accuracy and complexity considerations, in this work N-best lists were not applied – for details cf. Section 6.2. Therefore, if not stated otherwise, the set of alternative word sequences, $\mathcal{M}_r$, will usually be associated with word graphs containing information on the recognized words and the corresponding word boundaries [Ortmanns$^+$ 1997a]. The information on word boundaries is used explicitly for training with the *Viterbi*-approximation, whereas for state summation, the word boundaries only are used to find the allowed predecessor and successor words of a given word in order to build the HMM according to the word graph.

5.3 Recognition for Discriminative Training

For the small vocabulary applications of discriminative training, unconstrained recognition was performed in every iteration step. For large vocabulary applications, unconstrained recognition for whole training corpora in every iteration of discriminative training would clearly be unrealistic by means of computation time.

5.3.1 MMI Training using Word Graphs with Fixed Word Boundaries

In [Valtchev$^+$ 1997], discriminative training using the WSJ SI-284 training corpus is reported, where unconstrained recognition was performed only once in order to produce an initial word lattice, which was then used for constrained recognition in each iteration step

of discriminative training. Preliminary experiments for discriminative training applying acoustic and language model rescoring on word graphs with fixed boundary times showed only little effect or even degradations in performance, as the recognition results on the WSJ corpus show in Table 5.1.

Table 5.1: Comparison of rescoring and constrained recognition using word graphs for the iterative determination of alternative word sequences during discriminative training. Results on the *Wall Street Journal* corpus (5k), training and recognition with bigram language model.

criterion	method	word error rates[%]		
		dev	eval	dev& eval
ML	–	6.91	6.78	6.86
MMI	rescoring	6.96	6.41	6.72
	constrained recognition	6.71	6.20	6.48

5.3.2 Constrained Recognition for Discriminative Training

In consequence, a method of constrained recognition has been developed in this work, where the boundary times are relaxed to intervals around the boundary times given by the word graph. At each time frame τ, where new word hypotheses are to be started, not only the word hypotheses starting at exactly this time frame in the word graph are allowed in this approach, but also those words starting at time frames in the vicinity of time frame τ defined by the interval $[\tau - \Delta\tau, \tau + \Delta\tau]$, as shown by a section of a word graph in Fig. 5.1. The successor word candidates thus obtained from the word graph are then used to reduce the possible search space by constraining the lexical tree, as illustrated in Fig. 5.2.

This method of extended constrained recognition even enables to recognize new word sequences not originally represented by the corresponding word graph, which would not be produced by simple acoustic or language model rescoring on the word graph, because boundary times of subsequent word hypotheses might not match. In addition the approach makes still use of the advantage of a tree lexicon. In the experiments reported here, a time interval of 11 frames was used, i.e. $\Delta\tau = 5$. In order to reduce computation time, the *Viterbi* state alignment paths from constrained recognition were saved on disk, such that they did not need to be estimated again word-wise for accumulation of statistics.

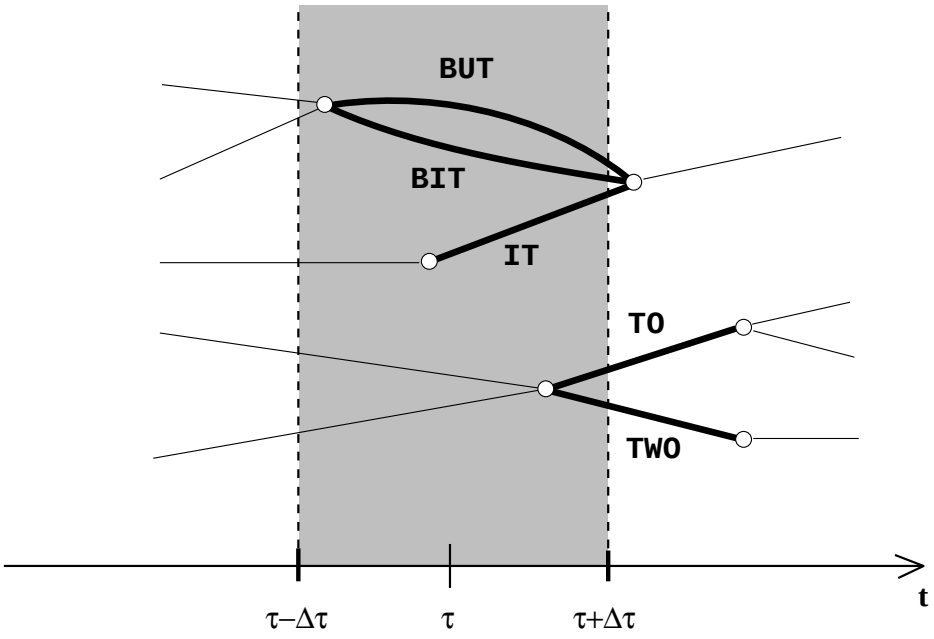

Figure 5.1: Constrained recognition: words of the word graph, which are allowed to be started at time τ.

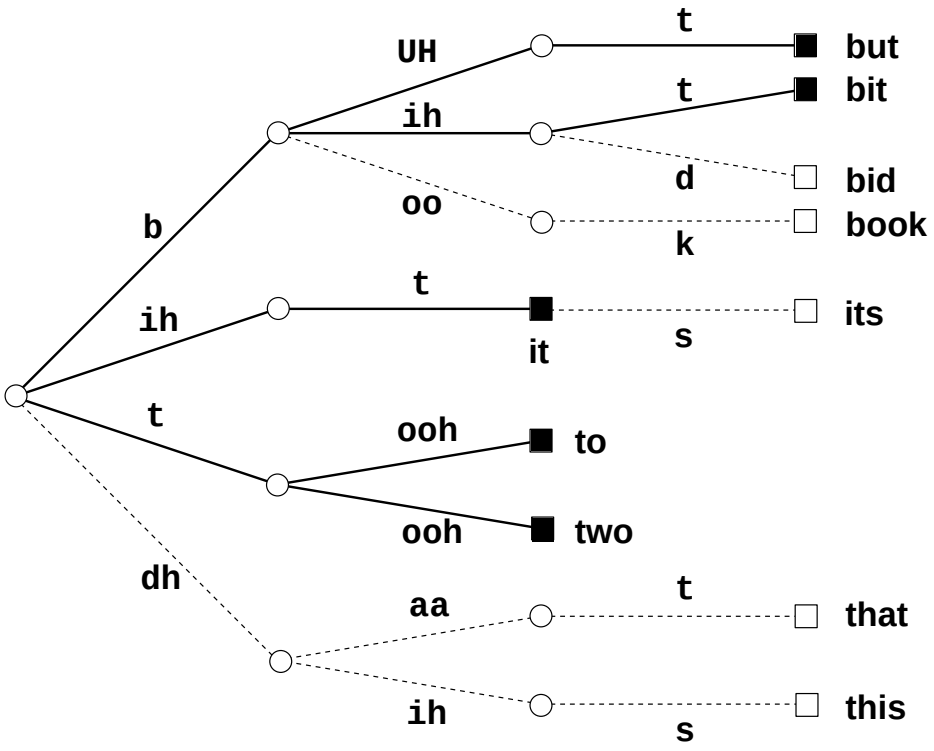

Figure 5.2: Lexical prefix tree for constrained recognition. Allowed words are marked with black squares.

5.3.3 Evaluation of Constrained Recognition

The completion of the unconstrained recognition pass on the WSJ0 training data took a bit less than a week on an ALPHA 5000 PC, resulting in a real time factor (RTF) of 10.4. Word graphs were produced with a word graph density of 29, which then were used for constrained recognition in every training iteration. The extended constrained recognition presented here reduced the corresponding recognition time by a factor of nearly 5, resulting in an RTF of 2.3 on the same machine.

As shown in Table 5.2, without any deterioration in recognition performance, the constrained recognition algorithm reduced the corresponding recognition time by a factor of more than 5, resulting in an RTF of 1.9 on an ALPHA 5000 PC. Note that these experiments were performed on unseen data. Therefore the corresponding RTFs differ from those given for training above.

Table 5.2: Comparison of full (unrestricted) recognition and constrained recognition using word graphs with $\Delta\tau = 5$. Recognition with bigram language model. The search space is indicated by the numbers of state, arc, tree, and word hypotheses. The real time factors (RTF) correspond to an ALPHA 5000 PC. Results on the ARPA WSJ0 Nov. '92 corpus.

recognition method	search space: number of				WER [%]	RTF
	states	arcs	trees	words		
full	6472	1835	36	106	6.86	10.5
constrained	989	239	17	67	6.86	1.9

5.4 Choice of Language Models

Another specific aspect of discriminative training for speech recognition is the choice of language models, which are supposed to be given throughout the training process. From the definition of discriminative training it is not at all clear, what the best choice of language models for discriminative training would be.

5.4.1 Possible Effects of Language Models on Discriminative Training

Firstly, there are three levels, at which the choice of language models might be important:

1. the determination of alternative word sequences;
2. the discriminative criterion; and
3. the correlation between training and recognition.

The first point should not have any considerable effect. In the worst case, a non-matching language model for the recognition of alternative word sequences would lead to missing word sequences in the word graphs, which should not be a problem, if the word graph

densities are high enough.

On the other hand, the word probabilities (cf. Section 6.2) directly depend on the language model chosen for the discriminative criterion, which should therefore have considerable effect on the training results. Two diametrical hypotheses shall clarify possible effects of the choice of language models for the discriminative criterion [Schlüter$^+$ 1999b].

Correlation hypotheses: On the one hand, one would expect that only those acoustic models need optimization, which do not sufficiently discriminate between correct and incorrect models. If this argument is valid, a strong correlation between the language models chosen for training and recognition has to be concluded.

Masking hypotheses: On the other hand, the language model usually largely improves the recognition accuracy and might mask deficiencies of the acoustic models. Such an effect would call for suboptimal language models *for training*. Moreover, the choice of language models for training should not considerably correlate with those chosen for recognition.

5.4.2 Experiments with Varying Language Models for MMI Training

In order to check the above hypotheses, extensive experiments using different language models for training and recognition were performed on the WSJ corpus. Table 5.3 gives an overview of the language models used here and the corresponding perplexities.

Table 5.3: Language model perplexities: ARPA WSJ0 training and testing corpora. The notations "bi-phr" and "tri-phr" refer to language models containing phrases/multiwords.

corpus	language model perplexity					
	zero	uni	bi	bi-phr	tri	tri-phr
Training	10110	1372	398	–	289	–
Nov. '92 Dev.	–	–	107	94	58	54
Nov. '92 Eval.	–	–	107	91	53	48

As shown in Table 5.4, discriminative training using a zerogram, unigram, bigram, and trigram language model has been performed. The initial recognition and the constrained recognition for the trigram training has been performed using the trigram language model, and the constrained recognitions for the zerogram, unigram, and bigram training were performed using the bigram.

For recognition with either bigram or trigram language models, clearly the best results are obtained using a unigram language model for the discriminative criterion resulting in relative improvements of about 10% in word error rate. Moreover, for recognition with the bigram, the training results for the trigram language model are even worse than for the zerogram. Even for recognition with the trigram, the training results for the trigram language model are only slightly better than those for the zerogram.

Table 5.4: Comparison of several language models for MMI training and recognition. Results on the *Wall Street Journal* corpus (5k).

language models		criterion	word error rates [%]		
recognition	training		dev	eval	dev & eval
bi	–	ML	6.91	6.78	6.86
	zero	MMI	6.71	6.03	6.41
	uni		**6.59**	**6.00**	**6.33**
	bi		6.71	6.20	6.48
	tri		6.87	6.54	6.72
tri	–	ML	4.82	4.11	4.51
	zero	MMI	4.63	4.05	4.38
	uni		**4.30**	**3.64**	**4.01**
	bi		4.48	3.94	4.24
	tri		4.58	4.00	4.33
bi-phrase	–	ML	6.40	5.79	6.13
	bi	MMI	5.91	5.60	5.78
tri-phrase	–	ML	4.76	4.26	4.54
	bi	MMI	4.48	4.07	4.30

5.4.3 Interdependence between Language Models for Recognition and MMI Training

In another experiment, the correlation between the language models chosen for training and recognition was examined. For this case, the ML training results were compared to discriminative training with the bigram language model. As shown in Table 5.4, the improvements obtained by discriminative training in comparison to ML training remain about the same for recognition with the bigram, trigram, phrase-bigram and phrase-trigram language model. The relative improvements in word error rate range between 5 and 6% in all these cases.

It should be noted that both sets of experiments clearly support the masking hypothesis, which means that language models chosen for discriminative training have to be somewhat less accurate than the optimal language model according to the recognition. Moreover, the experiments presented here indicate that results obtained by discriminative training using a particular language model are fairly independent of the choice of language models for recognition.

5.5 Spontaneous Speech

Up to now, discriminative training for large vocabulary has only been performed for clean speech corpora [Valtchev$^+$ 1996, Valtchev$^+$ 1997, Schlüter$^+$ 1999b], for which significant

improvements in word error rate have been reported.

Experiments for MMI training on spontaneous speech are now performed on the *Verbmobil I* corpus (cf. Annex A) as a continuation of this work. At the current stage, no statement on the performance of MMI training for spontaneous speech is possible.

Chapter 6

Efficient Estimation of Discriminative Statistics

The evaluation of discriminative training criteria always involves some kind of discriminative model, i.e. discriminative training optimizes the correct model against some competing or confusable model. For MMI and MCE training, the discriminative model involves a sum over a set of alternative or competing classes. For continuous speech recognition, this means a sum over all possible word sequences (of unknown length), which, especially for large vocabulary tasks, clearly leads to very high complexity.

Firstly, the sets of competing word sequences have to be defined, which usually involves some kind of recognition process on the training data, which has been discussed in Chapter 5. Secondly, the sums over all word sequences defined by these sets have to be calculated efficiently.
One possibility would be to represent the competing word sequences in N-best lists, as the definition of the generalized *forward-backward* (FB) probability suggests (cf. Eq. (4.4)):

$$\gamma_{rt}(s) = \sum_{W \in \mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum_{V \in \mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)} \cdot \gamma_{rt}(s;W).$$

The disadvantage of the use of N-best lists would be that most of the calculations done here would be redundant, since the corresponding word sequences usually only differ by a few words [Normandin$^+$ 1994b].

An alternative to the use of N-best lists is discriminative training using word graphs [Normandin$^+$ 1994b, Valtchev$^+$ 1996]. The idea for word graph based discriminative training is based on the conventional *forward-backward* algorithm known from *Baum-Welch* training of HMM parameters. It will be seen that word graph based discriminative training using the *Viterbi* approximation [Valtchev$^+$ 1996] involves an efficient summation over word sequences, which is very much analogous to the case of summing over state sequences in the case of *Baum-Welch* training. In Section 6.2 a full description of this algorithm for the use of m-gram language models will be given. The method will be extended to those cases (e.g. MCE), where specific word sequences have

to be excluded from the set of competing word sequences.

Furthermore, in Section 6.3 a FB algorithm for MMI training *without Viterbi* approximation will be presented, which therefore includes efficient summation over state sequences which correspond to all word sequences included in the set of alternative word sequences.

6.1 Word Sequence Based HMM-State Posterior Probabilities

For the evaluation of the *Baum-Welch* reestimation equations, the *forward-backward* (FB) probabilities, i.e. the posterior probabilities to hypothesize state s at time t, given the acoustic observations X_r and the spoken word sequence W_r of training utterance r have to be calculated (cf. Eq. (4.1)),

$$\begin{aligned}\gamma_{rt}(s;W_r) &= p_\theta(s_t = s|X_r, W_r) \\ &= \frac{p_\theta(s_t = s, X_r|W_r)}{p_\theta(X_r|W_r)}.\end{aligned}$$

Considering the HMM structure, the corresponding state summation could be separated into two independent state summations. These form the corresponding forward and backward calculations as indicated in Fig. 6.1:

$$\begin{aligned}p_\theta(s_t = s, X_r|W_r) &= \sum_{\substack{s_1^{T_r}:W_r \\ s_t=s}} p_\theta(s_1^{T_r}, {x_r}_1^{T_r}) \\ &= \underbrace{\Big[\sum_{\substack{s_1^t:W_r \\ s_t=s}} p_\theta(s_1^t, {x_r}_1^t)\Big]}_{=:\, a_{rt}(s|W_r)} \cdot \frac{1}{p_\theta(s_t, x_{rt})} \cdot \underbrace{\Big[\sum_{\substack{s_t^{T_r}:W_r \\ s_t=s}} p_\theta(s_t^{T_r}, {x_r}_t^{T_r})\Big]}_{=:\, b_{rt}(s|W_r)},\end{aligned}$$

where the first sum is taken over those state sequences $s_1^{T_r}$ of length T_r, which correspond with word sequence W_r. The corresponding forward, $a_{rt}(s|W_r))$, and backward probabilities, $b_{rt}(s|W_r)$, could then be obtained by the following recursion equations:

$$\begin{aligned}a_{rt}(s|W_r) &= p_\theta(x_{rt}|s, W_r) \cdot \sum_{\sigma:W_r} a_{r,t-1}(\sigma|W_r) \cdot p(s|\sigma) \\ b_{rt}(s|W_r) &= p_\theta(x_{rt}|s, W_r) \cdot \sum_{\sigma:W_r} b_{r,t+1}(\sigma|W_r) \cdot p(\sigma|s).\end{aligned}$$

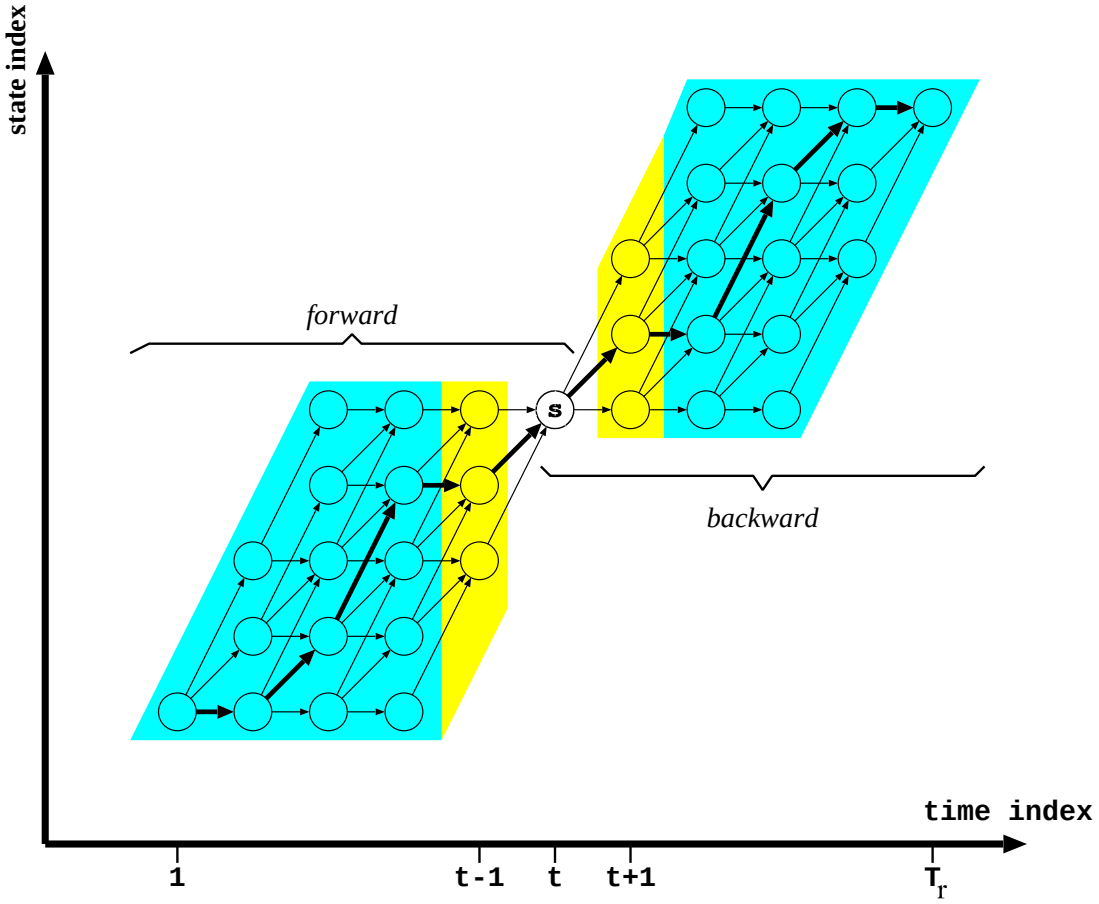

Figure 6.1: Time-state space for calculation of *forward-backward* (FB) probability $\gamma_{rt}(s|W_r)$. Highlighted by thick arrows is a possible *Viterbi* alignment path.

6.1.1 ML Training: *Viterbi* Approx. vs. State Summation

In order to evaluate the effect of the *Viterbi* approximation on ML training, comparative experiments were performed for both *Viterbi* and *Baum-Welch* ML training on the *Verbmobil* task and on the *SieTill* corpus. As shown in Tables 6.1 and 6.2, no significant differences could be observed for both ML training approaches.

Table 6.1: ML training using the *Viterbi* approximation or the *Baum-Welch* algorithm. Results for *Verbmobil I*, evaluation corpus 1996 (cf. Appendix A.2.1).

criterion	method	dns	word error rates[%]			
			del	ins	sub	WER
ML	Viterbi	166k	3.4	2.5	10.1	16.0
	BW	156k	3.2	2.8	10.1	16.0

6.2 Word Posterior Probabilities

6.2.1 Definition of Word Probabilities

Using word graphs and applying the *Viterbi* approximation, the generalized FB probability could equally well be divided into forward and backward calculations as the conventional FB probability in the case of *Baum-Welch* training, i.e., the words of the word graph in the former calculation correspond to the allowed states of the latter [Valtchev$^+$ 1996]. Therefore the generalized FB probability could be written as follows,

$$\begin{aligned}\gamma_{t,r}(s) &\overset{\text{Viterbi}}{\approx} \sum_{W\in\mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum\limits_{V\in\mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)}\delta(s,s_t(X_r,W)) \\ &\overset{\text{Viterbi}}{=} \frac{\sum\limits_{\tau,\tilde{t}:\tau\le t\le\tilde{t}} q([w;\tau,\tilde{t}],X_r)\;\delta_{s,s_t(x_r{}_\tau^{\tilde{t}},w)}}{\sum\limits_\tau\sum\limits_w q([w;\tau,T_r],X_r)}\end{aligned}$$

with the *Viterbi* alignment path $s_t(x_r{}_\tau^t, w)$ for word w with start and end time τ and t respectively. Interpreting $(w;\tau,t)$ as the equivalence class containing all word sequences W, which contain the word w with start and end time τ and t respectively, the definition of the word probability $q_{\tau,t}(w;X_r)$ is given by:

$$\begin{aligned}q([w;\tau,t],X_r) &= \sum_{\substack{W\in\mathcal{M}_r:\\ [w;\tau,t]\in W}} p_\theta^\alpha(x_r{}_1^{T_r}|W)\cdot p^\alpha(W) \\ &= \sum_{n\in\mathbb{N}}\sum_{\eta\in\mathbb{N}} \sum_{\substack{v_1^n,u_1^\eta:\\ (v_1^n,w,u_1^\eta)\in\mathcal{M}_r}} p_\theta^\alpha(x_r{}_1^{t_{s-1}}|v_1^n)\cdot p_\theta^\alpha(x_r{}_\tau^t|w)\cdot p_\theta^\alpha(x_r{}_{t+1}^{T_r}|u_1^\eta)\cdot p^\alpha(v_1^n,w,u_1^\eta).\end{aligned} \tag{6.1}$$

It should be noted that, the same word probabilities build the basis for the calculation of confidence measures on word graphs, which significantly reduced the confidence error rate on a variety of tasks [Wessel$^+$ 1998, Wessel$^+$ 2000b].

6.2.2 Decomposition into Forward and Backward Word Probabilities

The sum in Eq. (6.1) is already separated into a sum over the preceding word sequences v_1^n of w; and the successing word sequences u_1^η of word w. Using an m-gram language model, the history (of predecessors) of a word is defined as $h = (h_1, ..., h_{m-1})$. Correspondingly, the future (of successors) of a word is defined as $f = (f_1, ..., f_{m-1})$. Now, in order to properly consider the structure of an m-gram language model, the above sum is further structured into sums over histories h closing with word w and all word sequences v_1^n closing with history h; as well as sums over futures f beginning with word w and all word

sequences u_1^η ending with future f. Interpreting h as the equivalence class containing all word sequences closing with h; and f as the equivalence class containing all word sequences starting with f, the word probability could be written as:

$$
\begin{aligned}
q([w;\tau,t],X_r) = \sum_{\substack{h,f:\\ h_{m-1}=w=f_1}} \Bigg\{ & \underbrace{\Bigg[\sum_{\substack{n\in\mathbb{N};v_1^n\in\mathcal{M}_r:\\ v_{n-m+3}^n=h_1^{m-2}}} p_\theta^\alpha(x_{r_1}^{\ \tau-1}|v_1^n)\cdot p_\theta^\alpha(x_{r_\tau}^{\ t}|w)\cdot p^\alpha(v_1^n)\Bigg]}_{\Phi_r(h_1^{m-2},[w;\tau,t])}\cdot \\
& \cdot \frac{1}{p_\theta^\alpha(x_{r_\tau}^{\ t}|w)} \cdot \prod_{i=2}^{m-1} p^\alpha(f_i|h_{i-1}^{m-2},f_1^{i-1}) \cdot \\
& \cdot \underbrace{\Bigg[\sum_{\substack{\eta\in\mathbb{N};u_1^\eta\in\mathcal{M}_r:\\ u_1^{m-2}=f_2^{m-1}}} p_\theta^\alpha(x_{r_{t+1}}^{\ T_r}|u_1^\eta)\cdot p_\theta^\alpha(x_{r_\tau}^{\ t}|w)\cdot p_\theta^\alpha(u_{m-1}^\eta|f_1^{m-1})\Bigg]}_{\Psi_r([w;\tau,t],f_2^{m-1})} \Bigg\} \\
= \sum_{\substack{h,f:\\ h_{m-1}=w=f_1}} & \frac{\Phi_r(h_1^{m-2},[w;\tau,t])\cdot\Psi_r([w;\tau,t],f_2^{m-1})}{p_\theta^\alpha(x_{r_\tau}^{\ t}|w)} \cdot \prod_{i=2}^{m-1} p^\alpha(f_i|h_{i-1}^{m-2},f_1^{i-1})
\end{aligned}
\tag{6.2}
$$

For a very simple case of a word with two predecessor words and two successor words, the *Viterbi* paths that contribute to the word posterior probability are highlighted by thick arrows in Fig. 6.2.

Eq. (6.2) includes some kind of language model correction term. For language models with $m > 2$, this correction term considers those contributions, which are not already considered within the backward probability, because the corresponding histories are not already known.

6.2.3 Forward and Backward Word Recursions

Similar to the *Baum-Welch* case, the recursion equations for the word graph based forward, $\Phi_r(h_2^{m-1},[w;\tau,t])$, and backward probabilities, $\Psi_r([w;\tau,t],f_2^{m-1})$, are given by:

$$
\begin{aligned}
\Phi_r(h_2^{m-1},[w;\tau,t]) &= p_\theta^\alpha(x_{r_\tau}^{\ t}|w)\cdot\sum_{\tilde\tau,h_1}\Phi_r(h_1^{m-2},[h_{m-1};\tilde\tau,\tau-1])\cdot p(w|h) \\
\Psi_r([w;\tau,t],f_1^{m-2}) &= p_\theta^\alpha(x_{r_\tau}^{\ t}|w)\cdot\sum_{\tilde t,f_{m-1}}\Psi_r([f_1;t+1,\tilde t],f_2^{m-1})\cdot p(f_{m-1}|w,f_1^{m-2}).
\end{aligned}
$$

6.2.4 Disadvantages of N-Best Lists

It should be noted that this work refrains from using N-best approaches to discriminative training at all, since word graph based calculations both are by far more efficient *and*

more exact. N-best lists usually are limited by a maximum number of word sequences they should contain. On the other hand, it would be desirable to have the number of competing words for a given spoken word more or less constant, i.e., the density of the word graphs representing the sets of competing word sequences should be more or less constant. For increasing length of the training utterances, it could be shown that constant word graph density in general would result in an exponentially increasing number of competing word sequences included in the corresponding word graphs. Hence, the efficient word graph calculation not only circumvents redundancy, it also makes possible to consider many more competing word sequences, as would be feasible using N-best lists.

Using the algorithm for calculating the word graph based word probabilities, the number of word sequences contained in a word graph could easily be obtained by setting all language model and emission probabilities to 1. This calculation was performed in [Wessel$^+$ 1999] for speech data with an average utterance length of 18 words and an average word graph density of 131.6. The resulting average number of word sequences contained in a word graph was more than 10^{38}. A corresponding N-best list approach, which reaches about the same efficiency would not contain considerably more word sequences as the number of competing words per spoken word, this is the word graph density in a word graph based approach, which would be around 130 in this case.

6.3 Word Graph based HMM-State Posterior Probabilities

Although slightly more complex, also the combined case of word sequence and state summation could be divided into a *forward-backward*-like procedure, where now both transitions between words and between states have to be considered.

6.3.1 Definition of State Probabilities

If no approximations to state and word sequence summation are made, the generalized FB probability could be written as follows (cf. Eq. (4.4)):

$$\begin{aligned}
\gamma_{rt}(s) &= \sum_{W\in\mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum\limits_{V\in\mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)} \cdot \gamma_{rt}(s;W) \\
&= \sum_{W\in\mathcal{M}_r} \frac{p_\theta^\alpha(X_r|W)p^\alpha(W)}{\sum\limits_{V\in\mathcal{M}_r} p_\theta^\alpha(X_r|V)p^\alpha(V)} \cdot \frac{p_\theta(s_t=s,X_r|W)}{p_\theta(X_r|W)} \\
&= \frac{g_{rt}(s;X_r)}{\sum\limits_{\sigma} g_{rT_r}(\sigma;X_r)},
\end{aligned} \tag{6.3}$$

with the definition of the state probability:

$$g_{rt}(s; X_r) = \sum_{W \in \mathcal{M}_r} p_\theta^{\alpha-1}(X_r|W) \cdot p_\theta(s_t = s, X_r|W) \cdot p^\alpha(W). \tag{6.4}$$

6.3.2 Handling of the Smoothing Exponent

An efficient *forward-backward* separation of the summation in Eq. (6.4) is only possible, if either the exponent α is equal to one, or the exponent α is included into the state summation. Since the former is included in the latter, the state summation of the acoustic emission probabilities is therefore *defined* by:

$$p_\theta^\alpha(X_r|W) := \sum_{s_1^{T_r}:W} p^\alpha(s_1^{T_r}, X_r). \tag{6.5}$$

It should be noted that, for exponents $\alpha \neq 1$, this means a slight but consistent change in the criteria included in the unifying approach. Nevertheless, the MMI criterion remains unchanged by this definition (because $\alpha = 1$ for MMI).

Using the definition of Eq. (6.5), the derivatives of the unified criterion have to be recalculated, and the result is the following variant to the word graph based state probability, while the formal expression for the generalized FB probability of Eq. (6.3) are retained:

$$\begin{aligned} g_{rt}(s; X_r) &= \sum_{W \in \mathcal{M}_r} p_\theta^\alpha(s_t = s, X_r|W) \cdot p^\alpha(W) \\ &= \sum_{W \in \mathcal{M}_r} \sum_{\substack{s_1^{T_r}:W \\ s_t = s}} p_\theta^\alpha(s_1^{T_r}, X_r) \cdot p^\alpha(W). \end{aligned}$$

6.3.3 Decomposition into Forward and Backward State Probabilities

The sum over word sequences W and state sequences $s_1^{T_r}$ will now be separated as it was done for the states or words alone, cf. Sections 6.1 and 6.2. The word sequence is separated into $W = (v_1^n, w, u_1^\eta)$. Furthermore, by separation of the corresponding state sequence at time t and by using the word sequence summation according to history h and future f as defined in Section 6.2, the following structured calculation of the word-sequence-independent state probability is obtained:

$$
\begin{aligned}
g_{rt}(s;X_r) &= \sum_{W\in\mathcal{M}_r} \sum_{\substack{s_1^{T_r}:W\\ s_t=s}} p_\theta^\alpha(s_1^{T_r}, x_{r1}^{\ T_r}) \cdot p^\alpha(W) \\
&= \sum_{\substack{w\in\mathcal{M}_r:\\ s\in w}} \sum_{\substack{h,f:\\ h_{m-1}=w=f_1}} \Bigg\{ \underbrace{\Bigg[\sum_{\substack{v_1^n\in\mathcal{M}_r:n\in\mathbb{N},\\ v_{n-m+3}^n=h_2^{m-1}}} \sum_{\substack{s_1^t:(v_1^n,w),\\ s_t=s}} p_\theta^\alpha(s_1^t, x_{r1}^{\ t}) \cdot p^\alpha(v_1^n) \Bigg]}_{\phi_{rt}(h;s)} \\
&\qquad \cdot \frac{1}{p_\theta^\alpha(s,x_{rt})} \cdot \prod_{i=2}^{m-1} p^\alpha(f_i|h_{i-1}^{m-2}, f_1^{i-1}) \\
&\qquad \cdot \underbrace{\Bigg[\sum_{\substack{u_1^\eta\in\mathcal{M}_r:\eta\in\mathbb{N},\\ u_1^{m-2}=f_2^{m-1}}} \sum_{\substack{s_t^{T_r}:(w,u_1^\eta),\\ s_t=s}} p_\theta^\alpha(s_t^{T_r}, x_{rt}^{\ T_r}) \cdot p_\theta^\alpha(u_{m-1}^\eta|f_1^{m-1}) \Bigg]}_{\psi_{rt}(s;f)} \Bigg\} \\
&= \sum_{\substack{w\in\mathcal{M}_r:\\ s\in w}} \sum_{\substack{h,f:\\ h_{m-1}=w=f_1}} \frac{\phi_{rt}(s;h)\cdot\psi_{rt}(s;f)}{p_\theta^\alpha(s,x_{rt})} \cdot \prod_{i=2}^{m-1} p^\alpha(f_i|h_{i-1}^{m-2}, f_1^{i-1}).
\end{aligned}
$$

For a very simple case of a word with two predecessor words and two successor words, the corresponding calculation is illustrated in Fig. 6.2. For this example, four possible word sequences contribute to the probability of state s at time t.

The above summations over state sequences are to be interpreted as follows. The symbol $s_1^{T_r} : W, s_t = s$ denotes all possible state sequences $s_1^{T_r}$ with $s_t = s$, which are compatible with the word sequence W, with s_1 the first state of the first word of W and s_{T_r} the last state of the last word of W. Moreover, $s_1^t : (v_1^n, w), s_t = s$ denotes all possible state sequences s_1^t with $s_t = s$, which are compatible with the word sequence (v_1^n, w), with s_1 the first state of word v_1 and s_t a state of word w, i.e. s_1^t ends not necessarily at the end of w. In a similar manner, $s_t^{T_r} : (w, u_1^\eta), s_t = s$ denotes all possible state sequences $s_t^{T_r}$ with $s_t = s$, which are compatible with the word sequence (w, u_1^η), with s_{T_r} the last state of word u_η and s_t a state of word w, i.e. $s_t^{T_r}$ starts not necessarily at the beginning of word w. As was the case for the word graph based word probabilities, for language models with $m > 2$ (cf. Eq. (6.2)), the equation of the word-sequence-independent state *forward-backward* probability also includes a language model correction term in order to include those contributions, which could not be considered within the backward state probability, since the corresponding histories are not already known.

The corresponding forward state probability $\phi_{rt}(s;h)$ now has to be interpreted as the probability to hypothesize state s of word h_{m-1} at time t, given the acoustic observations of utterance r up to time t, and all word sequences allowed by the set of competing word sequences $\mathcal{M}_r$ that close with word sequence h. Similarly, the backward state probability

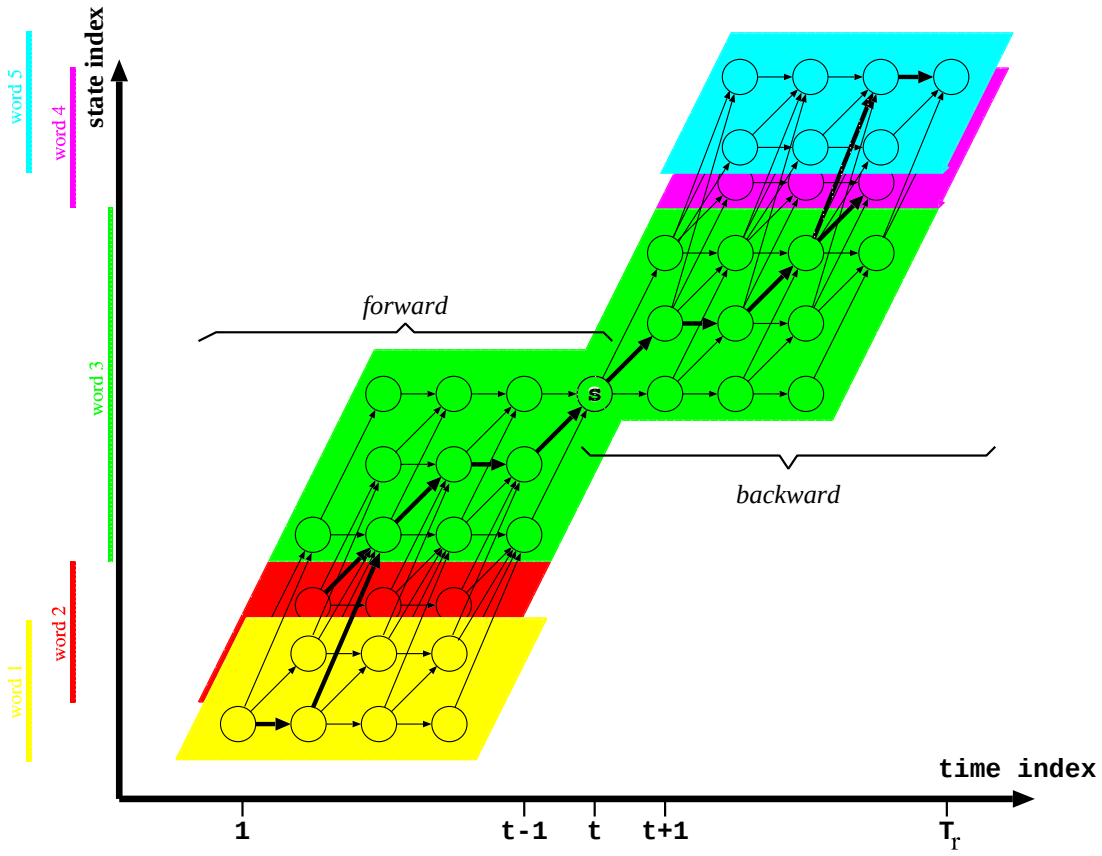

Figure 6.2: Time-state space for calculation of generalized *forward-backward* (FB) probability $\gamma_{rt}(s|W_r)$. Highlighted by thick arrows are possible *Viterbi* alignment paths for the individual words, as they would contribute to the word posterior probabilities.

$\psi_{rt}(s;f)$ has to be interpreted as the probability to hypothesize state s of word f_1 at time t, given the acoustic observations of utterance r beginning at time t, and all word sequences allowed by the set of competing word sequences $\mathcal{M}_r$ that begin with word sequence f.

6.3.4 Forward and Backward State Recursions

It has to be noted that, for the recursion equations of the forward and backward state probabilities, both state transitions *within* words as well as *between* words do occur, where, for the latter, language model probabilities have to be considered. Now define σ_h as the closing states of word h_{m-1}; and σ_f as the starting states of word f_1, then the recursion equations are given by:

$$\phi_{rt}(h;s) = p(x_{rt}|s) \cdot \Big\{ \sum_{v} \sum_{\sigma_h} \phi_{r,t-1}(v, h_1^{m-2}; \sigma_h) \cdot p(s|\sigma_h) \cdot p(h_{m-1}|v, h_1^{m-2}) + \sum_{\sigma} \phi_{r,t-1}(h;\sigma) \cdot p(s|\sigma) \Big\},$$

$$\psi_{rt}(s;f) = p(x_{rt}|s) \cdot \Big\{ \sum_{u} \sum_{\sigma_f} \psi_{r,t-1}(\sigma_f; f_2^{m-1}, u) \cdot p(\sigma_f|s) p(u|f) + \sum_{\sigma} \psi_{r,t-1}(\sigma; f) \cdot p(\sigma|s) \Big\}.$$

These recursion equations could be evaluated using dynamic programming similar to a full search procedure, which has been implemented for the whole word based small vocabulary speech recognition system. A more efficient implementation could also use initially recognized word graphs.

6.3.5 MMI Training: *Viterbi* Approx. vs. State Summation

As for the ML case, experiments were performed on the *SieTill* corpus, in order to compare MMI training using the *Viterbi* approximation and exact state summation.

As shown in Table 6.2, no significant differences occur for the case of single Gaussian densities, whereas for mixture densities with 8 densities per mixture a relative improvement of nearly 1/6 was obtained for MMI training with exact state summation in comparison to MMI training using the *Viterbi* approximation.

Finally, for mixture densities with 32 densities per mixture, no significant improvement has been observed. Hence, on this task MMI training with exact state summation gives improvements over MMI training with *Viterbi* approximation for intermediate model complexity only, but no improvements for acoustic models of low, as well as optimal complexity.

Table 6.2: MMI and ML training using the *Viterbi* approximation and exact state summation. Results for the *SieTill* corpus.

dns	criterion	method	del - ins - sub	WER[%]	SER[%]
1	ML	Viterbi	304-270-1053	3.78	9.74
		BW	330-221-1059	3.74	9.64
	MMI	Viterbi	349-175- 684	2.81	7.13
		BW	261-252- 695	2.81	7.31
8	ML	Viterbi	190-298- 579	2.48	6.47
		BW	198-301- 528	2.39	6.39
	MMI	Viterbi	250-213- 512	2.28	5.97
		BW	170-152- 497	1.91	5.21
32	ML	Viterbi	198-201- 450	1.97	5.31
		BW	225-212- 412	1.97	5.21
	MMI	Viterbi	182-161- 407	1.74	4.80
		BW	181-163- 402	1.75	4.59
64	ML	Viterbi	198-162- 419	1.81	4.93
		BW	164-217- 404	1.82	4.90

6.4 MCE Training using Word Graphs

If specific word sequences have to be excluded from the set $\mathcal{M}_r$ of competing word sequences, a problem occurs for the algorithms using word graphs for efficient discriminative training. This, for example, is the case for MCE training, where the spoken word sequence has to be excluded from the set of competing word sequences. In this case $\mathcal{M}_r$ could not completely be represented by a word graph. In general, specific word sequences could not be excluded from a word graph, without coincidentally excluding other word sequences, because particular word hypotheses of the spoken word sequence might be part of other sequences, too.

6.4.1 Competing-Word Probabilities for MCE Training

In order to be still able to make efficient use of the word graph representation as opposed to N-best based calculations, the word graph based algorithms presented in Sections 6.2 and 6.3 have to be extended to the case of MCE training. The sum over all word sequences in the word graph (represented by $\mathcal{M}_r$) *including* the spoken word sequence is performed first, which afterwards is subtracted by the probability of the spoken word sequences if necessary. For the word graph based word probability, this could be written as (cf. Eq. (6.1)):

$$
\begin{aligned}
q_{\mathrm{MCE}}([w;\tau,t],X_r) &= \sum_{\substack{W\in\mathcal{M}_r\setminus W_r:\\ [w;\tau,t]\in W}} p_\theta^\alpha(x_{r1}^{T_r},W) \\
&= q([w;\tau,t],X_r) \quad - \sum_{\substack{W=W_r:\\ [w;\tau,t]\in W}} p_\theta^\alpha(x_{r1}^{T_r},W).
\end{aligned}
\tag{6.6}
$$

Note that a word graph could contain multiple copies of the spoken word sequence with different word boundary times. This is reflected in the sum subtracted in Eq. (6.6).

Similar extensions could be made to the corresponding word graph based state probabilities of Section 6.3. For all calculations involving the word graph based word or state probability for MCE training, this extended version has to be used instead of the usual definitions given in Sections 6.2 and 6.3.

Chapter 7

Refined Scoring Approaches

Based on the calculation of word and state posterior probabilities, in this section a novel approach to the computation of the posterior probability of a word sequence given a sequence of acoustic observation vectors is presented, as it is needed for the realization of *Bayes*' decision rule to optimize sentence error rate in continuous speech recognition. The exact calculation of the posterior of a word sequence implies a sum over all possible word boundaries, which is usually approximated by a maximum operation. In the approach presented here, the posterior probability of a sequence of words is decomposed into word posterior probabilities, which are calculated using forward and backward word probabilities. Up to a certain extent, this method takes into account the summation over word boundaries, which leads to consistent improvements in word error rate on a range of varying tasks.

7.1 Decompositions of the Posterior and Joint Probabilities

Let $w_1^N = w_1 \dots w_N$ denote a sequence of words, $\tau_1 \dots \tau_N$ the sequence of corresponding starting and $t_1 \dots t_N$ the sequence of corresponding ending times. A word sequence with given boundary times can then be written as: $[w;\tau,t]_1^N = [w_1;\tau_1,t_1]\dots[w_N;\tau_N,t_N]$, with the constraints $\tau_1 = 1$, $t_N = T$, and $t_{n-1} = \tau_n - 1$ for all $n = 2 \dots N$.

7.1.1 Standard Model for the Joint Probability

Presuming given word boundary times and the application of an m-gram language model, the following decomposition of the joint probability is normally used:

$$\begin{aligned} p([w;\tau,t]_1^N, x_1^T) &= p(x_1^T|[w;\tau,t]_1^N) \cdot p(w_1^N) \\ &\overset{\text{model}}{=} \prod_{n=1}^{N} p(x_{\tau_n}^{t_n}|w_n) \cdot p(w_n|w_{n-m+1}^{n-1}) \,, \end{aligned} \tag{7.1}$$

where $x_1^T = x_1 \ldots x_T$ denotes the sequence of acoustic observations. For convenience, we define sequences with descending indices to be empty, for example $w_i^j = \emptyset$ for all $j < i$. As Eq. (7.1) reflects, it is usually assumed that the acoustic probability of a word w_n with specific starting τ_n and ending times t_n depends only on the current word w_n and the acoustic observations within the word boundaries:

$$\begin{aligned} p(x_1^T|[w;\tau,t]_1^N) &= \prod_{n=1}^{N} p(x_{\tau_n}^{t_n}|x_1^{t_{n-1}},[w;\tau,t]_1^N) \\ &\overset{\text{model}}{=} \prod_{n=1}^{N} p(x_{\tau_n}^{t_n}|w_n). \end{aligned} \tag{7.2}$$

The exact search criterion for the word sequence with the highest joint probability involves a sum over all possible word boundaries, as is reflected in the following equation:

$$\begin{aligned} p(x_1^T, w_1^N) &= \sum_{\{\tau_1^N, t_1^N\}} p(x_1^T, [w;\tau,t]_1^N) \\ &\approx \max_{\{\tau_1^N, t_1^N\}} p(x_1^T, [w;\tau,t]_1^N). \end{aligned} \tag{7.3}$$

The second part of Eq. (7.3) represents the word boundary optimization. Using this approximation, the search problem could be solved efficiently by dynamic programming, see e.g. [Ney 1984]. In the following sections, possibilities to calculate the joint probability or, equivalently the posterior probability for a word sequence without word boundary optimization will be discussed.

7.1.2 Summation over Word Boundaries

In this section, the effect of the use of word boundaries on the calculation of the posterior probabilities will be discussed. In general, the posterior probability of a word sequence w_1^N *independent* of word boundaries underlies the following normalization constraint:

$$\sum_{\{w_1^N\}} p(w_1^N|x_1^T) = \sum_{\{w_1^N\}} \frac{p(x_1^T, w_1^N)}{\sum_{\{v_1^N\}} p(x_1^T, v_1^N)} = 1. \tag{7.4}$$

In order to obtain this normalization, a sum over all possible word sequences has to be performed for the calculation of the denominator in Eq. (7.4). The normalization now depends on the method of calculation of the joint probability. The summation over all possible word sequences is not feasible in most practical cases. Therefore, it could be approximated by a sum over word sequences from an N-best list or by a sum over the word sequences derived from a word graph. In this case, the normalization will also be limited to these N-best lists or word graphs respectively. In general, the calculation of the joint probability also implies a summation over all word boundaries, which is

usually replaced by a maximization, the word boundary optimization, cf. Eq. (7.3). The search space for large vocabulary continuous speech recognition comprises an extremely high number of sentence hypotheses. In order to be able to efficiently search this space in one pass, dynamic programming and beam search have to be applied. A condition for breaking down the search problem into a dynamic programming problem, the word boundary optimization becomes essential.

Without requiring any assumptions or approximations, two schemes for the calculation of the posterior probability $p(w_1^N|x_1^T)$ of a word sequence are conceivable. The first possibility would be to perform the word boundary summation individually for each word sequence that is to be hypothesized. For this approach a preselection of word sequences could be extracted from N-best lists. The computational complexity of such an approach grows exponentially with the length of the word sequences, if a given number of words has to be hypothesized per spoken word.

Another approach to the calculation of the posterior probability of word sequences with word boundary *summation* is possible, whose computational complexity is only linear in the length of the word sequence hypotheses. In this approach, the posterior probability of a word sequence is decomposed into word posterior probabilities by successive application of *Bayes*' rule [van Kampen 1992]:

$$p(w_1^N|x_1^T) = \prod_{n=1}^{N} p(w_n|w_1^{n-1}, x_1^T, N).$$

For this purpose, *position* and word history dependent word posterior probabilities $p_{N,n}(w_n|w_1^{n-1}, x_1^T)$ have to be defined, which are identified by the position index n of word w_n in a word sequence of length N. In this case, the normalization of the posterior probability of a whole word sequence translates into an individual normalization at each word position n:

$$\sum_{w} p(w|w_1^{n-1}, x_1^T, N) = 1. \tag{7.5}$$

After the introduction of word boundaries, the position dependent word posterior probabilities have to be written as follows:

$$
\begin{aligned}
p(w_n|w_1^{n-1}, x_1^T, N) &= \frac{\sum\limits_{\{w_{n+1}^N\}} p(w_1^N, x_1^T)}{\sum\limits_{\substack{v_n^N: \\ v_1^{n-1}=w_1^{n-1}}} p(v_1^N, x_1^T)} \\
&= \frac{\sum\limits_{w_{n+1}^N} \sum\limits_{\tau_1^N, t_1^N} p([w;\tau,t]_1^N, x_1^T)}{\sum\limits_{\substack{v_n^N: \\ v_1^{n-1}=w_1^{n-1}}} \sum\limits_{\tilde{\tau}_1^N, \tilde{t}_1^N} p([v;\tilde{\tau},\tilde{t}]_1^N, x_1^T)} \\
&= \frac{\sum\limits_{w_{n+1}^N} \sum\limits_{\tau_1^N, t_1^N} \prod\limits_{i=1}^{N} p(x_{\tau_i}^{t_i}|w_i) \cdot p(w_i|w_1^{i-1})}{\sum\limits_{\substack{v_n^N: \\ v_1^{n-1}=w_1^{n-1}}} \sum\limits_{\tilde{\tau}_1^N, \tilde{t}_1^N} \prod\limits_{j=1}^{N} p(x_{\tilde{\tau}_j}^{\tilde{t}_j}|v_j) \cdot p(v_j|v_1^{j-1})} \\
&\approx \frac{\sum\limits_{w_{n+1}^N} \max\limits_{\tau_1^N, t_1^N} \prod\limits_{i=1}^{N} p(x_{\tau_i}^{t_i}|w_i) \cdot p(w_i|w_1^{i-1})}{\sum\limits_{\substack{v_n^N: \\ v_1^{n-1}=w_1^{n-1}}} \max\limits_{\tilde{\tau}_1^N, \tilde{t}_1^N} \prod\limits_{j=1}^{N} p(x_{\tilde{\tau}_j}^{\tilde{t}_j}|v_j) \cdot p(v_j|v_1^{j-1})}.
\end{aligned}
\tag{7.6}
$$

The calculation of such position dependent word posterior probabilities could be performed using an extended forward-backward scheme on word level that includes the word position in a word sequence. Position dependent word posterior probabilities allow simple normalization constraints, which are independent for each word position n, cf. Eq. (7.5). However, the use of position dependent word posteriors will only be efficient, if their corresponding word history is limited.

The possibility of calculating position dependent word posterior probabilities has only been recognized in the final phase of this work, therefore no experiments have been performed yet for this approach, which allows word boundary summation without further requirements, except for the limitation of the word history. Nevertheless, the above reflections were included to give motivation for the improvements obtained by an alternative decomposition of the posterior probability of word sequences into word boundary dependent word posterior probabilities, which partly retains the summation over word boundaries.

7.1.3 Alternative Decomposition for the Posterior Probability

Beginning from the position dependent word posterior probabilities, an approximated decomposition of the posterior probability into word boundary dependent word posterior probabilities could be defined:

$$\begin{aligned} p(w_1^N|x_1^T) &= \prod_{n=1}^{N} p(w_n|w_1^{n-1}, x_1^T, N) \\ &\approx \max_{\tau_1^N, t_1^N} \prod_{n=1}^{N} p([w_n;\tau_n,t_n]|w_1^{n-1}, x_1^T) \\ &\overset{\text{model}}{=} \max_{\tau_1^N, t_1^N} \prod_{n=1}^{N} p([w_n;\tau_n,t_n]|w_{n-\nu+1}^{n-1}, x_1^T). \end{aligned} \tag{7.7}$$

Here, the introduction of word boundaries still includes the requirement that successive ending and starting times should match ($t_n + 1 = \tau_{n+1}$, cf. Section 7.1). Although the introduction of word boundaries seems similar to the standard word boundary optimization, the calculation of the word posterior probabilities still involves a summation over all word boundaries except for the word boundaries of the word to be hypothesized.
Here, the only further approximation is made by reducing the history of the word to be hypothesized to the $\nu - 1$ predecessor words. Note that the length ν of the history used here is not necessarily equal to the length m of the history used in the underlying language model.

Possibilities to calculate the posterior probabilities for single words, given a word history and the acoustic observations up to a given word boundary time will be presented in the following sections. A further advantage of the proposed approach is discussed in Section 7.3, where an approach to relax the word boundary optimization as well as the *Viterbi* approximation in favor of state summation *and* to apply dynamic programming to the search problem is proposed.

7.2 Search Using Forward and Backward Word Probabilities

In *Viterbi* approximation, the word posterior probabilities from Eq. (7.7) could be calculated using the forward and backward word probabilities defined in Section 6.2:

$$p([w;\tau,t]|w_{m-\nu+1}^{m-1}, x_1^T) = \frac{p([w;\tau,t], w_{m-\nu+1}^{m-1}, x_1^T)}{p([\,\cdot\,;\tau,t], w_{m-\nu+1}^{m-1}, x_1^T)} \tag{7.8}$$

with:

$$p([w;\tau,t],w_{m-\nu+1}^{m-1},x_1^T) = \sum_{w_2^{m-\nu}} \sum_{v_1^{m-2}} \Bigg[\frac{\Phi(w_2^{m-1},[w;\tau,t]) \cdot \Psi([w;\tau,t],v_1^{m-2})}{p(x_\tau^t|w)} \cdot \cdot \prod_{n=1}^{m-2} p(v_n|w_{n+1}^{m-1}\, w\, v_1^{n-1}) \Bigg], \tag{7.9}$$

As will be discussed later in the following Section 7.2.1, *no* normalization term $p([\ \cdot\ ;\tau,t],w_{m-\nu+1}^{m-1},x_1^T)$ has been found so far that both fulfills a reasonable normalization constraint on $p([w;\tau,t]|w_{m-\nu+1}^{m-1},x_1^T)$ for $\nu > 2$, *and* leads to improved recognition performance. Therefore, we define the following approximative normalization, which is a geometric mean over frame-wise normalization terms at all time frames in the interval $[\tau,t]$:

$$p([\,\cdot\,;\tau,t],w_{m-\nu+1}^{m-1},x_1^T) = \prod_{t'=\tau}^{t} \Bigg[\sum_{\substack{\tilde{\tau},\tilde{t}:\\ \tilde{\tau}\le t'\le\tilde{t}}} \sum_{\tilde{w}} p([\tilde{w};\tilde{\tau},\tilde{t}],w_{m-\nu+1}^{m-1},x_1^T) \Bigg]^{\frac{1}{t-\tau+1}}. \tag{7.10}$$

However, for word posterior probabilities without context ($\nu = 1$), the above normalization term reduces to the acoustic marginal distribution for the utterance, $p(x_1^T)$. It will be shown in the next section how this normalization leads to a normalization, which is valid on a frame by frame basis.

7.2.1 Normalization Properties of Word Posterior Probabilities

In the experiments, the cases $\nu = 1$ and $\nu = 2$ were investigated. Especially for the case of word posterior probabilities with single word context or more ($\nu \ge 2$), it is not clear how the normalization has to be chosen. Especially the fact that we hypothesize words with boundary times poses the question, which boundary times have to be considered for the normalization. Therefore I will first discuss the much simpler case of word posterior probabilities without context ($\nu = 1$), where the normalization term simply reduces to the acoustic marginal distribution for the utterance:

$$p([w;\tau,t]|x_1^T) = \frac{p([w;\tau,t],x_1^T)}{p(x_1^T)}.$$

The normalization in the case of $\nu = 1$ has a clear interpretation: the posterior probabilities $p([w;\tau,t]|x_1^T)$ of all words which are hypothesized at a specific point $\bar{t}$ in time with $\tau \le \bar{t} \le t$ always sum up to unity. Such a point in time can be interpreted as a cut through the word graph and it is evident that the total probability for intersecting this cut by definition equals one.

In the following, very simple illustration, we assume, without loss of generality, that no language model is used and that all acoustic probabilities are equal. Fig 7.1 shows

this simplified word graph. The edges represent word and silence hypotheses. The forward-backward algorithm computes the posterior hypothesis probabilities shown next to the word graph edges in the illustration. As can be seen, the probabilities sum up to unity for any point in time $1 \leq t \leq 15$.

It should be noted that the word posterior probabilities could also be seen as the steady-state flow through a network, if the joint probabilities of the edges of a word graph are interpreted as resistances. In this case, the flow through a node equals the sum of the flows (posterior probabilities) running into the node, as well as the sum of the flows running out of the node. This could be verified at the example given in Fig. 7.1.

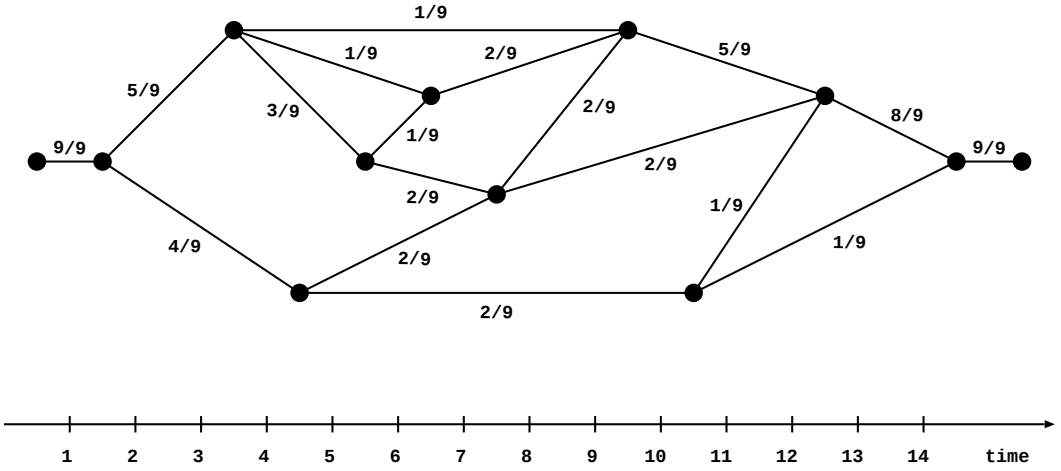

Figure 7.1: A simplified word graph for illustrational purposes. The solid edges in the graph above represent word and silence hypotheses. We assume that no language model is used and that all acoustic probabilities are equal. For each edge $[w;\tau,t]$ the posterior hypothesis probability $p([w;\tau,t]|x_1^T)$ is specified. As can be seen, these probabilities sum up to unity for any point in time.

A time-frame-wise normalization property could not be obtained for the case of word posterior probabilities with context, since the normalization will also depend on the context as well as the position in time in the word graph. For word posterior probabilities with single word context h, several simple normalization schemes are conceivable according to the boundary times of word w in $p([w;\tau,t]|h,x_1^T)$:

1. normalization according to fixed starting *and* ending time τ, t:

$$\sum_{\tilde{w}} p([w;\tau,t]|h,x_1^T) = 1 \quad \Leftrightarrow \quad p([\,\cdot\,;\tau,t],h,x_1^T) = \sum_{\tilde{w}} p([\tilde{w};\tau,t],h,x_1^T),$$

2. normalization according to fixed starting time τ:

$$\sum_{\tilde{t}} \sum_{\tilde{w}} p([w;\tau,\tilde{t}]|h,x_1^T) = 1 \quad \Leftrightarrow \quad p([\,\cdot\,;\tau,t],h,x_1^T) = \sum_{\tilde{t}} \sum_{\tilde{w}} p([\tilde{w};\tau,\tilde{t}],h,x_1^T);$$

3. normalization according to fixed ending time t:

$$\sum_{\tilde{\tau}} \sum_{\tilde{w}} p([w;\tilde{\tau},t]|h,x_1^T) = 1 \quad \Leftrightarrow \quad p([\,\cdot\,;\tau,t],h,x_1^T) = \sum_{\tilde{\tau}} \sum_{\tilde{w}} p([\tilde{w};\tilde{\tau},t],h,x_1^T);$$

4. normalization of the probability "flow" through any time t' between the starting and the ending time, e.g. the midpoint between τ and t, $t' = (\tau + t)/2$:

$$\sum_{\substack{\tilde{\tau},\tilde{t}:\\ \tilde{\tau}\leq t'\leq\tilde{t}}} \sum_{\tilde{w}} p([w;\tau,t]|h,x_1^T) = 1 \quad \Leftrightarrow \quad p([\,\cdot\,;\tau,t],h,x_1^T) = \sum_{\substack{\tilde{\tau},\tilde{t}:\\ \tilde{\tau}\leq t'\leq\tilde{t}}} \sum_{\tilde{w}} p([\tilde{w};\tilde{\tau},\tilde{t}],h,x_1^T).$$

In analogy to the case of position dependent word posterior probabilities (Section 7.1.2), the first normalization constraint of the above list would have to be chosen. Nevertheless, preliminary experiments showed that *all* the above normalization approaches proved unsuccessful for the purpose of using word posterior probabilities with single word context as recognition scores. The main reason for this is that all the above normalization constraints are too much localized to specific time frames. As a consequence, for given context h, the normalization term varies not very smoothly for successive starting or ending times. In addition, for all of these normalization constraints, no straightforward generalization to the case without context ($\nu = 1$) does exist. Especially for the first constraint in the above list, a word boundary dependent normalization would be the result for the case without context, which does not make sense. In contrast to this, the normalization of the position dependent word posterior probability without context would be *independent* of the position as it should be, cf. Eq. (7.6).
Requiring a normalization for the case with context ($\nu \geq 2$) that generalizes correctly to the case without context ($\nu = 1$) leads to the normalization chosen in Eq. (7.10), which gave the expected improvements in word error rate for recognition with word posterior probabilities with single word context.

7.2.2 Experimental Results for Search using Word Posterior Probabilities

Following this work, experiments were performed in cooperation[1]. Five different speech corpora were used for these experiments, the Dutch *Arise* corpus, the German *Verbmobil II* corpus, the English *North American Business* (NAB) '94 development corpus, and the American English *Broadcast News* (BN) '96 evaluation corpus [Wessel$^+$ 2000a]. The corpora cover a range of very different conditions including clean as well as degraded recordings, clean as well as spontaneous speech, and vocabularies ranging between approx. 1000 and 65k words. Experimental details and the baseline recognition performance are summarized in Table 7.1. Further information on the corpora could be found in Annex A.

[1]Because of the close relation between this approach and the estimation of confidence measures, these experiments were performed in cooperation with my colleague Frank Wessel.

Table 7.1: Experimental details for the five different testing corpora. WGD denotes the word graph density, NGD the node graph density, BGD the boundary graph density, GER the word graph error rate in [%] and WER the word error rate in [%]. For details on the word graph measures see [Ortmanns$^+$ 1997a].

corpus	size of vocabulary	WGD	NGD	BGD	GER	trigram perplex.	standard recognition del - ins - **WER**
Arise	985	218.8	86.0	24.4	7.4	12.6	2.1 - 3.2 - **15.8**
Verbmobil II	7128	209.2	73.1	18.3	8.7	56.1	6.1 - 6.9 - **33.6**
NAB 20k	19987	98.4	47.5	10.9	4.1	124.5	1.9 - 2.1 - **13.2**
NAB 64k	64736	87.1	43.9	10.0	1.8	145.9	2.0 - 1.5 - **11.1**
BN	65491	105.5	39.1	10.1	10.6	213.7	6.0 - 4.3 - **33.3**

Here, the calculation of the forward and backward probabilities and the search procedure have been performed using word graphs. Nevertheless, the proposed novel search approach could also be implemented as a full search procedure with an unconstrained forward pass with pruning, and a backward pass, which is embedded into the search space spanned by the forward pass.

So far, experiments have only been performed for the cases $\nu = 1$ (no consideration of word context during search) and $\nu = 2$ (single word context). It should be noted that *all* experiments reported in this section were performed using a trigram language model ($m = 3$).

7.2.2.1 Word Posterior Probabilities Without Context

As could be seen in Table 7.2, the use of word posterior probabilities *without* context already gives consistent improvements over the standard search approach. Relative improvements in word error rate range between 1.5% and 5%. Although the final search procedure does not consider context anymore, it should be kept in mind that the calculation of the word posterior probabilities still does involve a trigram language model.

Moreover, it should be noted that the highest improvements in word error rate were obtained for the more difficult corpora, the American English *Broadcast News '96* (BN), the German *Verbmobil II* corpus, and the Dutch *Arise* corpus, which all include spontaneous speech and, except for *Verbmobil II*, also degraded recording conditions.

Table 7.2: Experimental results for five different testing corpora. Standard recognition is compared to recognition with word posterior probabilities without using context and using single word context. In all cases a trigram language model has been used. For further experimental details cf. Table 7.1. WER denotes the word error rate in [%].

corpus	standard recognition del - ins - **WER**	posterior word probability recognition	
		no context ($\nu = 1$) del - ins - **WER**	word context ($\nu = 2$) del - ins - **WER**
Arise	2.1 - 3.1 - **15.8**	2.8 - 2.2 - **15.0**	1.9 - 3.1 - **15.4**
Verbmobil II	6.2 - 6.4 - **33.4**	8.2 - 4.7 - **32.7**	7.7 - 5.2 - **32.7**
NAB 20k	1.9 - 2.1 - **13.2**	2.1 - 1.9 - **13.0**	1.9 - 2.0 - **12.9**
NAB 64k	2.0 - 1.5 - **11.1**	2.3 - 1.2 - **10.8**	2.0 - 1.3 - **10.7**
BN	6.0 - 4.3 - **33.3**	6.7 - 3.5 - **32.3**	6.1 - 4.2 - **32.5**

7.2.2.2 Word Posterior Probabilities With Single Word Context

In the last column of Table 7.2, the recognition results using word posterior probabilities with single word context ($\nu = 2$) are shown. As for the case of using no context, consistent improvements over the standard search approach could be observed - improvements in word error rate range between 2% and 4%. Except for the case of the *Arise* corpus with its mostly very short sentences, the results with and without context are comparable.

For the *Arise* corpus word posterior probabilities with context still perform better than standard recognition, but they do not perform as good as word posterior probabilities without context. The explanation for this different behaviour in comparison to the other corpora lies in the low average sentence length for the *Arise* corpus, which is little more than 3, cf. Appendix A.2.3. Because of this, the number of word sequences with a given context is considerably reduced. This directly reduces the probability mass accumulated to the word posterior probabilities with context. Therefore, the effect of the summation is reduced for the *Arise* corpus.

In all cases, the results for word posterior probabilities *with* context are not significantly better than *without* using context, which supports the hypothesis that most of the positive effect of this new approach is due to the summation process. The length of the context therefore seems to be much more important for the case of the language model, which was a trigram language model in all cases presented here.

7.3 Efficient Search involving HMM-State Summation

It seems not clear how search involving exact state summation could be performed efficiently. As discussed for the problem of word boundary summation, one possibility to do this is to produce N-best lists using the *Viterbi* approximation, and then to rescore the acoustic model using state summation. In analogy, the definition of position dependent

word posterior probabilities also allows for state summation with reduced computational complexity, cf. Section 7.1.2. It should be noted that exact state summation does not involve the estimation of word boundaries anymore, which is the main reason for the difficulty to find efficient implementations for search with state summation.

The introduction of word boundaries to search with state summation enables the following efficient formulation using the forward and backward state probabilities as defined in Section 6.3:

$$p([w;\tau,t],w_{m-\nu+1}^{m-1},x_1^T) = \sum_{w_1^{m-\nu}} \sum_{v_1^{m-1}} \Bigg[\phi_{\tau-1}\left(w_1^{m-1};s(w_{m-1})\right) \cdot \psi_t\left(\sigma(v_1);v_1^{m-1}\right) \cdot p(w|h) \cdot \\ \cdot \prod_{n=1}^{m-1} p(v_n|w_{n+1}^{m-1}\, w\, v_1^{n-1}) \cdot \\ \cdot \sum_{s_\tau^t:w} p\left(s_\tau|s(w_{m-1})\right) \cdot p(x_\tau^t|s_\tau^t) \cdot p\left(\sigma(v_1)|s_t\right) \Bigg]. \tag{7.11}$$

Here, $s(w_{m-1})$ denotes the last state of word w_{m-1}, and $\sigma(v_1)$ denotes the first state of word v_1.

In addition to the advantages discussed above for the *Viterbi* approach, the advantages of the state summation approach consist of the fact that even if single *Viterbi* state paths might lead to poor probabilities, the sum over all possible paths might lead to more realistic results.

An interesting test before applying the proposed approach for dynamic programming would be, to use standard *Viterbi* search techniques to produce N-best lists, which then are rescored using full state summation. But even if this test fails to produce improvements on the standard search approach, the above proposed methods involving summation over predecessor words could lead to improvements.

Chapter 8

Discriminative Criterion Based Acoustic Modeling

In this section two novel methods using discriminative training criteria for the evaluation of acoustic models will be presented.

The first approach comprises an efficient discriminative mixture density splitting algorithm. It combines a model evaluation measure based on the Maximum Mutual Information (MMI) criterion with subsequent standard Maximum Likelihood (ML) training of the HMM parameters. The second approach presents an efficient method for discriminative training of linear feature transformations. A closed solution will be presented for an MMI based variant of *linear discriminant analysis* (LDA) [Beulen$^+$ 1995].

8.1 Mixture Density Splitting

The standard approach to acoustic modeling in speech recognition uses Hidden Markov Models (HMM) in combination with continuous mixture densities. When using mixture densities, a crucial point is the choice of the model complexity, i.e. the determination of the number of densities to be assigned to each mixture model. The usual splitting methods try to double the number of densities iteratively, as far as enough observations are assigned to a density. On the one hand, one would expect to increase the number of densities for a given mixture with the heterogeneity of the corresponding distribution of the acoustic data. On the other hand, this number is clearly limited by the amount of data available for a given task. In order to take this into account, likelihood thresholds could be used to limit the number of densities to be splitted.

8.1.1 Interdependence of Discriminative Training and Model Complexity

As could be seen in Table 3.2, the improvements obtained by discriminative training methods in comparison to conventional ML training are especially high for single Gaussian density acoustic models, i.e. for low model complexity. On the other hand,

the relative improvements obtained by discriminative training are reduced for more complex models. For the MMI criterion, this seems to be in agreement with the fact that, in the limit of an infinite amount of training data, ML and MMI both lead to the correct distributions, provided the model assumptions are correct. The comparatively good performance for low model complexity suggests that discriminative training criteria should be well suited to evaluate the ability of an acoustic model to describe the data.

8.1.2 Discriminative Splitting Choice

Taking a closer look at the decomposition of the discriminative counts for the case of *Viterbi* approximation and maximum approximation at the mixture level, (cf. Eqs. (4.7)-(4.9)):

$$\Gamma_{sl}(1) = \Gamma_{sl}^{spk}(1) - \Gamma_{sl}^{gen}(1), \tag{8.1}$$

the following interpretations are due.

For the case of the MMI criterion, the count $\Gamma_{sl}^{spk}(1)$ for the spoken word sequences gives the number how often an observation is aligned to density l and state s given the spoken word sequence. Ideally, the count $\Gamma_{sl}^{gen}(1)$ for the alternative word sequences would be nearly the same, if the posterior probability of the spoken word sequence is always considerably higher than the posterior probabilities of all other word sequences, which suggests suboptimal modeling. Correspondingly, if $\Gamma_{sl}^{gen}(1)$ is lower than $\Gamma_{sl}^{spk}(1)$, the spoken word sequence is underrepresented in the set of alternative word sequences. If $\Gamma_{sl}^{gen}(1)$ is higher than $\Gamma_{sl}^{spk}(1)$, then density l in state s even becomes contributions from more alternative word sequences than the spoken ones. Both latter cases suggest sufficiently well modeling. As stated in [Normandin 1995] for the case of the MMI criterion, this suggests that only those densities should be splitted, which have the highest values of the discriminative count $\Gamma_{sl}(1)$.

8.1.3 Alternative Derivation of Discriminative Splitting

Another heuristic derivation of model evaluation by discriminative averages might be drawn from the derivatives of the unified discriminative criterion with respect to the mixture weights c_{sl} (Eq. (4.6)),

$$\frac{\partial \mathcal{F}(\theta)}{\partial c_{sl}} = \frac{1}{c_{sl}} \cdot \Gamma_{sl}(1).$$

According to gradient descent based parameter optimization, large positive derivatives would indicate large increases in the criterion by increasing the corresponding mixture weight, considering the normalization constraint and provided the criterion is to be maximized. This could also be interpreted as the need of the corresponding density to be better modeled. If, in addition the derivative is multiplied by the corresponding mixture

weight itself, i.e. by its relative importance for a given state, we again arrive at the interpretation that the value of the discriminative count indicates the modeling ability of a density.

8.1.4 Determination of Parameters for Splitted Densities

After choosing a density for splitting, in conventional splitting the mixture weight is equally distributed upon both new densities and the mean vector is perturbed by small amounts in opposite directions. In the discriminative splitting approach, instead the density is reestimated according to ML and MMI to obtain the new pair of densities. In other words, the mean and mixture weight from ML reestimation are assigned to one density, and the mean and mixture weight from MMI reestimation are assigned to the other density, which should be a better estimation than a perturbation around the density to be splitted.

8.1.5 Hybrid MMI/ML Mixture Density Splitting Algorithm

In each splitting step, 50% of the densities are splitted according to their discriminative counts. Finally the resulting increased parameter set is trained until convergence by *Maximum Likelihood* (ML) training.

The hybrid MMI/ML splitting approach is summarized as follows, beginning from a ML trained single Gaussian density model [Schlüter$^+$ 1999a]:

1. Splitting phase:
   (a) Reestimate all parameters according to ML.
   (b) Determine the median $\widetilde{\Gamma}$ of $\Gamma_{sl}(1)$.
   (c) For each density (s,l) with $\Gamma_{sl}(1) \geq \widetilde{\Gamma}$ add a new density reestimated according to MMI.
2. Parameter optimization phase:
   (a) Reestimate all parameters according to ML.
   (b) Repeat (2a) until convergence.
   (c) Continue with the splitting phase until the desired number of densities is reached.

8.1.6 Experiments for Hybrid MMI/ML Splitting

Experiments were performed on the *SieTill* corpus. As shown in Table 8.1 the best result for conventional splitting with ML training was obtained using 64 densities per mixture, leading to a word error rate of 1.81%.

Table 8.1: Comparison of the discriminative and conventional mixture density splitting. Results on the *SieTill* corpus. In the column 'dns' the average number of densities per mixture is given.

split	criterion	dns	del-ins-sub	WER[%]	SER[%]
conventional	ML	32	198-201-450	1.97	5.31
		64	198-162-419	1.81	4.93
		128	193-169-431	1.85	4.94
	MMI	32	182-161-407	1.74	4.80
discriminative	ML	33	176-101-418	1.61	4.42
	MMI		172-103-417	1.61	4.46

The best result for conventional splitting with discriminative training was obtained using only 32 densities per mixture giving a word error rate of 1.67% (cf. Table 3.2). The best overall result of 1.61% word error rate on this task was obtained using discriminative splitting and ML training leading to, on the average, 33 densities per mixture.

For a solely discriminative splitting approach on the *TI digit string* task it was reported in [Normandin 1995] that further ML training gave an increase in error rate. Since the final models are ML trained in the first place, further experiments were performed, in order to investigate, if the corresponding results could be improved further by subsequent MMI training. This was not the case, as shown in Table 8.1. Fig. 8.1 clearly shows that the results for hybrid MMI/ML splitting are significantly better than those obtained by conventional splitting with both ML and MMI training, especially for equal parameter numbers.

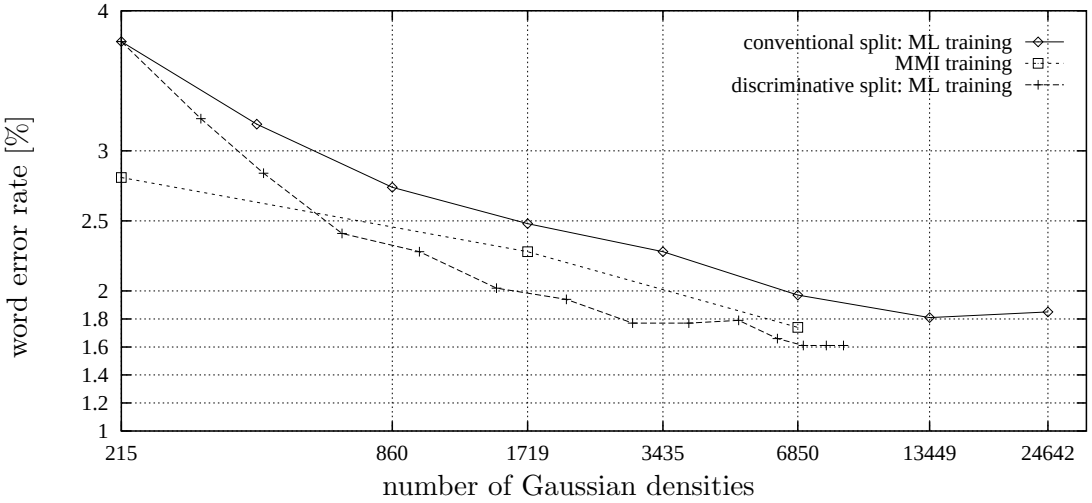

Figure 8.1: Evolution of word error rates on the *SieTill* test corpus for the proposed hybrid MMI/ML splitting approach and for conventional splitting with ML and MMI training.

8.1.6.1 Convergence

Fig. 8.2 shows a plot of the log-likelihood convergence for ML training with conventional and discriminative splitting. For equal number of parameters, the discriminative splitting approach clearly leads to lower likelihoods than the conventional splitting.

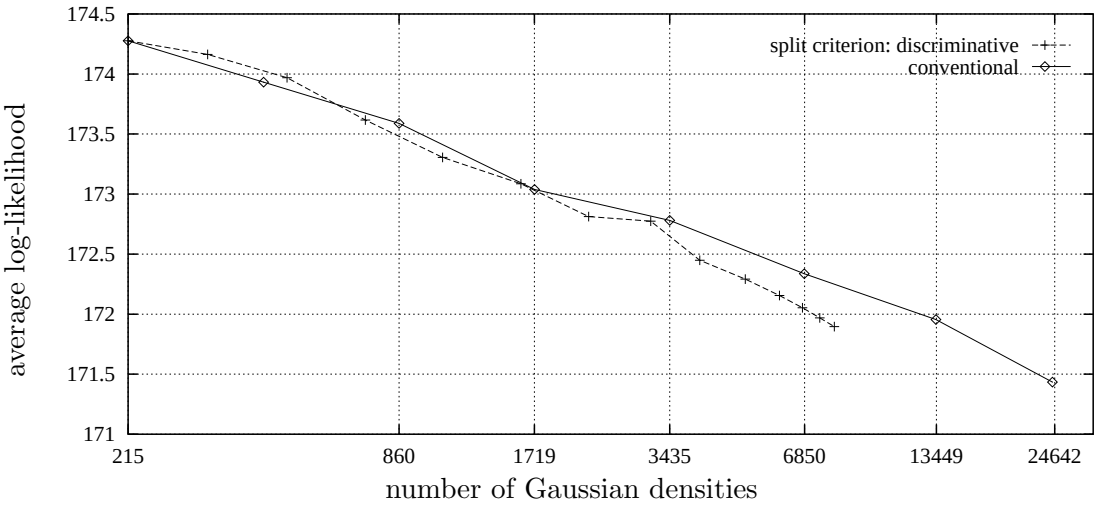

Figure 8.2: Comparison of the average log-likelihood from ML training against number of Gaussian densities for both splitting approaches considered here (female portion of the *SieTill* corpus).

8.1.6.2 Distribution of Densities

Finally, Fig. 8.3 shows the distribution of numbers of densities for each mixture of the whole word HMM including silence. The values clearly vary very much, ranging from a minimum of 1 density up to 270 densities for the silence mixture. The latter could be motivated by the high overall silence ratio of more than 55% in the *SieTill* corpus.

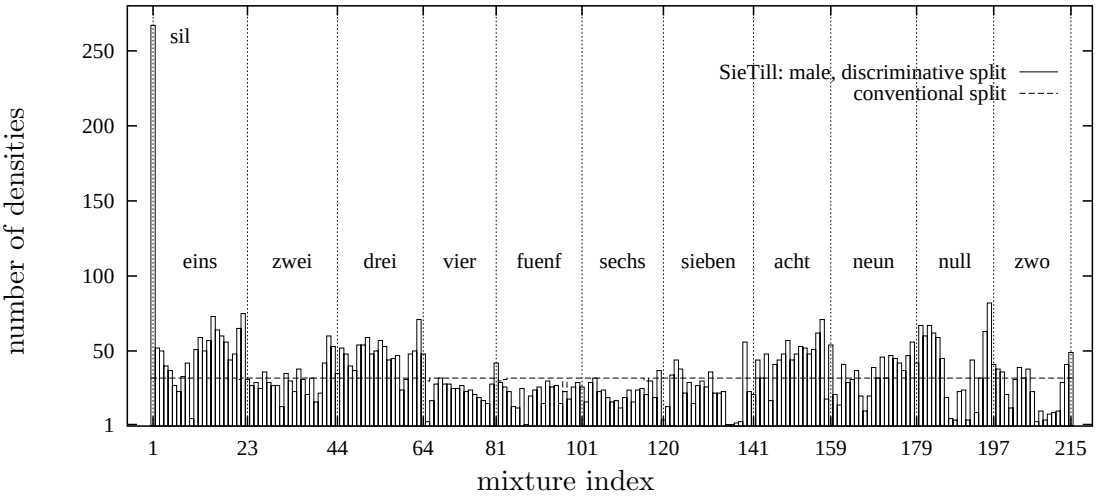

Figure 8.3: Distribution of the number of densities per mixture obtained by applying the proposed hybrid MMI/ML splitting approach to the male portion of the *SieTill* corpus.

8.2 Linear Feature Transformation

In this section a novel approach to the training of linear feature transformation is presented, the *linear MMI analysis* (LMA). The new approach makes use of the discrimination and model evaluation abilities of the MMI criterion to train linear feature transformations and choose promising features. In the style of the *linear discriminant analysis* (LDA), the approach gives a closed solution to the underlying optimization problem, which is independent of the training of the mixture density parameters.

8.2.1 Dimension Reducing Linear Feature Transformations

Consider the dimension reducing linear transformation $A \in \mathbb{R}^{D \times D'}$ of the original acoustic feature vectors, $x \in \mathbb{R}^D$, into the transformed feature vectors $y \in \mathbb{R}^{D'}$ with $D' < D$. The parameters of the emission distributions of the transformed features could be obtained from the original emission distributions, if these would be known. Here, only the original mean vectors μ_s of Gaussian single densities for the HMM states s are assumed to be known. The covariance matrices Σ of the transformed features are supposed to be pooled over all HMM states and indeterminate. It will be shown that the optimization of the linear transformation is independent of the choice of the pooled covariance matrix. Now, the emission distributions of the transformed features could be written in the following form:

$$\begin{aligned} p(y = A^T \cdot x|s) &= \mathcal{N}(A^T \cdot x|A^T \cdot \mu_s, \Sigma) \\ &= \frac{1}{\sqrt{\det(2\pi\Sigma)}} \quad e^{-\frac{1}{2}(x-\mu_s)^T \cdot A \cdot \Sigma^{-1} \cdot A^T \cdot (x-\mu_s)}. \end{aligned}$$

Covariance matrices by definition are symmetrical and could thus be decomposed into the following product of their square roots:

$$\Sigma = [\Sigma^{\frac{1}{2}}]^T \cdot \Sigma^{\frac{1}{2}}.$$

At this point, the square root of the covariance matrix could as well be integrated into the transformation matrix:

$$\overline{A} = A \cdot \Sigma^{-\frac{1}{2}}.$$

In the following, the transformation matrix will always be related to this integrated version and for simplicity, the notation for matrix A will remain the same. Now, using the identity $a^T \cdot b = \mathrm{Tr}[ab^T]$ for vectors $a, b \in \mathbb{R}^D$, the logarithm of the emission distribution could also be written using the trace operation for quadratic matrices:

$$\begin{aligned} -\log \mathcal{N}(A^T x | A^T \mu_s, \Sigma) =& \frac{1}{2}(x-\mu_s)^T A A^T (x-\mu_s) + \frac{1}{2}\log\det(2\pi\Sigma) \\ =& \frac{1}{2}\,\mathrm{Tr}\left[A^T(x-\mu_s)(x-\mu_s)^T A\right] + \mathrm{const}(A), \end{aligned} \tag{8.2}$$

where $\mathrm{const}(A)$ denotes an expression, which is independent of A.

8.2.2 Normalization

Since the discriminative criteria discussed here only involve likelihood *ratios*, the normalization of the linear transformation has to be ensured explicitly. Let the transformation matrix A be given by a set of D-dimensional vectors v_d with $d = 1, ..., D'$:

$$A = (v_1, v_2, ..., v_{D'}).$$

Then a general normalization constraint would be to require:

$$v_d^T \cdot C \cdot v_d = 1 \qquad \text{for each } d = 1, ..., D', \tag{8.3}$$

with a regular matrix $C \in \mathbb{R}^{D \times D}$.

8.2.3 Discriminative Linear Feature Transformation

The normalization constraints defined by Eq. (8.3) are included into the unified discriminative criterion (cf. Eq. (3.1)) using the *Lagrange* formalism with *Lagrangian* multipliers λ_d. Thus, the following discriminative criterion including linear feature transformation is obtained:

$$\widetilde{\mathcal{F}}(v_1, ..., v_{D'}) = \mathcal{F}(v_1, ..., v_{D'}) - \frac{1}{2}\sum_{d=1}^{D'} \lambda_d [v_d^T C v_d - 1].$$

For clarity, the only arguments of the criteria are given by the transformation parameters v_d. Using Eq. (8.2) and the state specific versions of Eq. (4.6) and the decomposition of the discriminative averages in Eq. (4.7), the derivative of the above criterion with respect to the transformation parameters gives:

$$\begin{aligned}\frac{\partial\widetilde{\mathcal{F}}}{\partial v_d} &= -\Big\{\alpha\sum_s \Gamma_s\big([x-\mu_s][x-\mu_s]^T\big)+\lambda_d C\Big\}v_d \\ &= -\alpha\sum_s \Gamma_s^{spk}\big([x-\mu_s][x-\mu_s]^T\big)v_d+\alpha\sum_s \Gamma_s^{gen}\big([x-\mu_s][x-\mu_s]^T\big)v_d-\lambda_d C v_d \\ &= -Wv_d+Tv_d-\lambda_d C v_d,\end{aligned} \tag{8.4}$$

with the definition of the discriminative within-class and total-scatter matrices, W and T respectively:

$$\begin{aligned}W &= \alpha\sum_s \Gamma_s^{spk}\big([x-\mu_s][x-\mu_s]^T\big) \\ T &= \alpha\sum_s \Gamma_s^{gen}\big([x-\mu_s][x-\mu_s]^T\big).\end{aligned} \tag{8.5}$$

It should be noted that for the case of the MMI criterion with *Viterbi* approximation, the above within-class-scatter matrix is identical with the within-class-scatter matrix defined for the LDA. From the definition of the averages on the sets of alternative word sequences (cf. Eq. (4.9)) it follows that the contribution of each state dependent local scatter term, $[x_{rt}-\mu_s][x_{rt}-\mu_s]^T$, to the total-scatter matrix is weighted by the generalized FB probability of the corresponding state, $\gamma_{rt}(s)$.

8.2.4 Limitations and Smoothing

In the case of the MMI criterion, often it could be observed that $\gamma_{rt}(s)$ is nearly 1 for the corresponding state of the *Viterbi* alignment of the spoken word sequence, and nearly zero otherwise. Considering this, it becomes clear that large parts of the summation for T are *within*-class contributions only. Hence, large parts of the difference between the total- and the within-class-scatter matrix cancel out, leaving only contributions from those parts of the training utterances, which were incorrectly modeled. In this way, the difference $T-W$ could be understood as a discriminative between-class-scatter matrix. As will become clear from the experiments, the qualitative reduction of the between-class-scatter to badly modeled training data makes necessary smoothing of the discriminative total-scatter matrix T. Here, for smoothing the scatter from the global pooled mean vector μ is chosen. Therefore the discriminative total-scatter matrix is replaced by the following smoothed version:

$$\widetilde{T}(\eta)=\eta T+(1-\eta)\alpha\sum_s \Gamma_s^{gen}(1)\cdot[\mu-\mu_s][\mu-\mu_s]^T,$$

with the smoothing parameter $\eta \in [0,1]$. Setting the derivative from Eq. (8.4) to zero and using the smoothed total-scatter matrix, the following equation for v_d is obtained:

$$(\widetilde{T}(\eta) - W)v_d = \lambda_d C v_d. \tag{8.6}$$

which is a generalized eigenvalue problem except for the fact that the total-scatter matrix, through the generalized FB probabilities, still depends on the vectors v_d of the transformation. In order to be able to find a closed solution, therefore it is assumed that T is independent of the transformation, i.e. the generalized FB probabilities are calculated using the untransformed features and acoustic models.

8.2.5 Linear MMI Analysis (LMA)

Now, the derivatives of the unified criterion are reintegrated with respect to the transformation parameters while keeping the approximations made. Setting the integration constants such that the resulting criterion still complies with the normalization constraint, the following training criterion for the linear transformation is obtained:

$$\mathcal{F}_{\text{LMA}}(v_1, ..., v_{D'}) = \sum_{d=1}^{D'} \Big\{ v_d^T \big[\widetilde{T}(\eta) - W\big] v_d - \lambda_d \big[v_d^T C v_d - 1\big] \Big\}. \tag{8.7}$$

For the solution of generalized eigenvalue problems, numerical algorithms do exist, e.g. [LAPACK]. Substituting Eq. (8.6) into Eq. (8.7), the following criterion function is obtained:

$$\mathcal{F}_{\text{LMA}}(v_1, ..., v_{D'}) = \sum_{d=1}^{D'} \lambda_d.$$

Hence, similar to the LDA, the optimal transformation matrix A is obtained by solving the generalized eigenvalue problem of Eq. (8.6) and composing A of those D' generalized eigenvectors v_d, whose corresponding eigenvalues λ_d are maximal. It should be noted that Eq. (8.6) strongly depends on the choice of the normalization, i.e. on the choice of matrix C. Here, $C = W$ is chosen, which leads to a generalized version of the LDA problem, which will be called *linear MMI analysis* (LMA):

$$\widetilde{T}(\eta) v_d = (\lambda_d + 1) W v_d.$$

8.2.6 Experiments using LMA

8.2.6.1 Determination of the Smoothing Parameter

Before using LMA for evaluation, the smoothing parameter η had to be optimized, which was performed by a line search on the *SieTill* training corpus. In Fig. 8.4 recognition

results on the *SieTill* training corpus are plotted for several values of η in the range between 0 and 1.

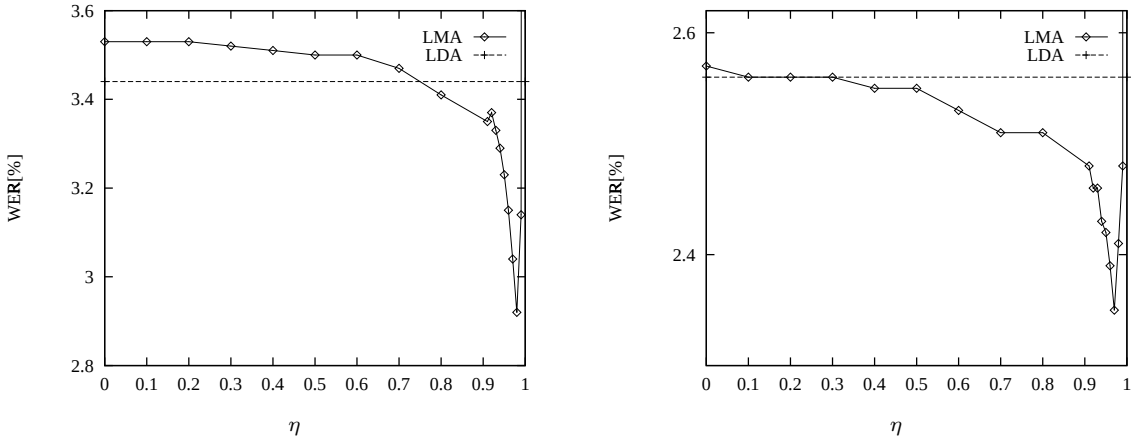

Figure 8.4: Preliminary optimization of the smoothing parameter η for the LMA. Recognition results on the *SieTill* training corpus.

For LMA without smoothing, i.e. $\eta = 1$, the error rates obtained on the training corpus were about a factor of 3 higher than those using LDA. This result clearly is a consequence of the fact that the unsmoothed LMA criterion only captures those parts of the training data, which are incorrectly modeled. In other words, due to necessary approximations, the discriminative between-class-scatter matrix could not reflect the effects of the transformation on subsequent recognition on the training corpus. Nevertheless, as shown in Fig. 8.4, the optimal smoothing parameter was very near to 1, which is reflected in the minima and the following sharp increases in word error rate.

8.2.6.2 Comparison of LMA and LDA

Using the optimal smoothing parameter found on the training corpus, evaluation of LMA was performed on the *SieTill* corpus. Table 8.2 summarizes recognition results for LMA and LDA with varying numbers of densities per Gaussian mixture.

Table 8.2: Comparison of feature transformation using *linear discriminant analysis* (LDA) and linear MMI analysis (LMA). Recognition results on the *SieTill* corpus.

method	dns	del - ins - sub	WER[%]	SER[%]
LDA	1	304-270-1053	3.78	9.74
	2	234-318- 824	3.19	8.31
	4	198-329- 652	2.74	7.18
	8	198-298- 579	2.48	6.47
	16	191-294- 499	2.28	5.92
LMA	1	320-214- 985	3.53	9.18
	2	236-298- 754	2.99	7.87
	4	212-343- 622	2.73	7.15
	8	218-317- 531	2.48	6.48
	16	197-304- 464	2.24	5.83

Although for single densities a relative improvement in word error rate of nearly 7% could be observed for LMA in comparison to LDA, the prominence of LMA diminishes with increasing parameter numbers. For 16 densities per mixture, LMA and LDA nearly produced the same recognition results.

In order to obtain improvements for mixture densities, the LMA approach could straight forwardly be extended to include mixture densities into the definition of the LMA criterion.

Chapter 9

Scientific Contributions

The aim of this work was to enhance small and large vocabulary speech recognition by development and implementation of efficient discriminative training methods. Starting from a class of promising criteria, several discriminative training methods have been presented and evaluated:

Unified Discriminative Criterion: A class of discriminative training criteria including the *Maximum Mutual Information* (MMI), the *Minimum Classification Error* (MCE), the *Corrective Training* (CT) and *Falsifying Training* (FT) approximations criteria have been merged into a unified discriminative criterion. Experimental evaluation of the MMI, MCE, CT and FT criterion for recognition of continuous digit strings on the *SieTill* corpus showed significant improvements in word error rate, especially for low acoustic model complexity. On this task, the MCE criterion performed best for low model complexity, giving a word error rate of 2.6% for single densities, i.e. a relative reduction of nearly 1/3 compared to ML training. On the other hand, for optimal model complexity FT showed the best results, with a word error rate of 1.67% for 32 densities per mixture which is a relative reduction of about 8% compared to the best ML result with 64 densities per mixture. All in all, the error rate based methods consistently gave better results than MMI and CT. Depending on the difficulty of the task, CT sooner or later becomes ineffective for increasing model complexity.

Asymptotical Behaviour of MCE and related Criteria: Proofs for the following properties for the MCE criterion and the *generalized Gini* criterion have been given. The MCE criterion represents an upper bound to the true (optimal) *Bayes*' error rate independent of the underlying model distribution. Model-free optimization of the MCE criterion with sufficient training data leads to a closed form solution. The corresponding model distribution leads to the true *Bayes*' error rate. A new discriminative training criterion similar to the MCE criterion, the *generalized Gini* (GG) criterion is defined. The GG criterion gives a closer upper bound to the *Bayes*' error rate than the MCE criterion. Model-free optimization of the GG criterion with infinite training data leads to the same optimal model distribution than the MCE criterion.

Uniform Parameter Optimization: The Parameter optimization techniques for discriminative criteria, the extended *Baum* (EB) algorithm and the gradient descent method (GD) were analytically and experimentally shown to be approximately equivalent. Both methods are suitable to optimize any specific criterion represented by the unified approach.

Optimal Language Model Choice: The algorithms presented are generally suited to include m-gram language models. Experiments on the WSJ0 task with a recognition vocabulary of 5k words showed significant differences for discriminative training using language models with context lengths between zero and three for training. On the task in question, a unigram language model emerged as optimal choice for MMI *training*, whereas best *recognition* results currently are obtained using a trigram language model. Using the unigram for MMI training, a word error rate of 4% was obtained for recognition with a trigram language model. This is a relative improvement of 11% compared to the best result for ML training with trigram recognition.

On the other hand, using a particular language model for *training*, several experiments with varying language models for *recognition* showed no correlation between the choice of language model for training and recognition. The experiments were performed for MMI training with a bigram language model on the WSJ0 corpus. Subsequent recognition on the test corpora with bigram, trigram, phrase-bigram and phrase-trigram consistently gave relative improvements in word error rate between 5 and 6% compared to the corresponding ML results.

Discriminative Training and Search: For iterative extraction of alternative word hypotheses from word graphs an efficient constrained recognition approach has been developed that reduces the recognition time by a factor of approximately 5 compared to a full recognition pass.

Based on calculation techniques developed for discriminative training, a novel scoring approach for dynamic programming-based search has been proposed that makes use of word posterior probabilities calculated on word graphs. In comparison to standard scoring, the new approach takes advantage of summation over word boundaries. On a range of speech corpora, improvements in word error rate of up to 5% were obtained. The approach allows to take into account HMM-state summation as opposed to the *Viterbi* approximation.

Discriminative Acoustic Modeling: An efficient hybrid MMI/ML algorithm for mixture density splitting has been proposed. Here the discriminative criterion is used to train the structure of the acoustic model, whereas parameter optimization is performed by standard ML training. The approach lead to significant improvements in word error rate while producing significantly smaller acoustic models. By definition, the approach is much more efficient than full discriminative training of the model parameters and could

not be improved by subsequent discriminative training of the model parameters. The word error rate obtained on the *SieTill* corpus was 1.61%, which was better than both the results for ML and MMI training with conventional splitting.

A linear feature transformation algorithm based on the MMI criterion, the *linear MMI analysis* (LMA) has been developed, for which a closed solution has been found. First experiments on the *SieTill* corpus show consistent improvements for LMA in comparison to LDA using acoustic models of reduced complexity.

Chapter 10

Outlook

Several developments and investigations are conceivable so as to further improve discriminative training, especially for large vocabulary speech recognition. According to the experiments presented in this work, MCE training should be the discriminative training method of choice. Although the efficiency of discriminative training methods has been largely improved, discriminative training times for large vocabulary speech recognition applications still are too high to really allow for extensive optimization of empirical parameters. MCE training especially involves the choice of suitable values for the smoothing function, which has considerable effect on the performance. In order to avoid very time consuming empirical optimizations, an automatic means for the determination of the appropriate smoothing parameter for MCE would be desirable.

The main reason for the comparatively very long training times of discriminative training is the necessity to find the sets of alternative word sequences so as to build up the competing model. Already, methods have been proposed successfully, which alleviate this problem by defining word-independent discriminative models. Similar performance as compared to the MMI approach has been obtained, with significantly reduced computational complexity [Bahl$^+$ 1996, Povey & Woodland 1999]. Here, similar approximations to the MCE criterion would be desirable, since MCE was found to perform better, than MMI.

Improvements obtained by discriminative training criteria in comparison to ML training significantly decrease, while improving the acoustic modeling. On the other hand, discriminative modeling approaches like mixture density splitting show that discriminative training still bears a great potential to improve acoustic models. Therefore approaches are imaginable that use discriminative criteria for other modeling aspects like phonetic modeling based on classification and regression trees, or speaker adaptation. Discriminative training might especially well be suited for speaker adaptive training, because of the very good performance on the training corpora.

Starting from the initial success of LMA, investigations on the proposed extension of LMA to mixture densities might be promising. In addition, the transformation matrix could be made model dependent, where suitable parameter tying schemes would have to be investigated.

Due to the success of the novel scoring approach presented in this work, further investigations are due on the effect of the summation over the word boundaries. Especially the approach using position dependent posterior probabilities should be pursued, since it allows to sum over *all* word boundaries.

Appendix A

Speech Corpora and Recognition Systems

This annex summarizes information on the different speech corpora and baseline systems for small and large vocabulary speech recognition used in this work.

A.1 Continuous Digit Strings

A.1.1 Digit Corpora

The experiments for continuous digit recognition reported in this work have been performed on the *TI digit string* corpus [Leonhard 1984] for continuously spoken American English digits recorded under clean conditions from adult speakers, and on the *SieTill* corpus [Eisele$^+$ 1996] for continuously spoken German digits recorded over the telephone line from adult speakers. The vocabulary of both corpora comprises the ten digits in the corresponding language plus one pronunciation variant, 'oh' as variant for 'zero' in the American English case, and 'zwo' as variant for 'zwei' in the German case. Statistics on both corpora are summarized in Table A.1.

Table A.1: Corpus statistics: Digit corpora.

corpus		female		male	
		strings	digits	strings	digits
TI	test	4389	14424	4311	14159
	train	4388	14414	4235	13915
SieTill	test	6176	20205	6938	22881
	train	6113	20115	6835	22463

A.1.2 Continuous Digit Recognition Systems

The recognition systems for both corpora are based on a one-pass decoder design. For the purpose of discriminative training the recognizer has been extended to produce word

graphs. Details on acoustic modeling are summarized in the following.

Acoustic modeling: *TI digit string* corpus.

- American English digits;
- 11 whole word HMMs incl. 'oh';
- per gender 357 states plus 1 for silence;
- Gaussian single densities;
- state-specific diagonal covariances;
- 16 mel-cepstral coefficients; plus first and second derivatives.

Acoustic modeling: *SieTill* corpus.

- telephone line recorded German digits;
- 11 whole word HMMs incl. 'zwo';
- per gender 214 states plus 1 for silence;
- HMM segments with 2 identical emission distributions;
- Gaussian mixture densities;
- pooled or state-specific diagonal covariances;
- 12 mel-cepstral coefficients plus first derivatives plus second derivative of the energy;
- LDA on 3 adjacent input frames (including derivatives: $3 \times 25 = 75$ input features), which are reduced to 25 output features.

A.2 Spontaneous Speech

Verbmobil is a project for automatic speech-to-speech translation of spontaneous speech. *Verbmobil* assists dialogs in the domain of appointment negotiation (phase I), travel planning and hotel reservation (phase II) for the languages German, English and Japanese. The *Verbmobil* project is a joint initiative of information technology companies, universities, and research centers, and is funded by the German Federal Ministry for Education, Science, Research and Technology (BMBF). Details on the *Verbmobil* project could be found on the [Verbmobil homepage]. One of the contributions of the *Lehrstuhl für Informatik VI* at *RWTH Aachen* to the *Verbmobil* project is a real-time continuous speech recognition system for German spontaneous speech. A detailed description of the system could be found in [Kanthak$^+$ 2000, Sixtus$^+$ 2000].

A.2.1 *Verbmobil I*

The *Verbmobil I* training corpus consists of German human-to-human dialogs recorded in clean conditions, making up a total of 11168 sentences and approx. 28 hours of speech respectively. The *Verbmobil I* 1996 evaluation corpus consists of 343 sentences, 6446 running words, of which 116 are unknown, and approx. 40 minutes of speech respectively. The perplexity of a phrase-trigram is 35 on the *Verbmobil I* 1996 evaluation corpus.

Recognition system: *Verbmobil I.*

- vocabulary:
  - 5328 words plus 35 pronounciation variants plus 29 spellings,
  - 4 hesitations, 14 noises (not evaluated),
  - 100 phrases/multiwords;
- phrase/multiword trigram language model (PP 35);
- 2001 decision tree based triphone states plus one silence state;
- mixtures with a total of 108k Gaussian densities;
- one pooled diagonal covariance;
- 16 mel-cepstral coefficients plus first derivatives plus second derivative of the energy.
- LDA on 3 adjacent input frames (including derivatives: $3 \times 33 = 99$ input features), which are reduced to 33 output features.

A.2.2 *Verbmobil II*

The *Verbmobil II* training corpus consists of German human-to-human dialogs recorded in clean conditions, making up a total of 23736 sentences and approx. 44 hours of speech respectively. The *Verbmobil II* 1998 evaluation corpus consists of 763 sentences, 8863 running words, of which 278 are unknown, and approx. 62 minutes of speech respectively. The perplexity of a trigram is 56 on the *Verbmobil II* 1998 evaluation corpus.

Recognition system: *Verbmobil II.*

- vocabulary:
  - 7128 words plus 35 pronounciation variants plus 29 spellings,
  - 4 hesitations, 14 noises (not evaluated),
- trigram language model (PP 35);
- 2500 decision tree based triphone states plus one silence state;
- mixtures with a total of 146k Gaussian densities;

- one pooled diagonal covariance;
- 16 mel-cepstral coefficients plus first derivatives plus second derivative of the energy.
- LDA on 3 adjacent input frames (including derivatives: $3 \times 33 = 99$ input features), which are reduced to 33 output features.

A.2.3 *Arise*

The *Arise* corpus consists of Dutch human-to-machine dialogs recorded over the telephone line in the context of a train information system. The training corpus contains a total of 22768 sentences and approx. 16 hours of speech respectively. The *Arise* evaluation corpus consists of 2136 sentences, 6889 running words, of which 460 are unknown, and approx. 94 minutes of speech respectively. The perplexity of a trigram is 13 on the *Arise* evaluation corpus.

Recognition system: *Arise*.

- vocabulary: 985 words
- trigram language model (PP 13);
- 1000 decision tree based triphone states plus one silence state;
- mixtures with a total of 63k Gaussian densities;
- one pooled diagonal covariance;
- 12 mel-cepstral coefficients plus first derivatives plus second derivative of the energy.
- LDA on 3 adjacent input frames (including derivatives: $3 \times 25 = 75$ input features), which are reduced to 25 output features.

A.2.4 *Broadcast News* Transcription

The *Broadcast News* '96 corpus consists of American English transcribed television and radio broadcasts. The recordings cover a range of six different conditions from clean to heavily degraded speech [Garofolo$^+$ 1996, Pitz$^+$ 1999].

The training corpus contains a total of 26167 sentences and approx. 96 hours of speech respectively. The *Broadcast News* 1996 evaluation corpus consists of 405 sentences, 20284 running words, of which 374 are unknown, and approx. 106 minutes of speech respectively. The perplexity of a phrase-trigram is 214 on the *Broadcast News* 1996 evaluation corpus.

Recognition system: *Broadcast News*.

- vocabulary: 65491 words plus 9603 pronounciation variants, including 900 phrases
- phrase-trigram language model (PP 214);

- 2000 decision tree based triphone states plus one silence state;
- mixtures with a total of 318k Gaussian densities;
- one pooled diagonal covariance;
- 16 mel-cepstral coefficients plus first derivatives plus second derivative of the energy;
- LDA on 9 adjacent input frames (*without* derivatives: $9 \times 16 = 144$ input features), which are reduced to 45 output features.

A.3 Read Speech

In this work, American English read speech was investigated on the Wall Street Journal (WSJ) Corpora. The WSJ corpora are composed of business journal texts, which are read by American journalists [Pallett$^+$ 1993, Pallett$^+$ 1995, Kubala 1995] and recorded under clean conditions. The WSJ data has been collected by the *National Institute of Standards and Technology* (NIST) under the *Advanced Research Projects Agency (ARPA) Human Technology Research Program.*

A.3.1 *Wall Street Journal* (WSJ) 5k

The WSJ0 training corpus consists of 84 speakers, 7237 sentences, and approx. 15 hours of speech respectively. For evaluation, the Nov. '92 evaluation and development corpora were used which consist of 330+410 sentences, 5353+6784 running words and approx. 40+46 minutes of speech respectively and the vocabulary is closed.

Recognition system: WSJ 5k.

- vocabulary:
  - 4987 words plus 668 pronounciation variants,
  - discriminative training: 10776 words;
- 2000 decision tree based triphone states plus one silence state;
- mixtures with a total of 96k Gaussian densities;
- one pooled diagonal covariance;
- 16 mel-cepstral coefficients plus first derivatives plus second derivative of the energy;
- LDA on 3 adjacent input frames (including derivatives: $3 \times 33 = 99$ input features), which are reduced to 33 output features.

A.3.2 *North American Business* (NAB) corpus

The November '94 NAB training corpus consists of the 84 speakers of the WSJ0 corpus (see above) plus the 200 additional speakers of the WSJ1 corpus, leading to a total of approx. 81 hours of speech. Recognition systems are available for vocabularies of 20k and 64k words, for which evaluation both is performed on the NAB November '94 *H1 development test corpus*. The development corpus is composed of 310 sentences from 20 speakers, 7387 running words, and approx. 49 minutes of speech respectively. For the NAB 20k vocabulary, 199 words are unknown, and the trigram perplexity is 125. For the NAB 64k vocabulary, 39 words are unknown, and the trigram perplexity is 146.

Recognition system: Nov. '94 NAB development corpus.

- vocabularies:
  - 19978 words plus 2434 pronounciation variants (NAB 20k),
  - 64736 words plus 5234 pronounciation variants (NAB 64k);
- 3000 decision tree based triphone states plus one silence state;
- mixtures with a total of 269k Gaussian densities;
- one pooled diagonal covariance;
- 16 mel-cepstral coefficients plus first derivatives plus second derivative of the energy;
- LDA on 3 adjacent input frames (including derivatives: $3 \times 33 = 99$ input features), which are reduced to 33 output features.

Appendix B

Symbols and Acronyms

B.1 Mathematical Symbols

θ	set of all parameters of the acoustic model
$\mathbf{f}$	smoothing function
$\mathbf{f}'$	derivative of the smoothing function
α	weighting exponent
ϱ	slope of a sigmoidal smoothing function
$\mathbf{r}$	index of a speech utterance
$\mathbf{t}$	time frame index
$\mathbf{T_r}$	number of time frames of utterance r
$\mathbf{X_r}$	sequence of acoustic observation vectors $x_{r1}, \ldots, x_{rT_r}$ of utterance r
$\mathbf{D}$	dimension of the acoustic observation vectors
$\mathbf{N_r}$	number of spoken words of utterance r
$\mathbf{W_r}$	sequence of spoken words $w_{r1}, \ldots, w_{rN_r}$ of utterance r
$\mathbf{p}_\theta(\mathbf{X_r}\|\mathbf{W_r})$	acoustic emission probability density for utterance r given the spoken word sequence W_r
$\mathbf{p}_\theta(\mathbf{W_r}\|\mathbf{X_r})$	*a-posteriori* probability for the spoken word sequence of utterance r
$\mathbf{p}(\mathbf{W_r})$	language model probability of the spoken word sequence of utterance r
$\mathbf{w}, \mathbf{v}$	word indices
$\mathbf{W}, \mathbf{V}$	word sequences of any length

$\mathbf{h}$	history of a word: a sequence of $m-1$ words $h_1, \ldots, h_{m-1}$ preceding a given word
$\mathbf{f}$	future of a word: a sequence of $m-1$ words $f_1, \ldots, f_{m-1}$ following a given word
$\mathcal{M}_\mathbf{r}$	set of alternative word sequences of utterance r
$\mathcal{F}$	unified discriminative criterion
$\mathcal{F}_\mathbf{MMI}$	*Maximum Mutual Information* (MMI) criterion
$\mathcal{F}_\mathbf{MCE}$	*Minimum Classification Error* (MCE) criterion
$\mathcal{F}_\mathbf{CT}$	*Corrective Training* (CT) criterion
$\mathcal{F}_\mathbf{FT}$	*Falsifying Training* (FT) criterion
$\mathcal{F}_\mathbf{GG}$	*Generalized Gini* (GG) criterion
$\delta(\mathbf{i},\mathbf{j})$	Kronecker delta, equals 1 for $i=j$, and 0 otherwise
$\mathbf{s}$	state of a *Hidden Markov Model* (HMM)
$\mathbf{l}$	density index of a mixture density
$\mathbf{c_{sl}}$	mixture weight for density l in state s
$\mu_\mathbf{sl}$	mean vector parameter of a single Gaussian density l in state s
$\Sigma_\mathbf{sl}$	covariance matrix of a single Gaussian density l in state s
$\sigma^\mathbf{2}_\mathbf{sl}$	variance vector of a single Gaussian density l in state s (diagonal covariance)
$\theta_\mathbf{sl}$	all parameters $\{c_{sl}, \mu_{sl}, \Sigma_{sl}\}$ of a single Gaussian probability density
$\mathbf{p}(\mathbf{x_{rt}}\|\mu_\mathbf{sl}, \Sigma_\mathbf{sl})$	single Gaussian emission probability density
$\mathbf{p}_\theta(\mathbf{x_{rt}}\|\mathbf{s})$	mixture Gaussian emission probability density conditioned by state s
$\mathbf{s_t}(\mathbf{X_r}, \mathbf{W})$	state of the optimal *Viterbi* alignment path at time t for utterance r given a word sequence W
$\mathbf{l}(\mathbf{x}, \mathbf{s})$	index of the mixture density component that maximizes the emission probability for state s given the acoustic observation x
$\gamma_\mathbf{rt}(\mathbf{s}; \mathbf{W})$	*forward-backward* (FB) probability to observe state s at time t for utterance r given a word sequence W
$\gamma_\mathbf{rt}(\mathbf{s})$	generalized *forward-backward* (FB) probability to observe state s at time t for utterance r given all alternative word sequences

$\mathbf{g}(\mathbf{x})$	any (usually polynomial) function of the acoustic observations
$\Gamma_{\mathbf{sl}}\big(\mathbf{g}(\mathbf{x})\big)$	discriminative averages over function g of the acoustic observations with respect to density l in state s
$\Gamma_{\mathbf{sl}}^{\mathbf{spk}}\big(\mathbf{g}(\mathbf{x})\big)$	averages over function g of the acoustic observations with respect to density l in state s for the spoken word sequences
$\Gamma_{\mathbf{sl}}^{\mathbf{gen}}\big(\mathbf{g}(\mathbf{x})\big)$	averages over function g of the acoustic observations with respect to density l in state s for all alternative word sequences
$\mathcal{S}(\theta, \hat{\theta})$	auxiliary function of the extended *Baum* (EB) algorithm
$\mathbf{D}, \mathbf{D_s}$	globally pooled or state specific iteration constants of the EB algorithm
$\Delta\mu_{\mathbf{sl}}$	step size for gradient descent (GD) optimization of mean parameter μ_{sl}
$\Delta\sigma_{\mathbf{s}}^{\pm\mathbf{2}}$	step size for gradient descent (GD) optimization of variance parameter σ_{sl}^2
$\Delta\mathbf{c_{sl}}$	step size for gradient descent (GD) optimization of mixture weight parameter c_{sl}
$\mathbf{p}_\theta(\mathbf{s_1^{T_r}}, \mathbf{x_{r1}^{T_r}})$	joint probability for the acoustic observations of utterance r and a state sequence $s_1^{T_r}$
$\mathbf{a_{rt}}(\mathbf{s}\|\mathbf{W_r})$	forward probability to observe state s at time t given the acoustic observations of utterance r up to time t and the spoken word sequence W_r
$\mathbf{b_{rt}}(\mathbf{s}\|\mathbf{W_r})$	backward probability to observe state s at time t given the acoustic observations from time t to the end of utterance r and the spoken word sequence W_r
$[\mathbf{w}; \tau, \mathbf{t}]$	edge of a word graph given by a word index w, a starting time τ and an ending time t
$\mathbf{q}([\mathbf{w}; \tau, \mathbf{t}], \mathbf{X_r})$	joint probability to observe word w with word boundaries τ, t for utterance r with acoustic observations X_r
$\Phi_{\mathbf{r}}(\mathbf{h_1^{m-2}}, [\mathbf{w}; \tau, \mathbf{t}])$	forward probability to observe word w with word boundaries τ, t and predecessor words h_1^{m-2} for utterance r with the acoustic observations up to time t
$\Psi_{\mathbf{r}}([\mathbf{w}; \tau, \mathbf{t}], \mathbf{f_2^{m-1}})$	forward probability to observe word w with word boundaries τ, t and successor words f_2^{m-1} for utterance r with the acoustic observations from time τ up to the end of the utterance
$\mathbf{g_t}(\mathbf{s}; \mathbf{X_r})$	joint probability to observe state s at time t with acoustic observations X_r given all alternative word sequences

$\phi_{\mathbf{rt}}(\mathbf{h};\mathbf{s})$	forward probability to observe state s at time t in word h_{m-1} with predecessor words h_1^{m-2} for utterance r with the acoustic observations up to time t
$\psi_{\mathbf{rt}}(\mathbf{s};\mathbf{f})$	backward probability to observe state s at time t in word f_1 with successor words f_2^{m-1} for utterance r with the acoustic observations from time t up to the end of the utterance
$\mathbf{p}([\mathbf{w};\tau,\mathbf{t}]\|\mathbf{h}_{\mathbf{m-\nu+1}}^{\mathbf{m-1}},\mathbf{x}_{\mathbf{1}}^{\mathbf{T}})$	posterior probability to observe word w with word boundaries τ, t given the $\nu-1$ predecessor words $h_{m-\nu+1}^{m-1}$ and the acoustic observations x_1^T
$\mathcal{N}(\mathbf{x}\|\mu,\Sigma)$	single Gaussian density with mean μ and covariance matrix Σ
$\mathbf{A}$	linear acoustic feature transformation matrix
$\mathbf{D'}$	output dimension of LDA/LMA
$\mathbf{v_1},\ldots,\mathbf{v_{D'}}$	columns of matrix A
$\mathbf{C}$	matrix for normalization constraint on LMA
$\lambda_{\mathbf{d}}$	Lagrangian multipliers, with $d = 1,\ldots,D'$
η	smoothing parameter of LMA
$\mathbf{W}$	within-class scatter matrix (only Section 8.2)
$\mathbf{T}$	total scatter matrix (only Section 8.2)
$\mathcal{F}_{\mathbf{LMA}}$	LMA-criterion for MMI-based estimation of linear feature transformations

B.2 Acronyms

ML	***M**aximum **L**ikelihood*
MMI	***M**aximum **M**utual **I**nformation*
MCE	***M**inimum **C**lassification **E**rror*
CT	***C**orrective **T**raining*
FT	***F**alsifying **T**raining*
GG	**g**eneralized ***G**ini*
GD	**g**radient **d**escent
EB	***e**xtended **B**aum* algorithm
BW	**B**aum **W**elch
FB	***F**orward-**B**ackward*
EM	**E**xpectation **M**aximization
HMM	**H**idden ***M**arkov* **M**odel
LDA	***l**inear **d**iscriminant **a**nalysis*
LMA	***l**inear **M**MI **a**nalysis*
MFCC	*m*el *f*requency *c*epstral *c*oefficients
PLP	*p*erceptual *l*inear *p*rediction
LM	**l**anguage **m**odel
PP	language model **p**er**p**lexity
WER	**w**ord **e**rror **r**ate
SER	**s**entence **e**rror **r**ate
del	**del**etion errors
ins	**ins**ertion errors
sub	**sub**stitution errors
dev	**dev**elopment test corpus
eval	**eval**uation test corpus
RTF	**r**eal **t**ime **f**actor
WSJ	**W**all **S**treet **J**ournal - a speech corpus provided by the ARPA
ARPA	**A**dvanced **R**esearch **P**rojects **A**gency

Appendix C

Detailed Calculations

C.1 Derivation of Minimal Iteration Constant Ensuring Positive Variance

Substituting the EB reestimation Eq. (4.11) for the means into the EB reestimation Eqs. (4.12a) and (4.12b) for the pooled and state specific variances respectively, we obtain:

$$\hat{\sigma}^2 = \frac{\Gamma(x^2) + \sum_s D(\sigma^2 + \sum_l c_{sl}\mu_{sl}^2) - \sum_{s,l} \frac{(\Gamma_{sl}(x) + Dc_{sl}\mu_{sl})^2}{\Gamma_{sl}(1) + Dc_{sl}}}{\Gamma(1) + \sum_s D}, \tag{C.1a}$$

$$\hat{\sigma}_s^2 = \frac{\Gamma_s(x^2) + D_s(\sigma_s^2 + \sum_l c_{sl}\mu_{sl}^2) - \sum_l \frac{(\Gamma_{sl}(x) + D_s c_{sl}\mu_{sl})^2}{\Gamma_{sl}(1) + D_s c_{sl}}}{\Gamma_s(1) + D_s}. \tag{C.1b}$$

Consider the following polynomial division for the last term in the nominators of Eqs. (C.1a) and (C.1b):

$$\frac{(\Gamma_{sl}(x) + D_s c_{sl}\mu_{sl})^2}{\Gamma_{sl}(1) + D_s c_{sl}} = c_{sl}\mu_{sl}^2 D_s + 2\Gamma_{sl}(x)\mu_{sl} - \Gamma_{sl}(1)\mu_{sl}^2 + \frac{(\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl})^2}{\Gamma_{sl}(1) + D_s c_{sl}}. \tag{C.2}$$

Substituting Eq. (C.2) into Eqs. (C.1a) and (C.1b), and considering Ineq. (4.24) for the nominator of the last term in Eq. (C.2), the claim of positive variances:

$$\hat{\sigma}^2 > \sigma_{\min} > 0 \tag{C.3a}$$

$$\hat{\sigma}_s^2 > \sigma_{\min} > 0 \tag{C.3b}$$

become linear in the iteration constants. Therefore, the inequations could be solved with respect to the iteration constants:

$$D > D_{\min} = \frac{1}{\sum_s \sigma^2 - \sigma_{\min}} \cdot \Big[-\Gamma(x^2) + \sigma_{\min}\Gamma(1) + \sum_{s,l} \left[2\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right] \mu_{sl} + \beta_s \sum_{s,l} \left[\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]^2 \Big], \tag{C.4a}$$

$$D_s > D_{s,\min} = \frac{1}{\sigma_s^2 - \sigma_{\min}} \cdot \Big[-\Gamma_s(x^2) + \sigma_{\min}\Gamma_s(1) + \sum_{l} \left[2\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right] \mu_{sl} + \beta_s \sum_{l} \left[\Gamma_{sl}(x) - \Gamma_{sl}(1)\mu_{sl}\right]^2 \Big]. \tag{C.4b}$$

C.2 Derivative of the Acoustic Emission HMM

In the following, a detailed derivation of the derivative of the acoustic emission distribution $p(x_1^T)$ using a first order HMM with respect to a class dependent parameter θ_s is given:

$$
\begin{aligned}
\frac{\partial \log p(x_1^T)}{\partial \theta_s} &= \frac{1}{p(x_1^T)} \frac{\partial}{\partial \theta_s} \sum_{\{s_1,\ldots,s_T\}} p(x_1^T, s_1^T) && \text{chain rule and HMM} \\
&= \frac{1}{p(x_1^T)} \frac{\partial}{\partial \theta_s} \sum_{\{s_1,\ldots,s_T\}} \prod_{\tau=1}^{T} p(x_\tau, s_\tau | x_1^{\tau-1}, s_1^{\tau-1}) && \text{Bayes} \\
&\approx \frac{1}{p(x_1^T)} \frac{\partial}{\partial \theta_s} \sum_{\{s_1,\ldots,s_T\}} \prod_{\tau=1}^{T} p(x_\tau|s_\tau)\, p(s_\tau|s_{\tau-1}) && \text{first order HMM} \\
&= \frac{1}{p(x_1^T)} \sum_{\{s_1,\ldots,s_T\}} \sum_{t=1}^{T} \left[\frac{\partial}{\partial p(x_t|s)} \prod_{\tau=1}^{T} p(x_\tau|s_\tau)\, p(s_\tau|s_{\tau-1}) \right] \frac{\partial p(x_t|s)}{\partial \theta_s} \\
& && \text{product rule} \\
&= \frac{1}{p(x_1^T)} \sum_{t=1}^{T} \frac{\partial p(x_t|s)}{\partial \theta_s} && \text{partial differentiation} \\
&\quad \cdot \sum_{\{s_1,\ldots,s_{t-1},s_{t+1},\ldots,s_T\}} \delta_{s,s_t} \left[p(s_t|s_{t-1}) \prod_{\tau\neq t,\tau=1}^{T} p(x_\tau|s_\tau)\, p(s_\tau|s_{\tau-1}) \right] \\
&= \frac{1}{p(x_1^T)} \sum_{t=1}^{T} \frac{1}{p(x_t|s)} \frac{\partial p(x_t|s)}{\partial \theta_s} && \text{rearrangement} \\
&\quad \cdot \sum_{s_t=s,\{s_1,\ldots,s_{t-1},s_{t+1},\ldots,s_T\}} \left[\prod_{\tau=1}^{T} p(x_\tau|s_\tau)\, p(s_\tau|s_{\tau-1}) \right] \\
&= \sum_{t=1}^{T} \frac{\sum_{\{s_1,\ldots,s_{t-1},s_{t+1},\ldots,s_T\}} p(x_1^T, s_1^T, s_t = s)}{p(x_1^T)} \frac{\partial \log p(x_t|s)}{\partial \theta_s} \\
&= \sum_{t=1}^{T} \frac{p(x_1^T, s_t = s)}{p(x_1^T)} \frac{\partial \log p(x_t|s)}{\partial \theta_s} \\
&= \sum_{t=1}^{T} p(s_t = s|x_1^T) \frac{\partial \log p(x_t|s)}{\partial \theta_s}.
\end{aligned}
$$

Bibliography

[Alleva+ 1996] P. Alleva, X. D. Huang, M.-Y. Hwang. "Improvements on the Pronunciation Prefix Tree Search Organization," Proc. *1996 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 133-136, Atlanta, GA, May 1996.

[Aubert & Ney 1995] X. Aubert, H. Ney. "Large Vocabulary Continuous Speech Recognition Using Word Graphs," Proc. *1995 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 49-52, Detroit, MI, May 1995.

[Ayer+ 1993] C. M. Ayer, M. J. Hunt, D. M. Brookes. "A Discriminatively Derived Transform for Improved Speech Recognition," Proc. *1993 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 583-586, Berlin, September 1993.

[Bahl+ 1983] L. R. Bahl, F. Jelinek, R. L. Mercer. "A Maximum Likelihood Approach to Continuous Speech Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 5, pp. 179-190, March 1983.

[Bahl+ 1986] L. R. Bahl, P. F. Brown, P. V. de Souza, R. L. Mercer. "Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition," Proc. *1986 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 49-52, Tokyo, May 1986.

[Bahl+ 1996] L. R. Bahl, M. Padmanabhan, D. Nahamoo, P. S. Gopalakrishnan. "Discriminative Training of Gaussian Mixture Models for Large Vocabulary Speech Recognition Systems," Proc. *1996 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 613-616, Atlanta, GA, May 1996.

[Bahl & Padmanabhan 1998] L. R. Bahl, M. Padmanabhan. "A Discriminant Measure for Model Complexity Adaptation," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 453-456, Seattle, WA, May 1998.

[Baker 1975a] J. K. Baker. "Stochastic Modeling for Automatic Speech Understanding," in D. R. Reddy (ed.), *Speech Recognition*, Academic Press, New York, pp. 512-542, 1975.

[Baker 1975b] J. K. Baker. "The Dragon System – An Overview," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 23, No. 1, pp. 24-29, February 1975.

[Bakis 1976] R. Bakis. "Continuous Speech Word Recognition via Centisecond Acoustic States," Proc. ASA Meeting, Washington, DC, April 1976.

[Bauer 1998] J. Bauer. "Application of Discriminative Methods for Isolated Word Recognition," *Classification, Data Analysis, and Data Highways*, I. Balderjahn, R. Mathar, M. Schader (eds.), pp. 287-294, Springer Verlag, Berlin - Heidelberg, 1998.

[Baum & Eagon 1967] L. E. Baum, J. A. Eagon. "An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology," Bulletin of the American Mathematical Society, Vol. 73, pp. 360-363, 1967.

[Baum 1972] L. Baum. "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov processes," Inequalities, Vol. 3, pp. 1-8, 1972.

[Ben-Bassat 1982] M. Ben-Bassat. "Use of Distance Measures, Information Measures and Error Bounds in Feature Evaluation," in P. R. Krishnaiah, L. N. Kanal (eds.): *Handbook of Statistics*, Vol. 2, pp. 773-791, North Holland Publishing Company, Amsterdam, 1982.

[Bengio$^+$ 1992] Y. Bengio, R. De Mori, G. Flammia, R. Kompe. "Global Optimization of a Neural Network-Hidden Markov Model Hybrid," IEEE Transactions on Neural Networks, Vol. 3, No. 2, pp. 252-259, March 1992.

[Bellman 1957] R. E. Bellman. *Dynamic Programming*, Princeton University Press, Princeton, NJ, 1957.

[Beulen$^+$ 1995] K. Beulen, L. Welling, H. Ney. "Experiments with Linear Feature Extraction in Speech Recognition," *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 2, pp. 1415-1418, Madrid, September 1995.

[Beulen$^+$ 1996] K. Beulen, E. Bransch, M. Kramer, H. Ney. "State-Tying für kontextabhängige Phonemmodelle," Sprachkommunikation: Vorträge der ITG-Fachtagung, Frankfurt am Main, September 1996, pp. 51-54, in: A. Lacroix (ed.), *ITG-Fachbericht; 139*, Springer, Berlin, 1996.

[Beulen$^+$ 1997] K. Beulen, E. Bransch, H. Ney. "State-Tying for Context Dependent Phoneme Models," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 3, pp. 1179-1182, Rhodes, Greece, September 1997.

[Beulen & Ney 1998] K. Beulen, H. Ney. "Automatic Question Generation for Decision Tree Based State Tying," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 805-808, Seattle, WA, May 1998.

[Biem & Katagiri 1997] A. Biem, S. Katagiri. "Cepstrum-Based Filter-Bank Design using Discriminative Feature Extraction Training at Various Levels," Proc. *1997 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 1503-1506, Munich, April 1997.

[Blomberg$^+$ 1984] M. Blomberg, R. Carlson, K. Elenius, B. Granström. "Auditory Models in Isolated Word Recognition," Proc. *1984 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 17.9.1-17.9.4, San Diego, CA, March 1984.

[Bocchieri 1993] E. Bocchieri. "Vector Quantization for the Efficient Computation of Continuous Density Likelihoods," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 692-695, Minneapolis, MN, April 1993.

[Breiman 1984] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone. *Classification and Regression Trees*, The Wadsworth Statistics/Probability Series, Belmont, CA, 1984.

[Bridle 1990] J. S. Bridle. "Alpha-Nets: A Recurrent 'Neural' Network Architecture with Hidden Markov Model Interpretation," Speech Communication, Vol. 9, No. 1, pp. 83-92, February 1990.

[Brown 1987] P. F. Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition,* Ph.D. thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 1987.

[Brown$^+$ 1992] P. Brown, V. Della Pietra, P. deSouza, J. Lai, R. Mercer. "Class–Based n–gram Models of Natural Language," Computational Linguistics, Vol. 18, No. 4, pp. 467-479, 1992.

[Cardin$^+$ 1993] R. Cardin, Y. Normandin, E. Millien. "Inter-Word Coarticulation Modeling and MMIE Training for Improved Connected Digit Recognition," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 243-246, Minneapolis, MN, April 1993.

[Chen$^+$ 1999] S. S. Chen, E. M. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky, P. Olsen. "Recent Improvements to IBM's SPeech Recognition System for Automatic Transcription of Broadcast News," Proc. *DARPA 1999 Broadcast News Workshop*, pp. 89-93, Herndon, VA, February/March 1999.

[Chou$^+$ 1992] W. Chou, B.-H. Juang, C.-H. Lee. "Segmental GPD Training of HMM Based Speech Recognizer," Proc. *1992 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 473-476, San Francisco, CA, March 1992.

[Chou$^+$ 1993] W. Chou, C.-H. Lee, B.-H. Juang. "Minimum Error Rate Training based on N-Best String Models," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 652-655, Minneapolis, MN, April 1993.

[Chou$^+$ 1994] W. Chou, C.-H. Lee, B.-H. Juang. "Minimum Error Rate Training of Inter-Word Context Dependent Acoustic Model Units in Speech Recognition," Proc. *1994 Int. Conf. on Speech and Language Processing*, Vol. 2, pp. 439-442, Yokohama, September 1994.

[Chow 1990] Y.-L. Chow. "Maximum Mutual Information Estimation of HMM Parameters for Continuous Speech Recognition using the N-best Algorithm," Proc. *1990 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 701-704, Albuquerque, NM, April 1990.

[Cohen 1989] J. R. Cohen. "Application of an Auditory Model to Speech Recognition," Journal Acoustical Society of America, Vol. 85, No. 6, pp. 2623-2629, June 1989.

[Dahmen+ 1999] J. Dahmen, R. Schlüter, H. Ney. "Discriminative Training of Gaussian Mixtures for Image Object Recognition," Proc. *21. DAGM Symposium Mustererkennung*, W. Frstner, J. Buhmann, A. Faber, P. Faber (eds.), pp. 205-212, Bonn, September 1999.

[Davis & Mermelstein 1980] S. B. Davis, P. Mermelstein. "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28, No. 4, pp. 357-366, August 1980.

[Dempster+ 1977] A. P. Dempster, N. M. Laird, D. B. Rubin. "Maximum Likelihood from Incomplete Data via the EM Algorithm," Journal Roy. Stat. Soc., Vol. 39, No. 1, pp. 1-38, 1977.

[Devijver & Kittler 1982] P. A. Devijver, J. Kittler. *Pattern Recognition: A Statistical Approach,* Prentice-Hall, London, 1982.

[Duda & Hart 1973] R. O. Duda, P. E. Hart. *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1973.

[Dugast+ 1995] C. Dugast, P. Beyerlein, R. Haeb-Umbach. "Application of Clustering Techniques to Mixture Density Modelling for Continuous Speech Recognition," Proc. *1995 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 524-527, Detroit, MI, May 1995.

[Eisele+ 1996] T. Eisele, R. Haeb-Umbach, D. Langmann, "A comparative study of linear feature transformation techniques for automatic speech recognition," Proc. *1996 Int. Conf. on Spoken Language Processing*, Vol. I, pp. 252-255, Philadelphia, PA, October 1996.

[Fritsch 1997] J. Fritsch. "ACID/HNN: A Framework for Hierarchical Connectionist Acoustic Modeling," in: S. Furui, B.-H. Juang, W. Chou (eds.), Proc. *1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 164-171, Santa Barbara, CA, December 1997.

[Gao+ 1992] Y. Gao, T. Huang, S. Chen, J.-P- Haton. "Auditory Model Based Speech Recognition," Proc. *1992 Int. Conf. on Speech and Language Processing*, Vol. 1, pp. 73-76, Banff, Alberta, Canada, October 1992.

[Garofolo+ 1996] J. S. Garofolo, J. G. Fiscus, W. M. Fisher. "Design and Preparation of the 1996 HUB-4 Broadcast News Benchmark Test Corpora," Proc. *Speech Recognition Workshop*, pp. 15-21, San Francisco, CA, February 1997.

[Generet+ 1995] M. Generet, H. Ney, F. Wessel. "Extensions to Absolute Discounting for Language Modeling," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 2, pp. 1245-1248, Madrid, September 1995.

[Ghitza 1986] O. Ghitza. "Auditory Nerve Representation as a Front-End for Speech Recognition in a Noisy Environment," Computer Speech and Language, Vol. 1, pp. 109-130, 1986.

[Gopalakrishnan+ 1991] P. S. Gopalakrishnan, D. Kanevsky, A. Nádas, D. Nahamoo. "An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems," IEEE Transactions on Information Theory, Vol. 37, No. 1, pp. 107-113, January 1991.

[Haeb-Umbach & Ney 1992] R. Haeb-Umbach, H. Ney. "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition," Proc. *1992 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 13-16, San Francisco, CA, March 1992.

[Haeb-Umbach & Ney 1994] R. Haeb-Umbach, H. Ney. "Improvements in Beam Search for 10000-Word Continuous-Speech Recognition," IEEE Transactions on Speech and Audio Processing, Vol. 2, No. 2, April 1994.

[Hermansky 1990] H. Hermansky. "Perceptual Linear Predictive (PLP) Analysis of Speech," Journal Acoustical Society of America, Vol. 87, No. 4, pp. 1738-1752, April 1989.

[Huang & Jack 1989] X. D. Huang, M. A. Jack. "Semi-Continuous Hidden Markov Models for Speech Signals," Computer Speech and Language, Vol. 3, No. 3, pp. 329-252, 1989.

[Huang+ 1990] X. D. Huang, K. F. Lee, H. W. Hon. "On Semi-Continuous Hidden Markov Modeling," Proc. *1990 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 689-692, Albuquerque, NM, April 1990.

[Hunt & Lefèbvre 1987] M. Hunt, C. Lefèbvre. "Speech Recognition using an Auditory Model with Pitch-Synchronous Analysis," Proc. *1987 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 20.5.1-20.5.4, Dallas, TX, April 1987.

[Hunt & Lefèbvre 1989] M. Hunt, C. Lefèbvre. "A Comparison of Several Acoustic Representations for Speech Recognition with Degraded and Undegraded Speech," Proc. *1989 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 262-265, Glasgow, May 1989.

[Hwang+ 1992] M. Y. Hwang, X. D. Huang, F. Alleva. "Predicting Unseen Triphones with Senones," Technischer Report, No. 510.7808 C28R 93-139 2, Carnegie Mellon University, 1993.

[Jardino 1996] M. Jardino. "Multilingual Stochastic n-Gram Class Language Models," Proc. *1996 Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 161-163, Atlanta, GA, May 1996.

[Jelinek 1969] F. Jelinek. "A Fast Sequential Decoding Algorithm Using a Stack," IBM Journal of Research and Development, Vol. 13, pp. 675-685, November 1969.

[Jelinek 1976] F. Jelinek. "Continuous Speech Recognition by Statistical Methods," Proc. ot the IEEE, Vol. 64, No. 10, pp. 532-556, April 1976.

[Jelinek 1991] F. Jelinek. "Self-Organized Language Modeling for Speech Recognition," pp. 450-506, in: A. Waibel, K.-F. Lee (eds.), *Readings in Speech Recognition*, Morgan Kaufmann Publ., San Mateo, CA, 1991.

[Juang & Katagiri 1992] B.-H. Juang, S. Katagiri. "Discriminative Learning for Minimum Error Classification," IEEE Transactions on Signal Processing, Vol. 40, No. 12, pp. 3043-3054, December 1992.

[Kanevsky 1995] D. Kanevsky. "A Generalization of the Baum Algorithm to Functions on Non-Linear Manifolds," Proc. *1995 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 473-476, Detroit, MI, May 1995.

[Kanthak+ 2000] S. Kanthak, A. Sixtus, S. Molau, R. Schlüter, H. Ney. "The RWTH Large Vocabulary Speech Recognition System for Spontaneous Speech," submitted to *Konvens 2000 / Sprachkommunikation*, Ilmenau, Germany, October 2000.

[Kapadia+ 1993] S. Kapadia, V. Valtchev, S. J. Young. "MMI Training for Continuous Phoneme Recognition on the TIMIT Database," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 491-494, Minneapolis, MN, April 1993.

[Katz 1987] S. M. Katz. "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 35, pp. 400-401, March 1987.

[Klakow 1998] D. Klakow. "Language-Model Optimization by Mapping of Corpora," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 701-704, Seattle, WA, May 1998.

[Kneser & Ney 1993] R. Kneser, H. Ney. "Improved Clustering Techniques for Class–Based Statistical Language Modelling," Proc. *1993 Europ. Conf. on Speech Communication and Technology*, pp. 973–976, Berlin, 1993.

[Kubala 1995] F. Kubala. "Design of the 1994 CSR Benchmark Tests," Proc. *ARPA Human Language Technology Workshop*, pp. 41-46, Austin, TX, January 1995.

[Kuhn & de Mori 1990] R. Kuhn, R. de Mori. "A Cache–Based Natural Language Model for Speech Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12, pp. 570–583, June 1990.

[LAPACK] Linear Algebra PACKage (LAPACK), http://www.netlib.org/lapack/, February 1992.

[Leonhard 1984] R. G. Leonard. "A database for speaker-independent digit recognition," *1984 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 42.11.1-42.11.4, San Diego, CA, March 1984.

[Levinson+ 1983] S. E. Levinson, L. R. Rabiner, M. M. Sondhi. "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition," Bell System Techn. Journal, Vol. 62, No. 4, pp. 1035-1074, April 1983.

[Liporace 1982] L. Liporace. "Maximum Likelihood Estimation for Multi-Variate Observations of Markov Sources," IEEE Transactions on Information Theory, Vol. 28, No. 5, pp. 729-734, 1982.

[Lowerre 1976] B. Lowerre. *A Comparative Performance Analysis of Speech Understanding Systems,* Ph.D. thesis, Carnegie-Mellon University, Pittsburgh, PA, 1976.

[Martin$^+$ 1997] S. C. Martin, J. Liermann, H. Ney. "Adaptive Topic-Dependent Language Modeling Using Word-Based Varigrams," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 3, pp. 1447-1450, Rhodes, Greece, September 1997.

[Martin$^+$ 1998] S. C. Martin, J. Liermann, H. Ney. "Algorithms for Bigram and Trigram Word Clustering," Speech Communication, Vol. 24, No. 1, pp. 19–37, 1998.

[Martin$^+$ 1999] S. Martin, C. Hamacher, J. Liermann, F. Wessel, H. Ney, "Assessment of Smoothing Methods and Complex Stochastic Language Modeling," Proc. *1999 Europ. Conf. on Speech Communication and Technology*, pp. 1939-1942, Budapest, Hungary, September 1999.

[McDermott & Katagiri 1997] E. McDermott, S. Katagiri. "String-Level MCE for Continuous Phoneme Recognition," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 123-126, Rhodes, Greece, September 1997.

[Merialdo 1988] B. Merialdo. "Phonetic Recognition Using Hidden Markov Models and Maximum Mutual Information," Proc. *1988 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 111-114, New York, April 1988.

[Nádas$^+$ 1988] A. Nádas, D. Nahamoo, M. A. Picheny. "On a Model-Robust Training Method for Speech Recognition," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 36, No. 11, pp. 1432-1436, September 1988.

[Nene & Nayar 1996] S. A. Nene, S. K. Nayar. "Closest Point Search in High Dimensions," Proc. *IEEE Conf. on Computer Vision and Pattern Recognition*, San Francisco, CA, pp. 859-865, June 1996.

[Ney 1984] H. Ney. "The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition," IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-32, No. 2, pp. 263-271, April 1984.

[Ney$^+$ 1987] H. Ney, D. Mergel, A. Noll, A. Paeseler. "A Data-Driven Organization of the Dynamic Programming Beam Search for Continuous Speech Recognition," Proc. *1987 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 833-836, Dallas, TX, April 1987.

[Ney & Noll 1988] H. Ney, A. Noll. "Phoneme Modeling using Continuous Mixture Densities," Proc. *1988 Int. Conf. on Acoustics, Speech and Signal Processing*, New York, April 1988.

[Ney 1990] H. Ney. "Acoustic Modeling of Phoneme Units for Continuous Speech Recognition," *Fifth Europ. Signal Processing Conf.*, Barcelona, Spain, September 1990, in L. Torres, E. Masgrau, M. A. Lagunas (eds.): *Signal Processing V: Theories and Applications*, pp. 65-72, Elsevier Science Publishers B. V., 1990.

[Ney+ 1992] H. Ney, R. Haeb-Umbach, B.-H. Tran, M. Oerder. "Improvements in Beam Search for 10000-Word Continuous Speech Recognition," Proc. *1992 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 9-12, San Francisco, CA, March 1992.

[Ney 1993] H. Ney. "Search Strategies for Large-Vocabulary Continuous-Speech Recognition," NATO Advanced Studies Institute, Bubion, Spain, June/July 1993, pp. 210-225, in: A. J. Rubio Ayuso, J. M. Lopez Soler (eds.), *Speech Recognition and Coding – New Advances and Trends*, Springer, Berlin, 1995.

[Ney & Aubert 1994] H. Ney, X. Aubert. "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition ," *1994 Int. Conf. on Spoken Language Processing*, Vol. 3, pp. 1355-1358, Yokohama, September 1994.

[Ney+ 1994] H. Ney, U. Essen, R. Kneser. "On Structuring Probabilistic Dependencies in Language Modeling," Computer Speech and Language, Vol. 2, No. 8, pp. 1-38, 1994.

[Ney+ 1997] H. Ney, S. C. Martin, F. Wessel. "Statistical Language Modeling using Leaving-One-Out," in S. Young, G. Bloothooft (eds.), *Corpus Based Methods in Language and Speech Processing*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1997.

[Normandin 1991] Y. Normandin. *Hidden Markov Models, Maximum Mutual Information Estimation, and the Speech Recognition Problem,* Ph.D. thesis, Department of Electrical Engineering, McGill University, Montreal, 1991.

[Normandin & Morgera 1991] "An Improved MMIE Training Algorithm for Speaker-Independent, Small Vocabulary, Continuous Speech Recognition," Proc. *1991 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 537-540, Toronto, May 1991.

[Normandin+ 1994a] Y. Normandin, R. Cardin, R. De Mori. "High–Performance Connected Digit Recognition Using Maximum Mutual Information Estimation," IEEE Transactions on Speech and Audio Processing, Vol. 2, No. 2, pp. 299-311, April 1994.

[Normandin+ 1994b] Y. Normandin, R. Lacouture, R. Cardin. "MMIE Training for Large Vocabulary Continuous Speech Recognition," Proc. *1994 Int. Conf. on Spoken Language Processing*, Vol. 3, pp. 1367-1370, Yokohama, September 1994.

[Normandin 1995] Y. Normandin. "Optimal Splitting of HMM Gaussian Mixture Components with MMIE Training," Proc. *1995 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 449-452, Detroit, MI, May 1995.

[Normandin 1996] Y. Normandin. "Maximum Mutual Information Estimation of Hidden Markov Models," *Automatic Speech and Speaker Recognition*, C.-H. Lee, F. K. Soong, K. K. Paliwal (eds.), pp. 57-81, Kluwer Academic Publishers, Norwell, MA, 1996.

[Normandin 1999] Y. Normandin, Locus Dialogue, Montreal, Canada. Private Communication, January 1999.

[Oerder & Ney 1993] H. Ney, M. Oerder. "Word Graphs: An Efficient Interface Between Continuous Speech Recognition and Language Understanding," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 119-122, Minneapolis, MN, April 1993.

[Odell$^+$ 1994] J. J. Odell, V. Valtchev, P. C. Woodland, S. J. Young. "A One-Pass Decoder Design for Large Vocabulary Recognition," Proc. *ARPA Spoken Language Technology Workshop*, Plainsboro, NJ, pp. 405-410, March 1994.

[Ohshima & Stern 1994] Y. Ohshima, R. M. Stern. "Environmental Robustness in Automatic Speech Recognition using Physiologically-Motivated Signal Processing," *1994 Int. Conf. on Spoken Language Processing*, Vol. 3, pp. 1347-1350, Yokohama, September 1994.

[Ortmanns & Ney 1995] S. Ortmanns, H. Ney. "An Experimental Study of the Search Space for 20000-Word Speech Recognition," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 2, pp. 901-904, Madrid, September 1995.

[Ortmanns$^+$ 1996a] S. Ortmanns, H. Ney, A. Eiden. "Language-Model Look-Ahead for Large Vocabulary Speech Recognition," Proc. *1996 Int. Conf. on Spoken Language Processing*, pp. 2095-2098, Philadelphia, PA, October 1996.

[Ortmanns$^+$ 1996b] S. Ortmanns, H. Ney, A. Eiden, N. Coenen. "Look-Ahead Techniques for Improved Beam Search," Proc. *CRIM-FORWISS Workshop*, pp. 10-22, Montreal, October 1996.

[Ortmanns$^+$ 1997a] S. Ortmanns, H. Ney, X. Aubert. "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition," Computer Speech and Language, Vol. 11, No. 1, pp. 43-72, January 1997.

[Ortmanns$^+$ 1997b] S. Ortmanns, L. Welling, K. Beulen, F. Wessel, H. Ney. "Architecture and Search Organization for Large Vocabulary Continuous Speech Recognition," in: M. Jarke, K. Pasedach, K. Pohl (eds.), *Informatik 97: Informatik als Innovationsmotor*, pp. 456-465, Springer, Berlin, 1997.

[Ortmanns$^+$ 1997c] S. Ortmanns, H. Ney, T. Firzlaff. "Fast Likelihood Computation Methods for Continuous Mixture Densities in Large Vocabulary Speech Recognition," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 139-142, Rhodes, Greece, September 1997.

[Ortmanns 1998] S. Ortmanns. *Effiziente Suchverfahren zur Erkennung kontinuierlich gesprochener Sprache*, Ph.D. thesis, RWTH Aachen, Becker-Kuns Verlag, Aachen, November 1998.

[Paliwal+ 1995] K. K. Paliwal, M. Bacchiani, Y. Sagisaka. "Minimum Classification Error Training Algorithm for Feature Extractor and Pattern Classifier in Speech Recognition," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 541-544, Madrid, September 1995.

[Pallett+ 1993] D. S. Pallett, J. G. Fiscus, W. M. Fisher, J. S. Garofolo. "Benchmark Tests for the DARPA spoken language program," Proc. *ARPA Human Language Technology Workshop*, pp. 7-18, Princeton, NJ, March 1993.

[Pallett+ 1995] D. S. Pallett, J. G. Fiscus, W. M. Fisher, J. S. Garofolo, B. A. Lund, M. A. Przybocki. "1994 Benchmark Test for the ARPA spoken language program," Proc. *ARPA Human Language Technology Workshop*, pp. 5-36, Austin, TX, January 1995.

[Paul 1991] D. B. Paul. "Algorithms for an Optimal A^* Search and Linearizing the Search in the Stack Decoder," Proc. *1991 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 693-696, Toronto, May 1991.

[Picone 1993] J. Picone. "Signal Modeling Techniques in Speech Recognition," Proc. of the IEEE, Vol. 81, No. 9, pp. 1215-1247, September 1993.

[Pitz+ 1999] M. Pitz, S. Molau, R. Schlüter, H. Ney. "Automatic Transcription Verification of Broadcast News and Similar Speech Corpora," Proc. *1999 DARPA Broadcast News Workshop*, pp. 157-159, Herndon, VA, February/March 1999.

[Povey & Woodland 1999] D. Povey, P. C. Woodland. "Frame Discrimination Training of HMMs for Large Vocabulary Speech Recognition," Proc. *1999 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 333-336, Phoenix, AZ, May 1999.

[Rabiner & Juang 1986] L. Rabiner, B.-H. Juang. "An Introduction to Hidden Markov Models," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 3, No. 1, pp. 4-16, 1986.

[Rabiner 1989] L. R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proc. of the IEEE, Vol. 77, No. 2, pp. 257-286, February 1989.

[Rabiner & Juang 1993] L. R. Rabiner, B.-H. Juang. *Fundamentals of Speech Recognition*, Prentice Hall PTR, Englewood Cliffs, NJ, 1993.

[Rahim & Lee 1996] M. G. Rahim, C.-H. Lee. "Simultaneous ANN Feature and HMM Recognizer Design using String-Based Minimum Classification Error (MCE) Training," Proc. *1996 Int. Conf. on Spoken Language Processing*, Vol. 3, pp. 1824-1827, Philadelphia, PA, October 1996.

[Rahim+ 1997] M. Rahim, Y. Bengio, Y. LeCun. "Discriminative Feature and Model Design for Automatic Speech Recognition," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 75-78, Rhodes, Greece, September 1997.

[Ramasubramansian & Paliwal 1992] V. Ramasubramansian, K. K. Paliwal. "Fast k-dimensional Tree Algorithms for Nearest Neighbor Search with Application to Vector Quantization Encoding," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 40, No. 3, pp. 518-528, March 1992.

[Reichl$^+$ 1995] W. Reichl, S. Harengel, F. Wolfertstetter, G. Ruske. "Neural Networks for Nonlinear Discriminant Analysis in Continuous Speech Recognition," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 3, pp. 2163-2166, Madrid, September 1995.

[Reichl & Ruske 1995] W. Reichl, G. Ruske. "Discriminative Training for Continuous Speech Recognition," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 537-540, Madrid, September 1995.

[Rigoll & Willett 1998] G. Rigoll, D. Willett. "A NN/HMM Hybrid for Continuous Speech Recognition with a Discriminant Nonlinear Feature Extraction," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 9-12, Seattle, WA, May 1998.

[Robinson & Fallside 1991] T. Robinson, F. Fallside. "A Recurrent Error Propagation Network Speech Recognition System," Computer Speech and Language, Vol. 5, No. 3, pp. 259-274, 1991.

[Rosenfeld 1994] R. Rosenfeld: *Adaptive Statistical Language Modeling: A Maximum Entropy Approach*, Ph.D. thesis, Technical Report CMU-CS-94-138, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1994.

[Sakoe 1979] H. Sakoe. "Two-Level DP-Matching – A Dynamic Programming-Based Pattern Matching Algorithm for Connected Word Recognition," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 27, pp. 588-595, December 1979.

[Saul & Rahim 2000] L. K. Saul, M. G. Rahim. "Maximum Likelihood and Minimum Classification Error Factor Analysis for Automatic Speech Recognition," IEEE Transactions on Speech and Audio Processing, Vol. 8, No. 2, pp. 115-125, March 2000.

[Schlüter$^+$ 1997a] R. Schlüter, H. Ney, L. Welling. "Diskriminatives Training fr Spracherkennung bei kleinem Wortschatz," Proc. *9. Kolloquium Signaltheorie*, pp. 297-300, Aachen, Germany, March 1997.

[Schlüter$^+$ 1997b] R. Schlüter, W. Macherey, S. Kanthak, H. Ney, L. Welling. "Comparison of Optimization Methods for Discriminative Training Criteria," Proc. *1997 Europ. Conf. on Speech Communication and Technology*, Vol. 1, pp. 15-18, Rhodes, Greece, September 1997.

[Schlüter & Macherey 1998] R. Schlüter, W. Macherey. "Comparison of Discriminative Training Criteria," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 493-496, Seattle, WA, May 1998.

[Schlüter+ 1999a] R. Schlüter, W. Macherey, B. Müller, H. Ney. "Maximum Mutual Information and Maximum Likelihood Approach for Mixture Density Splitting," in Proc. *1999 European Conference on Speech Communication and Technology*, Vol. 4, pp. 1715-1718, Budapest, Hungary, September 1999.

[Schlüter+ 1999b] R. Schlüter, B. Müller, F. Wessel, H. Ney. "Interdependence of Language Models and Discriminative Training," Proc. *1999 Automatic Speech Recognition and Understanding (ASRU) Workshop*, Vol. 1, pp. 119-122, Keystone, CO, December 1999.

[Schlüter+ 2000] R. Schlüter, W. Macherey, B. Müller, H. Ney. "Comparison of Discriminative Training Criteria and Optimization Methods for Speech Recognition," accepted for publication in Speech Communication, March 2000.

[Schlüter & Ney 2000] R. Schlüter, H. Ney. "Model Based MCE Bound to the True *Bayes*' Error," submitted to IEEE Signal Processing Letters, April 2000.

[Schwartz+ 1985] R. Schwartz, Y.-L. Chow, O. Kimball, S. Roucos, U. Krasner, J. Makhoul. "Context-Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech," Proc. *1985 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 1205-1208, Tampa, FL, March/April 1985.

[Schwartz & Chow 1990] R. Schwartz, Y.-L. Chow. "The N-Best Algorithm: An Efficient and Exact Procedure for Finding the N most likely Sentence Hypotheses," Proc. *1990 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 81-84, Albuquerque, NM, April 1990.

[Schwartz & Austin 1991] R. Schwartz, S. Austin. "A Comparison of Several Approximate Algorithms for Finding Multiple (N-Best) Sentence Hypotheses," Proc. *1991 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 701-704, Toronto, May 1991.

[Sixtus+ 2000] A. Sixtus, S. Molau, S. Kanthak, R. Schlüter, H. Ney. "Recent Improvements of the RWTH Large Vocabulary Speech Recognition System on Spontaneous Speech," to appear in Proc. *2000 Int. Conf. on Acoustics, Speech and Signal Processing* Istanbul, June 2000.

[Steinbiss+ 1994] V. Steinbiss, B.-H. Tran, H. Ney. "Improvements in Beam Search," Proc. *1994 Int. Conf. on Speech and Language Processing*, pp. 2143-2146, Yokohama, September 1994.

[Valtchev+ 1993] V. Valtchev, S. Kapadia, S. J. Young. "Recurrent Input Transformations for Hidden Markov Models," Proc. *1993 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 287-290, Minneapolis, MN, April 1993.

[Valtchev 1995] V. Valtchev. *Discriminative Methods in HMM-based Speech Recognition,* Ph.D. thesis, St. John's College, University of Cambridge, Cambridge, March 1995.

[Valtchev+ 1996] V. Valtchev, J. J. Odell, P. C. Woodland, S. J. Young. "Lattice-Based Discriminative Training For Large Vocabulary Speech Recognition," In Proc. *1996 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 605-608, Atlanta, GA, May 1996.

[Valtchev+ 1997] V. Valtchev, J. J. Odell, P. C. Woodland, S. J. Young. "MMIE Training of Large Vocabulary Recognition Systems," Speech Communication, Vol. 22, No. 4, pp. 303-314, September 1997.

[Valtchev 1999] V. Valtchev, Entropic Ltd., Cambridge, England. Private communication, January 1999.

[van Kampen 1992] N. G. van Kampen. "Stochastic Processes in Physics and Chemistry," Elsevier Science Publishers B. C., Amsterdam, 1992.

[Verbmobil homepage] http://verbmobil.dfki.de

[Vintsyuk 1971] T. K. Vintsyuk. "Elementwise Recognition of Continuous Speech Composed of Words from a Specified Dictionary," Cybernetics, Vol. 7, pp. 133-143, March/April 1971.

[Viterbi 1967] A. Viterbi. "Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm," IEEE Transactions on Information Theory, Vol. 13, pp. 260-269.

[Welling+ 1995] L. Welling, H. Ney, A. Eiden, C. Forbrig. "Connected Digit Recognition using Statistical Template Matching," Proc. *1995 Europ. Conf. on Speech Communication and Technology*, Vol. 2, pp. 1483-1486, Madrid, September 1995.

[Welling 1999] L. Welling. *Merkmalsextraktion in Spracherkennungssystemen für großen Wortschatz*, Ph.D. thesis, RWTH Aachen, Becker-Kuns Verlag, Aachen, January 1999.

[Wessel+ 1997] F. Wessel, S. Ortmanns, H. Ney. "Implementation of Word Based Statistical Language Models," Proc. *SQEL Workshop on Multi-Lingual Information Retrieval Dialogs*, pp. 55-59, Pilsen, Czech Republic, April 1997.

[Wessel+ 1998] F. Wessel, K. Macherey, R. Schlüter. "Using Word Probabilities as Confidence Measures," Proc. *1998 Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 225-228, Seattle, USA, May 1998.

[Wessel+ 1999] F. Wessel, K. Macherey, H. Ney. "A Comparison of Wordgraph and N-Best List Based Confidence Measures," Proc. *1999 Europ. Conf. on Speech Communication and Technology*, pp. 315-318, Budapest, September 1999.

[Wessel+ 2000a] F. Wessel, R. Schlüter, H. Ney. "Using Posterior Word Probabilities for Improved Speech Recognition," to appear in Proc. *2000 Int. Conf. on Acoustics, Speech and Signal Processing*, Istanbul, June 2000.

[Wessel+ 2000b] F. Wessel, R. Schlüter, K. Macherey, H. Ney. "Confidence Measures for Large Vocabulary Continuous Speech Recognition," accepted for publication in IEEE Transactions on Speech and Audio Processing, March 2000.

[Woodland+ 1994] P. C. Woodland, J. J. Odell, V. Valtchev, S. J. Young. "Large Vocabulary Speech Recognition using HTK," Proc. *1994 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 125-128, Adelaide, April 1994.

[Woodland+ 1995] P. C. Woodland, C. J. Leggetter, J. J. Odell, V. Valtchev, S. J. Young. "The 1994 HTK Large Vocabulary Speech Recognition System," Proc. *1995 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 573-576, Detroit, MI, May 1995.

[Young 1992] S. J. Young. "The General Use of Tying in Phoneme Based HMM Recognisers," Proc. *1992 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 569-572, San Francisco, CA, March 1992.

[Young 1993] S. J. Young. "HTK: Hidden Markov Model Toolkit V1.4," User Manual, Cambridge University Engineering Department, Cambridge, England, February 1993.

[Young & Woodland 1993] S. J. Young, P. C. Woodland. "The Use of State Tying in Continuous Speech Recognition," Proc. *1993 Europ. Conf. on Speech Communication and Technology*, Vol. 3, pp. 2203-2206, Berlin, September 1993.

[Young+ 1994] S. J. Young, J. J. Odell, P. C. Woodland. "Tree-Based State Tying for High Accuracy Acoustic Modeling," Proc. *ARPA Human Language Technology Workshop*, pp. 307-312, Plainsboro, NJ, Morgan Kaufmann Publishers, March 1994.

Lebenslauf - Curriculum Vitae

Name:	Ralf Schlüter
Adresse:	Jakobstraße 43 52064 Aachen
Geburtstag:	12. März 1968
Geburtsort:	Rheda, jetzt Rheda-Wiedenbrück
Eltern:	Leo Schlüter Hedwig Schlüter, geb. Hegemann

Schulbildung:

August 1974 – Juli 1978:	Wilbrand-Grundschule, Herzebrock-Clarholz
August 1978 – Juni 1987:	Einstein-Gymnasium, Rheda-Wiedenbrück

Zivildienst:

August 1987 – März 1989:	Pflegehelfer am St. Josefs Alten- und Pflegeheim, Herzebrock-Clarholz

Studium:

April 1989 – November 1995:	Physikstudium an der RWTH Aachen
März 1991:	Vordiplom in Physik
Oktober 1991 – Juli 1992:	Akademisches Auslandsjahr an der University of Edinburgh
November 1995:	Diplom in Physik mit Auszeichnung an der RWTH Aachen

Arbeitstätigkeiten:

April 1991 – September 1991, Oktober 1992 – Oktober 1995:	Studentische Hilfskraft am Lehrstuhl für Höhere Mathematik, RWTH Aachen
Dezember 1995 – März 1996:	Wissenschaftlicher Angestellter am II. Physikalischen Institut, RWTH Aachen
Mai 1996 – September 2000:	Promotionsstudent am Lehrstuhl für Informatik 6, RWTH Aachen
Mai 1996 – April 1999:	Stipendiat der Siemens AG, München
März 1998 – September 2000:	Modulverantwortlicher für den Spracherkenner der RWTH Aachen im BMBF-geförderten *Verbmobil 2*-Projekt