Regularized Discrimination of High-Dimensional Signal Representations for Automatic Speech Recognition

高次元音声表現の正則化識別モデルを用いた音声認識

Yotaro Kubo

Human Interface Laboratory, Department of Computer Science and Engineering, Waseda University

B. Information Science, Waseda University, 2007
M. Information Science, Waseda University, 2008

A dissertation submitted to the Graduate School of Fundamental Science and Engineering, Waseda University
in partial fulfillment of the requirements for the degree of Doctor of Engineering

February, 2010

Thesis committee:

Katsuhiko Shirai, Professor
Yasuo Matsuyama, Professor
Yoshinori Sagisaka, Professor
Tetsunori Kobayashi, Professor

Abstract

Automatic speech recognition (ASR), which converts recorded speech signals into word sequences, is one of the most promising technologies for human-machine interaction and media understanding. Recent advances in computer technology have enabled various improvements in speech recognition technologies. Furthermore, the latest developments in machine learning theories and signal processing technologies have also supported the realization of accurate speech recognizers. Although these efforts can realize accurate speech recognition in some cases, further improvement in speech recognition technologies is still necessary to enable more diverse applications.

In conventional ASR technologies, “succinct” representations of speech signals are discriminated by using continuous density hidden Markov models (CD-HMMs). Typically, Mel-frequency cepstral coefficients (MFCCs) and their time-derivatives are used as representations of speech signals. Although the use of this technology has enabled accurate speech recognition, the accuracy of ASR is far from that of human speech recognition.

Because the design of succinct features that are robust against all environmental conditions is difficult, multiple feature extraction modules are often combined. However, the use of combination leads to an increase in the dimensionality of features, which might cause the “curse of dimensionality” problem that degrades the robustness of statistical models. Thus, the use of the multistream approach involves a tradeoff; that is, the increase in the number of combined feature extractions improves robustness in the feature extraction modules, but it degrades the robustness in the statistical models used in ASR.

This thesis tackles the problem arising from such a tradeoff by using regularized discriminative models that effectively handle high-dimensional features. Three elemental technologies are proposed and discussed with the aim of realizing regularized discrimination of high-dimensional signal representations.

This thesis consists of six chapters.

Chapter 1 discusses the current status of speech recognition research and describes the approach used in this thesis. The overview of the thesis is then presented.

Chapter 2 describes conventional feature extraction methods and acoustic models. Furthermore, emerging technologies related to the methods proposed in this thesis are also described.

Chapter 3 proposes a novel feature set based on frequency modulation (FM) of speech with the aim of constructing high-dimensional features. To satisfy the complementarity of feature extraction modules, this chapter focuses on the FM of speech signals. Because most conventional feature extraction methods are based on the spectral envelope and/or amplitude modulation (AM) of the speech signal, the use of FM is reasonable for the complementarity. Conventional studies on FM features use information from FM as supplementary information for conventional features. However, human speech recognition experiments have confirmed that the FM of speech signals also contains phonetic information. Thus, by extracting phonetic information precisely, the FM of speech can be used as independent features for ASR, and it should have complemental property with conventional features. To extract phonetic information from FM, the proposed method applies the nonlinear discriminant analysis method, which is based on multilayer perceptrons (MLPs), to instantaneous frequency sequences. Further, the multiple feature composition method, which is based on the HMM/MLP-tandem-based multistream method, was applied and evaluated in reverberant and noisy environments. The proposed FM features ware confirmed to have a performance comparable with conventional features, even if the FM features do not include spectral envelope information explicitly. Furthermore, combining the FM and AM features was also confirmed to reduce the word errors by 21% when compared with conventional features, and by 20% when compared with AM features used separately.

Chapter 4 proposes a method for constructing a regularized discriminative model based on CD-HMMs. To realize ASR based on regularized discriminative models, this chapter focused on regularized discriminative training of acoustic models. Discriminative training is a family of parameter estimation methods that is known to be effective for constructing highly accurate classifiers. In general, discriminative training can be easily corrupted by an overfitting problem when the training dataset is not sufficient. In contrast, Bayesian inference is known to be a robust method that achieves steady performance even if the training dataset is limited by considering parameters as random variables. Minimum relative entropy discrimination (MRED), also known as maximum entropy discrimination (MED), has been proposed in machine learning research communities with the aim of introducing the randomness of parameters into discriminative training methods, similar to that achieved in the Bayesian inference method. In this chapter, the MRED method is applied to the discriminative training of CD-HMMs. Conventionally, MRED is not applied to training of sequence classifiers with sequence labels (e.g. CD-HMMs). This thesis extends the MRED framework to construct a training algorithm of CD-HMMs. The effectiveness of the proposed method was confirmed by conducting continuous phoneme recognition experiments. The proposed method could reduce the phoneme errors by 6.4% when compared to maximum likelihood training, and by 2.1% when compared to conventional discriminative training.

Chapter 5 proposes a model-based feature augmentation method based on kernel meth-

ods. In conventional HMMs, Gaussian mixture models (GMMs) are used to model each sample in feature vector sequences. Furthermore, the nonlinear classification of each sample is realized by using GMM with a large number of mixture components. However, because the estimation processes of mixture models involve local optima in their optimization performance function, the training of CD-HMMs with a large number of mixture components is easily corrupted by local optima. Further, in general, mixture models often induce overfitting, especially with discriminative training. In contrast, several classification methods, e.g., support vector machines, use kernel methods to realize nonlinear classification of each input sample. To prevent these problems in GMMs, kernel methods are used to enhance emission probability density functions in CD-HMMs; these methods are described in this chapter. To apply the kernel methods, training and evaluation procedures for the models must be represented as the weighted sum of the inner products for each sample in the training dataset. In this chapter, the training and evaluation procedures of the proposed method are rewritten as the weighted sum of the inner products by using pseudo log-linear models and the MRED training method described in the previous chapter. Isolated phoneme recognition experiments were performed to evaluate the proposed methods. These experiments confirmed that the proposed methods could reduce the classification errors by 10.8% when compared to maximum likelihood classifiers, and by 5.8% when compared to discriminatively trained classifiers.

Chapter 6 summarizes the results achieved in this thesis and provides perspectives for future extensions.

Abstract in Japanese

近年の計算資源の増大，および音声信号処理技術や統計的機械学習理論の進歩に伴ない，自動音声認識の精度は飛躍的な向上を遂げてきている．しかし，多様な環境における様々な話者の発話を人間と同等の精度で認識する水準には至っていない．音声認識器は音声信号から音声特徴を分析する特徴分析器，単語毎の音声特徴生起確率を与える音響モデル，単語列の生起確率を与える言語モデルを用いて，最適な単語列を探し出す統計的推定問題の一つとして定式化されている．この枠組みの上では音響パターンの多様性は原則的に特徴分析器および音響モデルによって吸収しなければならない．そのため，特徴分析器および音響モデルの高精度化は重要な課題である．

従来の音声認識器は，音声信号から音韻情報を充分に示すと考えられる低次元の特徴ベクトル列を抽出し，それを隠れマルコフモデル (Hidden Markov Model; HMM) を用いてモデル化することで識別器を構成してきた．しかし，現状の特徴分析法では環境の変化等に頑健であるとは言い難い．また，適切な特徴分析法は環境に応じて変化することが様々な実験によって確認されている．そのため，特徴分析法の多重化は多様な話者／環境に対応するためには必須の手法であると考えられる．これらの手法はマルチストリーム法と呼ばれ音声認識の高精度化のために有効な手法として注目を集めている．しかし同時に，単純な多重化ではモデル化対象のベクトルが高次元化するため，「次元の呪い」と呼ばれる統計モデル上の問題が発生することが指摘されている．つまり，現状の音声認識は，特徴分析法の頑健性を確保するために多重化を行なうと，次元の呪いによって識別器の頑健性が低下してしまうというトレードオフを内在しているということができる．従来のマルチストリーム法では直接これを解決することを避け，多重化のレベルを下げることで問題に対応してきた．

そこで本論文では，多様かつ高次元の音声特徴量を過学習に強い正則化識別モデルとの組合せの中で利用する枠組みを提案する．従来，過学習に強いモデルに関する検討や多様な音声特徴量を用いる検討は個別の問題として取り扱われてきたが，自然言語処理における機械学習の分野等ではこの両者を同時に検討することで大きな成果を上げている．以上を踏まえ，本論文ではこの「高次元音声特徴の正則化識別モデルを用いた音声認識器」の構築に必要な技術について論じる．

本論文は全6章から構成される．

第1章では，本論文の背景と目的について述べ，論文全体の構成を示す．

第2章では，既存の音声認識器について詳説する．最初に統計的アプローチに基づい

た音声認識問題の定式化について解説する．次に現在広く用いられている Mel-Frequency Cepstral Coefficients (MFCC) 特徴分析法，およびそのモデルとしての HMM とその最尤推定について，解説と考察を行なう．続けて，本論文に関連の深い最新の研究についての解説を行なう．

第 3 章では，高次元音声特徴をどのように定義したら良いかについての検討を行ない，新しい特徴分析法の提案を行なう．MFCC 法に代表される従来の音声特徴量分析手法は，音声認識がソースフィルタモデルにおけるフィルタ推定の問題であるという解釈から，より音声認識に適した対数スペクトル包絡を求める問題として検討されてきた．しかし，聴覚的には対数スペクトル包絡以外の情報によっても，音声の知覚がなされていると考えられている．本論文では従来手法との相補性を考え，対数スペクトル包絡にはあらわれにくい特徴である周波数変調を用いる手法を提案する．従来より周波数変調を対数スペクトルの補助情報として用い，音声認識を行なうことで対雑音性能を向上させる検討があったが，それらは全て周波数変調を対数スペクトルの補助として利用するための手法であり，単独では充分な効果を持たないものであった．しかし，人間の音声認識においては，対数スペクトルの情報を人工的に失わせた周波数変調正弦波の混合であっても音韻性を知覚できることが聴覚実験によって確認されている．そのため，周波数変調情報のみによって自動音声認識を行なうことも可能であることが考えられる．さらに，このような従来法との相補性の高い特徴分析法を従来法と多重化させることにより，さらに高精度な認識が可能であることも考えられる．そこで本論文では，瞬時周波数のゆるやかな時間変化に着目し，瞬時周波数系列から識別に有意な変調成分を強調することで特徴量を得た．提案した分析法は既存の周波数変調分析法と比べ 21% の単語誤りを削減することができることを確認した．また，提案した周波数変調分析法を組み合わせることで既存の音声認識器から 20% のエラーを削減することができることを確認した．これにより適切な高次元特徴を定義すれば音声認識の精度を向上させることができることを確認した．

第 4 章では，高次元特徴での利用を見据え過学習をしにくい音響モデルの構築法，特に既存の音声認識器において高い性能を達成している識別学習法について論じる．識別学習法は識別的な基準の最適化によって音響モデルパラメタを推定する方法の総称であり，充分なデータが与えられた上では良い性能を示すことが知られている．しかし，これらの推定法は過学習が起こりやすいことが知られており，これら学習法の利用にはタスクの複雑さに対して最尤推定法よりさらに多くのデータが必要なことが知られている．さらに，高次元特徴の利用も過学習を起こしやすいため，高次元特徴と識別学習の併用は難しい．他方では，過学習が起こりにくい手法として，ベイズ推論に基づく学習が知られている．ベイズ推論はパラメタの値を推定するのではなく，パラメタの分布を推論する枠組みであるため，パラメタ推定量のばらつきを適切にモデル化することができ，過学習を緩和できると言われている．最小相対エントロピー識別は，ベイズ推論と同様のパラメタ分布表現を識別学習に持ち込むことを目的として，機械学習の分野で提案された手法である．本章では高次元特徴量の識別モデルとして，この手法を用いることを考える．従来，最小相対エントロピー識別は静的パターンの識別に用いられてきたが，本章ではこれを時系列パター

ン認識，すなわち教師データが離散変数の系列で表現され，入力が実数値ベクトルの系列で表現されるような識別問題を解くことができるように再定式化を行なった．また，提案法が既存の識別学習法を包含する更新則を持ち，事前分布として特殊な分布形を用いた時のみ既存法に一致するという性質を発見した．事前分布の導入は定式化上，最適化問題への正則化項の導入と同義であると見なすことができ，提案法は正則化識別モデルを音声認識の音響モデルとして導入したことに相当する．正則化項の導入は高次元入力に起因する過学習を軽減する効果のある手法として静的パターン認識の分野では良く知られていることから，音声認識においても過学習の低減という点で効果が期待できる．音素認識実験で提案法の評価を行なったところ，最尤推定と比べ 6.4%，既存の識別学習法と比べ 2.1% の音素誤りを削減することができた．これらの実験を通し正則化識別モデルの有効性を確認した．

第 5 章では，高次元特徴と正則化識別を組み合わせた手法を提案する．音声認識で用いられている HMM では，混合ガウス分布を出力分布として利用することで，非線形の識別を取り扱っている．混合ガウス分布は各種推定で用いられている最適化関数に関して局所解を持つ分布であり，局所解に収束してしまう可能性が高い．また，混合ガウス分布は過学習が起こりやすいことが知られており，特に識別学習を行なった際の過学習が性能に大きく影響を及ぼすことが経験的に知られている．一方，サポートベクタマシンに代表される線形識別関数法に基づく手法では，非線形の識別を実現するため，特徴量を再生核ヒルベルト空間と呼ばれる高次元空間に写像し，その空間で線形識別を行なうことが提案されている．この場合，線形モデルが用いられるため局所解は存在せず，また過学習も起こりにくいと言われている．本章では，このカーネル法に基づく高次元空間への写像およびその空間内での線形の識別を HMM の枠組みの上で実現するために，HMM の出力分布をカーネル法に基づいて拡張することを提案する．カーネル法導入のためには，モデルの学習および評価が全てトレーニングデータに含まれるサンプルの内積に対する線形関数で表現されなければならない．本論文では，HMM の出力分布を非正規化対数線形分布とすることで，認識時に用いられるスコアをパラメタに対し線形の関数で表現できるようにした．また，そのようにして定義したモデルを第 4 章で導入した手法で学習させることにより，各サンプル点の内積の線形関数として表現される目的関数を得た．提案法は音素識別問題にて，最尤推定の HMM と比べて 10.8% の音素誤りを，最大相互情報量推定の HMM と比べて 5.8% の音素誤りを削減することに成功した．またこの結果は第 4 章で導入した学習法が次元の呪いを回避できていることを示しており，高次元特徴の正則化識別の効果を示していると考えられる．

第 6 章では，本論文を総括し結論を導く．また，今後の展望について述べる．

Contents

List of Figures

List of Tables

List of Notations

Constants, hyper-parameters and data

β^0	Hyper-parameter for normal-gamma distribution used as a prior pdf of Gaussian pdf
γ^0	Hyper-parameter for normal-gamma distribution used as a prior pdf of Gaussian pdf
κ	Central frequency of band pass filter
μ^0	Hyper-parameter for normal-gamma distribution used as a prior pdf of Gaussian pdf
η^0	Hyper-parameter for normal-gamma distribution used as a prior pdf of Gaussian pdf
$\phi^0_{s,m}$	Hyper parameter for Dirichlet distributions for mixture proportion
$\varphi^0_{s,m}$	Hyper parameter for Dirichlet distribution for state transition probability matrices
c^0	Hyper parameter for slack prior distribution
D	Number of the dimensionality
$\mathcal{L}$	Set of label sequences used for training of models; $\mathcal{L} = \{\boldsymbol{l}^1, \boldsymbol{l}^2, \cdots\}$
$\boldsymbol{l}^i$	Label sequence in training data set
$\mathcal{X}$	Set of feature vector sequences used for training of models; $\mathcal{X} = \{\boldsymbol{X}^1, \boldsymbol{X}^2, \cdots\}$
$\boldsymbol{X}^i$	i^{th} feature vector sequence in training dataset $\mathcal{X}$; $\boldsymbol{X}^i \overset{\text{def}}{=} \{\boldsymbol{x}^i_1, \boldsymbol{x}^i_2, \cdots\}$

Probabilistic density functions

Dir	Dirichlet distribution
$\mathcal{G}$	Gamma distribution
$\mathcal{N}$	Normal distribution (Gaussian distribution)
$\mathcal{N} \circ \mathcal{G}$	Normal-Gamma distribution

Operators, functions and functionals

$\langle g(x) \rangle_{f(x)}$	Expectation of a function $g(x)$ over a distribution function $f(x)$. i.e. $\langle g(x) \rangle_{f(x)} \overset{\text{def}}{=} \int_x f(x) g(x) dx$
$\mathbb{1}(x, y)$	Kronecker delta function

$\otimes$	Covolution operator between 2 sequences $(\boldsymbol{r} = \boldsymbol{e} \otimes \boldsymbol{v} \rightarrow r_t = \sum_{\tau=0}^{T(v)} e_{t-\tau} v_\tau)$
δ	Label similarity function
ς	Sigmoid function
$\chi^n(\boldsymbol{X}, \boldsymbol{l})$	n^{th} -order statistics obtained from feature vector sequence $\boldsymbol{X}$ and label sequence $\boldsymbol{l}$
$\Omega(\omega)$	Frequency warping function
$\boldsymbol{C}$	Connection weight matrix in MLP
$\mathsf{\mathbf{C}}$	Discrete Cosine transform
$\mathcal{D}$	Discriminant function
$\mathsf{\mathbf{F}}$	Discrete Fourier transform
$\mathcal{H}$	Heaviside-step function
H	Transfer function (or modulation transfer function)
$\mathrm{KL}[p(x) \mid\mid q(x)]$	Kullback Leibler (KL) divergence of p from q i.e. $\mathrm{KL}[p(x) \mid\mid q(x)] \overset{\mathrm{def}}{=} \langle \log p(x) - \log q(x) \rangle_{p(x)}$
$N(a)$	The length of a sequence a, the number of elements in a vector a, or the number of elements in a set a
$\mathcal{S}(\boldsymbol{l})$	Set of all possible state sequence with given label $\boldsymbol{l}$
$\mathcal{S}(A)$	Set of all possible state sequence with given lattice A
$\boldsymbol{\mathcal{T}}(\boldsymbol{x}; \Xi)$	MLP-function
$^\top$	Transpose operator
$\boldsymbol{w}$	Frequency domain window
$\boldsymbol{\zeta}$	Activation function of MLP
$Z(\alpha)$	Partition function of distribution

Variables

$\mathbb{O}$	Zero-crossing interval sequence
α	Set of Lagrange multipliers associated with inequality constraints
Δ^n	Difference in n^{th} -order statistics (delta statistics)
$\rho_{s,m}$	Mixture proportion of m^{th} component in s^{th} state
τ	Variable for time interval
Θ	Set of parameters of acoustic models
Ξ	Parameters for multilayer perceptron
ξ	Set of slack variables
A^i	Lattice corresponding to incorrect label sequences obtained from i^{th} training data
A^i	Lattice corresponding to the i^{th} label sequences in training data
g	Gaussian index

$\boldsymbol{h}$	Impulse response
i	Index for training example
k	Index for frequency bin
Λ	Set of parameters for emission probability density functions
λ_s	Set of parameters for emission probability density functions at s^{th} HMM-state
$\boldsymbol{l}$	Label sequence
m	mixture component index
n	Frame index
$\boldsymbol{\mathcal{P}}$	Transition probability matrix in hidden Markov models and hidden Markov kernel machines
q	Variable for sequence of hidden state in HMMs. $\boldsymbol{q} \overset{\mathrm{def}}{=} \{q_1, q_2, \cdots\}$
$\boldsymbol{R}_g$	Precision matrix
$\boldsymbol{r}$	Recorded audio signal
t	Variable for time
u	Index for MLP layer
$\boldsymbol{X}$	Variable for feature vector sequence

Mathematical notations

In this thesis, lower-case bold letters denote vectors (e.g. $\boldsymbol{x}$). All vectors are considered as column vectors ($N(\boldsymbol{x}) \times 1$ matrices where $N(\boldsymbol{x})$ is dimensionalities of vectors), and $^{\mathsf{T}}$ denotes the transpose operator. Therefore, inner-products of two vectors ($\boldsymbol{x}$ and $\boldsymbol{y}$) are denoted by $\boldsymbol{x}^{\mathsf{T}}\boldsymbol{y}$. The n^{th} element in a vector $\boldsymbol{x}$ is denoted by subscript as x_n. All matrices are denoted by upper-case bold letter (e.g. $\boldsymbol{A}$). $\boldsymbol{I}$ denotes the identity matrix.

Sequences of D-dimensional vectors are considered as $T \times D$ matrices, where T is the number of elements in the sequence. For example, the d^{th} dimension of the n^{th} element in the N-elements sequence of D-dimensional vectors ($\boldsymbol{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$) can be denoted as $x_{t,d} = \{\boldsymbol{x}_t\}_d$. Correspondingly, scalar sequences are considered as vectors ($T \times 1$ matrices), and therefore denoted by lower-case bold letters. The n^{th} element in a sequence $\boldsymbol{l}$ is denoted by l_n.

A variable denoted by a lower case latin alphabet with subscripts is considered as an element in the corresponding matrix and/or vector denoted by the same alphabet with bold letter. For example, a matrix denoted by $\boldsymbol{A}$ contains column vectors denoted by $\boldsymbol{a}_i$ and scalars denoted by $a_{j,i}$, as follows:

$$\begin{aligned}\boldsymbol{A} &\overset{\text{def}}{=} [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_i, \cdots] \overset{\text{def}}{=} \begin{bmatrix} a_{1,1} & \cdots & a_{1,i} & \cdots \\ \vdots & & \vdots & \\ a_{j,1} & \cdots & a_{j,i} & \cdots \\ \vdots & & \vdots & \end{bmatrix}, \\ \boldsymbol{x} &\overset{\text{def}}{=} [x_1, \cdots, x_d, \cdots]^{\mathsf{T}}.\end{aligned} \tag{1}$$

While detailed meanings of variables are different, the variables with the roughly same meaning are notated by the same letter. In this thesis, such variables are distinguished by superscripts. For example, feature sequences are denoted by the same letter $\boldsymbol{X}$ inspite of the difference in their extraction methods. In order to prevent this ambiguity, superscripts are used to distinguish the feature sequences extracted by different methods. Specifically, $\boldsymbol{X}^{\mathtt{FM}}$ denotes an FM feature vector sequence, and $\boldsymbol{X}^{\mathtt{AM}}$ denotes an AM feature vector sequence in Chapter 3.

In order to respect the above notation system, some conventional rules practiced in signal processing textbooks are not compatible with this thesis.

Chapter 1

Introduction

Automatic speech recognition (ASR), which converts recorded speech signals into word sequences, is one of the most promising technologies for human-machine interaction and media understanding. However, it is currently difficult to perform ASR accurately since speech signals include a wide variety due to several factors such as recording environments, speakers, and/or speaking styles. Thus, the speech recognition is now considered as one of the most difficult problems in pattern recognition research.

Recent advances in computer technology have enabled various improvements in ASR technologies. For instance, in [McDermott et al., , Woodland, 2002, Lööf et al., 2007], the complex probabilistic models, which involve over 20,000 Gaussian probability density functions (pdfs), are trained by the large datasets, such as *Corpus of Spontaneous Japanese* (CSJ) [Maekawa, 2003], *European parliament plenary sessions* (EPPS), and *SWITCHBOARD* [Godfrey et al., 1992]. Further, the latest developments of machine learning theories and signal processing technologies also support the construction of accurate speech recognizers. Although these efforts can realize accurate ASR in some particular cases, further improvements in ASR technologies are still necessary to enable more diverse applications. The main objective of this thesis is to provide methods aiming for construction of accurate speech recognizers that can identify contents of speech signals by enhancing signal processing and statistical estimation techniques used in speech recognition.

In this chapter, at first, the current scheme of speech recognition is briefly described. Then, the conceptual framework underlying this thesis is described.

1.1 Automatic speech recognition

The ASR problem is formulated as a probabilistic decision problem, i.e., the relationship between speech signals and recognition results is defined by probabilistic distribution functions (pdfs). In this formulation, the relevant word sequence $\hat{l}$ is selected so as to maximize a probability of a label sequence $\boldsymbol{l}$, given the speech sample sequence $\boldsymbol{r}$, as follows:

$$\hat{\boldsymbol{l}} = \underset{\boldsymbol{l}}{\mathrm{argmax}} P(\boldsymbol{l}|\boldsymbol{r}). \tag{1.1}$$

Here, since direct modeling of the speech sample sequence $\boldsymbol{r}$ is intractable, the conditional probability is approximated by introducing a feature extraction function $\boldsymbol{X} = \Phi(\boldsymbol{r})$ [*1], as follows:

$$\hat{\boldsymbol{l}} \approx \underset{\boldsymbol{l}}{\mathrm{argmax}} P(\boldsymbol{l}|\boldsymbol{X}), \tag{1.2}$$

where the feature vector sequence $\boldsymbol{X}$ is computed from the recorded speech signal $\boldsymbol{r}$. By using Bayes' law, the above equation is expanded as follows:

$$\hat{\boldsymbol{l}} \approx \underset{\boldsymbol{l}}{\mathrm{argmax}} P(\boldsymbol{l}|\boldsymbol{X}) = \underset{\boldsymbol{l}}{\mathrm{argmax}} \frac{P(\boldsymbol{X}|\boldsymbol{l})P(\boldsymbol{l})}{P(\boldsymbol{X})} = \underset{\boldsymbol{l}}{\mathrm{argmax}} P(\boldsymbol{X}|\boldsymbol{l})P(\boldsymbol{l}). \tag{1.3}$$

In this formulation, four components are used to construct speech recognizers, as follows:

- **Feature extractors** that compute a feature vector sequence $\boldsymbol{X} = \Phi(\boldsymbol{r})$ from a recorded speech signal $\boldsymbol{r}$,
- **Acoustic models** that are used to represent a emission probability of a feature vector sequence $\boldsymbol{X}$, given a label sequence $\boldsymbol{l}$ ($P(\boldsymbol{X}|\boldsymbol{l})$),
- **Language models** that are used to represent a probability of a label sequence $\boldsymbol{l}$ ($P(\boldsymbol{l})$), and
- **Decoders** that search the relevant word sequence $\hat{\boldsymbol{l}}$ that maximizes $P(\boldsymbol{X}|\hat{\boldsymbol{l}})P(\hat{\boldsymbol{l}})$.

Figure 1.1 shows the block diagram of speech recognizers.

The following subsections describe each component in the diagram. This chapter only provides a conceptual overview of each component. The detailed discussions and literature reviews about these components focused in this thesis are presented in the next chapter.

1.1.1 Feature extraction

The ultimate objective of feature extraction modules is to provide effective representations of speech signals to acoustic models. Therefore, the studies on acoustic modeling and feature extraction are inseparable. However, in general, these two modules are independently developed. Most conventional studies attempt to extract some physical quantities, which succinctly explain phenomena of speech production and transmission, since they are suitable for use with conventional generative acoustic models.

[*1] As mentioned in the page xvii, upper-case bold letters (e.g. $\boldsymbol{X}$) are used to represent sequences of vectors. Therefore, this equation implies that a vector sequence $\boldsymbol{X}$ is extracted from a scalar sequence $\boldsymbol{r}$ by using the feature extraction function Φ.

Since speech signals can be assumed to be stationary in a short-time segment, input signals are often split into short-time segments termed “frame.” Conventionally, fixed-length ($\approx$ 25 ms) segments are taken for each frame-shift length ($\approx$ 10 ms). Thus, the n^{th} element $\boldsymbol{x}_n$ in the feature vector sequence $\boldsymbol{X}$ is computed by using a framewise feature extraction function Φ_n and a short-time segment $\{r_t | \forall t \in [T_n^{\mathtt{START}}..T_n^{\mathtt{STOP}}]\}$, as follows:

$$\boldsymbol{x}_n = \Phi_n(\{r_t | \forall t \in [T_n^{\mathtt{START}}..T_n^{\mathtt{STOP}}]\}) \tag{1.4}$$

where $T_n^{\mathtt{START}}$ and $T_n^{\mathtt{STOP}}$ are the indices of the first sample and the last sample in the n^{th} frame, denoted as follows:

$$\begin{aligned} T_n^{\mathtt{START}} &= n\tau^{\mathtt{SHIFT}} \\ T_n^{\mathtt{STOP}} &= T_n^{\mathtt{START}} + \tau^{\mathtt{WINLEN}} - 1. \end{aligned} \tag{1.5}$$

Here, $\tau^{\mathtt{SHIFT}}$ is the frame-shift length, and $\tau^{\mathtt{WINLEN}}$ is the number of samples in the window (window length).

1.1.2 Acoustic models

Acoustic models are used to represent the pdf $P(\boldsymbol{X}|\boldsymbol{l})$, which represents the emission probability of the observed feature vector sequence $\boldsymbol{X}$, given a label sequence $\boldsymbol{l}$. One of the

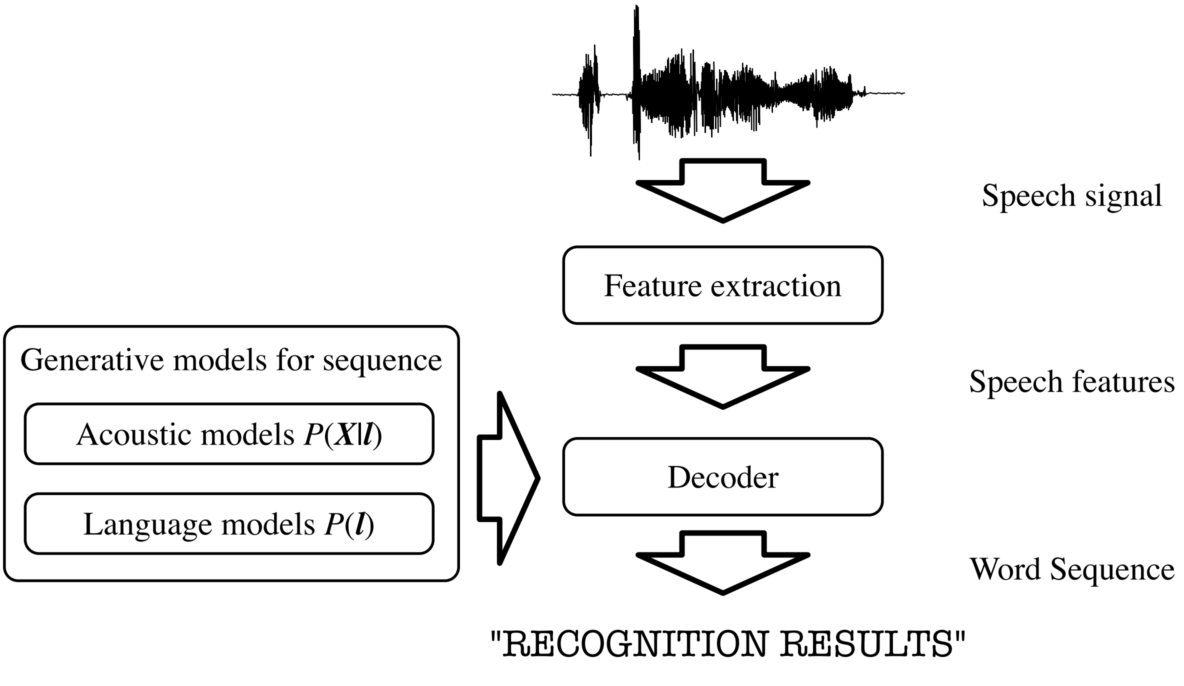


Figure 1.1 Schematic diagram of current speech recognizers

difficulties in acoustic models is that both of the variable to be modeled $\boldsymbol{X}$ and the conditioning variable $\boldsymbol{l}$ are structural (sequential), which involve combinatorial explosion. Therefore, multiple observations ($\boldsymbol{X}$s) are rarely obtained for the exactly same $\boldsymbol{l}$.

A widely accepted solution is to use models that are defined with respect to each label element l_n where $\boldsymbol{l} = \{l_1, l_2, \cdots\}$. Practically, l_n is designed to represent each word, phoneme, or context-dependent phoneme (e.g. triphone) in the label sequence. The continuous-density hidden Markov model (CD-HMM) is the most widely accepted model that use the abovementioned strategy. Since CD-HMMs have a concatenation operator, they can handle sequences of labels by concatenating CD-HMMs corresponding to each label element in the given label sequence.

1.1.3 Language models and decoders

Language models are used to represent probabilities of label sequences. In general, a word N-gram model is used as a language model. N-gram models predict label element l_n from preceding $N-1$ label elements For example, 3-gram models represent the probability of a label sequence $\boldsymbol{l}$ as $P(\boldsymbol{l}) = \prod_n P(l_n | l_{n-1}, l_{n-2})$. In language models, typically, elements l_n are designed to represent each word. It should be noted that the inconsistency in types of elements of label sequences used in the acoustic models and the language models can be resolved since probability of a word sequence is translated into probability of a phoneme sequence by decoders.

The decoders are one of the most important modules for computational efficiency of ASR. In large vocabulary continuous speech recognition (LVCSR) problems, exact search over hypothesis label sequences might be computationally prohibitive. Recent decoders enable LVCSR by employing approximated search algorithms, such as beam search methods and/or A* search methods. Although the use of these approximated search algorithms causes errors due to approximation, called “search error,” recent advances in decoding algorithms satisfy both of the computational efficiency and the recognition accuracy.

In this thesis, these modules are not mentioned because they are rarely related with acoustical fluctuations of the observed signals $\boldsymbol{r}$. However, the methods proposed in this thesis would cooperate with latest advances in language models and decoders.

1.2 High-dimensional speech representations

According to the above scheme, degradations due to the fluctuations in speech patterns should appear in feature extractors and acoustic models. Therefore, sophisticated feature extractors and acoustic models are required in order to achieve accurate ASR.

Conventionally, feature extractors are designed to extract *succinct* features that are considered as containers of phonetic information. Further, the acoustic models are trained to be a good generator of these succinct features. This scheme is effective when the extracted features have sufficient information required for ASR. However, generally speaking, most conventional feature extraction methods have pros and cons depend on situations. Thus, the combination of conventional features is considered as effective. *Multistream speech recognizer* is a generic term for speech recognizers that contain multiple feature extraction modules.

However, it is known that the use of multiple features leads to inefficiency called *curse of dimensionality*, which is the set of phenomena that appears when high-dimensional vectors are modeled by statistical models. The following undesirable effects are often observed in such cases.

- Distances between two vectors independently sampled converge to a constant (a.k.a. *concentration of measure* [Bishop, 2006]).
- Increase in the number of parameters in model pdfs results in increase in the bias term, which indicates the sensitivity to the training data shortage.
- Computational resources required for model estimation and evaluation increase.

Despite these effects, the conventional multistream speech recognizers work accurately, by using several techniques such as ensemble classification methods and hybrid classification methods [Morgan et al., 2005]. This thesis attempts to directly resolve this inefficiency by introducing regularized discrimination of high-dimensional signal representations.

1.3 Regularized discrimination of high-dimensional speech representations

Recently, the importance of regularization in optimization of classifiers is confirmed in several application areas. For example, in speech recognition, an I-smoothing technique is introduced as a regularization technique in discriminative training of acoustic models [Povey and Woodland, 2002].

One of the most successful methods with regularization is the support vector machine (SVM). In SVMs, L2-norm regularization is introduced to parameter vectors in order to obtain large margin linear classifiers [Boser et al., 1992]. Since SVMs achieved robust classification even if input vectors are mapped to a higher-dimensional space, it is assumed that the SVMs can prevent the *curse of dimensionality*.

Regularization is a technique that introduces additional terms in objective functions of optimization problems in order to prevent overfitting and reduce the generalization error. The term “generalization error” indicates the expectation of the amount of errors over the *true* distribution of input examples. Obviously, since the true distribution of the input examples is unknown in general, direct evaluation of generalization error is impossible. In several methods, empirical errors, which indicate the amount of errors with respect to the examples in the given dataset, are minimized instead of generalization errors. However, although an empirical risk minimization can easily be achieved by using complex models [*2] (e.g. 1-nearest neighbor classifiers can achieve zero empirical error), it is well known that most statistical models involve trade-off between the empirical error minimization and generalization error minimization as shown in Figure 1.2.

Regularization techniques are often introduced in order to adjust this trade-off. Especially, regularization techniques are important in high-dimensional discrimination since a model complexity is too high in several cases even if a naive model, such a linear model, is chosen. Thus, regularization techniques are necessary in order to adjust trade-off between empirical error and generalization error of high-dimensional models.

1.4 Contributions

The contributions of this thesis are aiming for realizing a scheme for ASR that can efficiently cope with high-dimensional features obtained by multiple feature extraction methods and feature augmentation methods.

For this purpose, regularized discrimination of high-dimensional speech features is introduced for ASR. The efficiency of the scheme that combines high-dimensional features and regularized discrimination is also confirmed in recent advances in natural language processing. Because mechanism in generation of natural language texts are rarely known, multiple

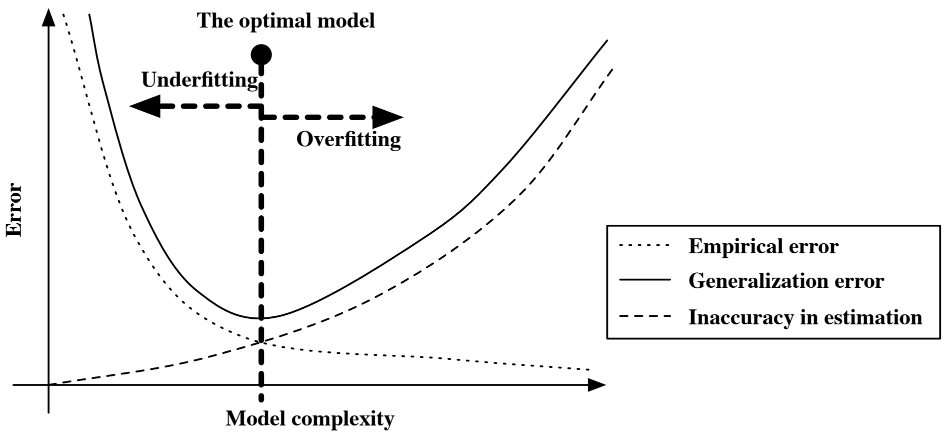


Figure 1.2 Trade-off between empirical risk minimization and generalization error minimization

[*2] In this thesis, the term "complex model" denotes models with a number of parameters.

and redundant features are used instead of succinct features extracted so as to avoid redundancy. Although the mechanism of speech perceptions and productions have been deeply investigated, the use of high-dimensional features would be efficient because these mechanisms are still under investigation.

Specifically, this thesis focused the following topics in construction of speech recognizers.

1. Constructing of high-dimensional features,
2. Obtaining regularized discriminative models in order to prevent *curse of dimensionality*, and
3. Transforming feature vectors in order to enrich representation of speech features.

1.5 Overview

The proposed scheme of speech recognition and the organization of this thesis is illustrated in Figure 1.3.

This thesis is organized as follows:

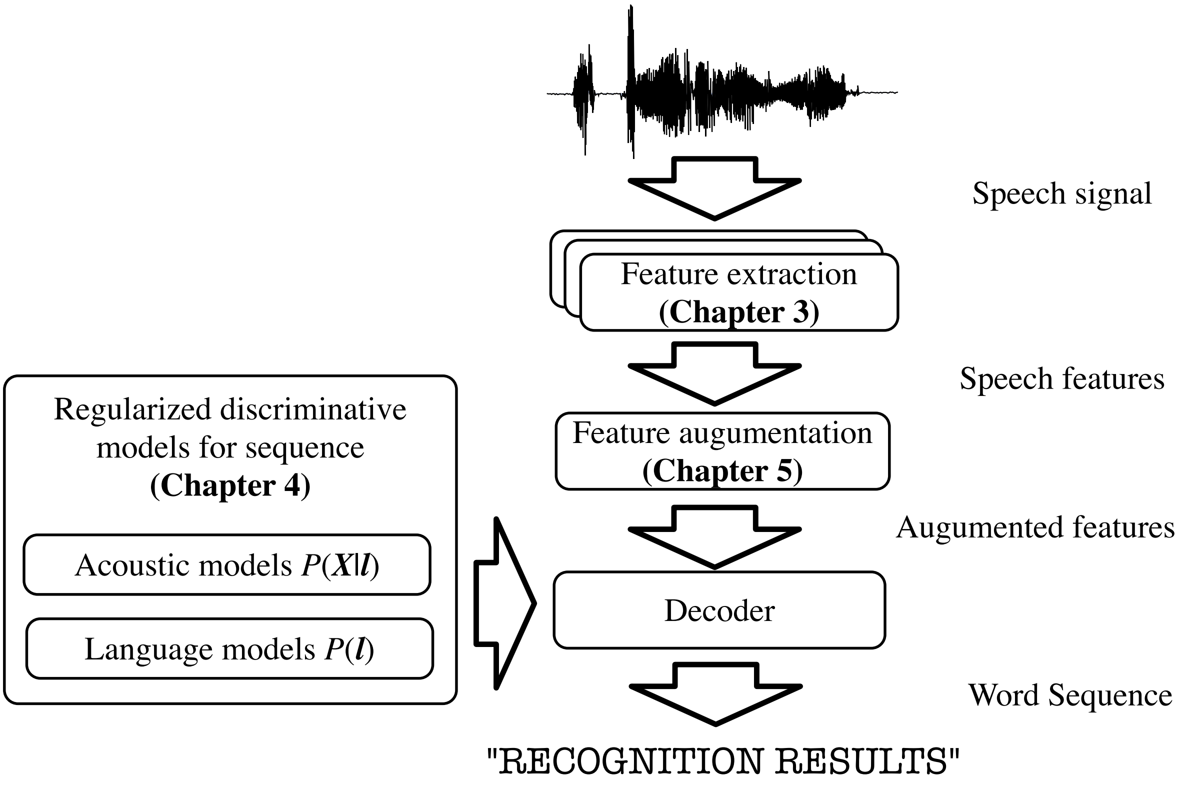


Figure 1.3 The schematic diagram of proposed speech recognizers

Chapter 2 describes conventional feature extraction methods and acoustic models. Furthermore, emerging technologies related to the methods proposed in this thesis are also described.

Chapter 3 proposes a novel feature set based on frequency modulation (FM) of speech with the aim of constructing high-dimensional features. To satisfy the complementarity of feature extraction modules, this chapter focuses on the FM of speech signals. Because most conventional feature extraction methods are based on the spectral envelope and/or amplitude modulation (AM) of the speech signal, the use of FM is reasonable for the complementarity. To extract phonetic information from FM, the proposed method applies a nonlinear discriminant analysis method, which is based on multilayer perceptrons (MLPs), to instantaneous frequency sequences. Further, the multiple feature composition method, which is based on the HMM/MLP-tandem-based multistream method, is applied and evaluated in noisy and reverberant environments.

Chapter 4 proposes a method for constructing a regularized discriminative model based on CD-HMMs. To realize ASR based on regularized discriminative models, this chapter focused on regularized discriminative training of acoustic models. In this chapter, the author proposes an application method of minimum relative entropy discrimination (MRED; a.k.a. maximum entropy discrimination (MED)) for CD-HMMs. By considering the Bayesian inference as a generalization of the maximum likelihood estimation, generalization for discriminative training methods of the CD-HMMs can be considered in a similar way. MRED is a way to generalize discriminative models [Jaakkola et al., 2000, Jebara, 2001]. In this chapter, a generalized method of the conventional discriminative training methods is derived by applying MRED to discriminative training of CD-HMMs.

Chapter 5 proposes a model-based feature augmentation method based on kernel methods. By applying MRED for HMMs, a feature augmentation method based on kernel methods is introduced to HMMs in a straightforward way. Hidden Markov kernel machines (HMKMs) are proposed as an extension to conventional CD-HMMs in this chapter. Since the kernel method project original feature vectors into a higher-dimensional space, the method proposed in this chapter can be assumed as a combination of high-dimensional features and regularized discriminative models.

Chapter 6 summarizes the results achieved in this thesis and provides perspectives for future extensions. Further, final remarks are presented.

Chapter 2

Background

In this chapter, conventional methods for feature extraction and acoustic model training are described in Section 2.1 and Section 2.2, respectively. Each section begins with a description about the most widely accepted method, and then the *state-of-the-art* methods that are closely related to the methods proposed in this thesis are described. Further, in Section 2.3, hidden Markov models/ multi-layer perceptron (HMM/MLP)-tandem approach is described as a conventional method used for handling high-dimensional features.

2.1 Feature extraction

Since raw signals are intractable in conventional acoustic models, a sequence of feature vectors $\boldsymbol{X} \overset{\text{def}}{=} \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots\}$ are extracted from the raw signal $\boldsymbol{r} \overset{\text{def}}{=} \{r_1, r_2, \cdots\}$.

2.1.1 Mel-frequency cepstral coefficients (MFCC)

Most feature extraction methods stand on the *source-filter* model of speech production. In this model, the speech recognition problems are considered as a blind filter estimation problem of speech signals by assuming randomness of the source signal. Mel-frequency cepstral coefficient (MFCC) feature extraction is one of the most widely accepted feature extraction methods based on this assumption.

By considering the *source-filter* model of speech production, speech signals are assumed as convolutions of source signals emanated from vocal cords and impulse responses of a vocal tract. In general, filters are assumed to be quasi-stationary, i.e. an impulse response of the filter is constant in a short-time segment. Thus, by applying the Wiener-Khinchin theorem [Oppenheim et al., 1989], k^{th} component of the short-time Fourier transform (STFT) of n^{th} frame ($x_{n,k}^{\texttt{STFT}}$) is expressed as the product of the STFT of the source signal $e_{n,k}$ and that of the impulse response $v_{n,k}$, as follows:

$$x_{n,k}^{\texttt{STFT}} = e_{n,k} \cdot v_{n,k}. \tag{2.1}$$

Taking logarithm of squared-magnitude yields:

$$\log |x^{\mathtt{STFT}}_{n,k}|^2 = 2\log|e_{n,k}| + 2\log|v_{n,k}|. \tag{2.2}$$

This equation implies that shifts in the logarithmic power spectrum can be assumed as the sum of shifts in frequency characteristics of the vocal tract and the source signal.

An extraction process of MFCCs attempts to suppress the variation due to the source signal. First, the MFCC extraction process applies high-pass filtering (pre-emphasis) as a preprocessing in order to flatten spectrum of the signal $x^{\mathtt{STFT}}_{n,k}$. Then, power spectrum components $|x^{\mathtt{STFT}}_{n,k}|^2$ are processed by frequency domain windows $\boldsymbol{w}^{\mathtt{MEL}}_d = [w^{\mathtt{MEL}}_{d,1}, w^{\mathtt{MEL}}_{d,2}, \cdots]$ which simulate human hearing characteristics of sound pitch.

$$x^{\mathtt{MEL}}_{n,d} = \log\left\{\sum_k w^{\mathtt{MEL}}_{d,k}|x^{\mathtt{STFT}}_{n,k}|^2\right\}. \tag{2.3}$$

As described in page xvii, $x^{\mathtt{MEL}}_{n,d}$ is considered as the $(n,d)^{\mathrm{th}}$ element of the matrix $\boldsymbol{X}^{\mathtt{MEL}}$ and d^{th} element of the vector $\boldsymbol{x}^{\mathtt{MEL}}_n$. Note that $x^{\mathtt{STFT}}_{n,k}$ in this equation is obtained from pre-emphasized signals. Central frequencies $\kappa^{\mathtt{MEL}}_d$ of the spectral windows $w^{\mathtt{MEL}}_{d,k}$ are equally spaced in Mel-frequency domain $\Omega^{\mathtt{MEL}}$, and the slopes of the windows are linear, which are defined as follows:

$$\begin{aligned}
w^{\mathtt{MEL}}_{d,k} &= \begin{cases} \frac{k-\kappa^{\mathtt{MEL}}_{d-1}}{\kappa^{\mathtt{MEL}}_{d}-\kappa^{\mathtt{MEL}}_{d-1}} & \kappa^{\mathtt{MEL}}_{d-1} < k \leq \kappa^{\mathtt{MEL}}_{d} \\ \frac{\kappa^{\mathtt{MEL}}_{d+1}-k}{\kappa^{\mathtt{MEL}}_{d+1}-\kappa^{\mathtt{MEL}}_{d}} & \kappa^{\mathtt{MEL}}_{d} < k < \kappa^{\mathtt{MEL}}_{d+1} \\ 0 & \text{otherwise,} \end{cases} \\
\kappa^{\mathtt{MEL}}_d &= \frac{K}{\omega^{\mathtt{RATE}}}\Omega^{\mathtt{IMEL}}\left(\frac{d\left(\Omega^{\mathtt{MEL}}(\omega^{\mathtt{STOP}}) - \Omega^{\mathtt{MEL}}(\omega^{\mathtt{START}})\right)}{D+1} + \Omega^{\mathtt{MEL}}(\omega^{\mathtt{START}})\right), \\
\Omega^{\mathtt{MEL}}(\omega) &= 2595\log_{10}\left(1+\frac{\omega}{1400\pi}\right), \\
\Omega^{\mathtt{IMEL}}(m) &= 1400\pi\left(10^{\frac{m}{2595}}-1\right), \\
\kappa^{\mathtt{MEL}}_0 &\overset{\mathrm{def}}{=} \frac{\omega^{\mathtt{START}}}{\omega^{\mathtt{RATE}}}K, \\
\kappa^{\mathtt{MEL}}_{D+1} &\overset{\mathrm{def}}{=} \frac{\omega^{\mathtt{START}}}{\omega^{\mathtt{RATE}}}K,
\end{aligned} \tag{2.4}$$

where $\omega^{\mathtt{START}}$ and $\omega^{\mathtt{STOP}}$ are lower and upper frequency cutoff in angular frequency (rad/s), respectively, $\Omega^{\mathtt{MEL}}(\omega)$ is the frequency warping from angular frequency to the Mel-frequency, $\Omega^{\mathtt{IMEL}}(\omega)$ is the inverse frequency warping of $\Omega^{\mathtt{MEL}}(\omega)$, $\omega^{\mathtt{RATE}}$ is the sampling rate in angular frequency, D is the number of channels in Mel-filter bank, and K is the number of frequency bin. Figure 2.1 shows $w^{\mathtt{MEL}}_{d,k}$ as a function of k.

Generally speaking, the spectrum of the source $e_{n,k}$ stay constant for all k when an unvoiced sound is presented, and $e_{n,k}$ has periodicity along the k-axis when a voiced sound is presented. Therefore, variation due to source signals of unvoiced sounds are suppressed by subtracting the average of the $x^{\texttt{MEL}}_{n,k}$s for all k, and that of voiced sounds are suppressed by removing microscopic fluctuations in the $x^{\texttt{MEL}}_{n,k}$ along the k-axis. These operations, the subtraction of the average and the microscopic fluctuations, can be performed by applying a band-pass filter (BPF) to $x^{\texttt{MEL}}_{n,k}$ where k is considered as a time-domain variable. This log-spectral domain filtering operation is termed "liftering." The spectrum of log-spectrum is termed "cepstrum", and the frequency-axis of a cepstrum is termed "quefrency."

By obtaining a cepstrum by using discrete cosine transform (DCT), the d^{th} element of MFCCs at n^{th} frame is defined as follows:

$$x^{\texttt{MFCC}}_{n,d} \overset{\mathrm{def}}{=} \left(\mathbf{C} \cdot \boldsymbol{x}^{\texttt{MEL}}_{n}\right)_{d+d^{\texttt{LOQUE}}} \quad (1 \leq d \leq d^{\texttt{HIQUE}} - d^{\texttt{LOQUE}}) \tag{2.5}$$

where $d^{\texttt{LOQUE}}$ and $d^{\texttt{HIQUE}}$ are the lower cutoff quefrency and the upper cutoff quefrency of the band-pass-liftering, $\mathbf{C}$ is the DCT matrix. Typically, $d^{\texttt{LOQUE}}$ and $d^{\texttt{HIQUE}}$ are set at 1 and 13, respectively.

MFCCs are often used with an energy feature and their first/second-order derivatives [Furui, 1981]. Typical feature vector $\boldsymbol{x}^{\texttt{MFCC\_E\_D\_A}}_{n}$ is defined by augmenting the energy feature

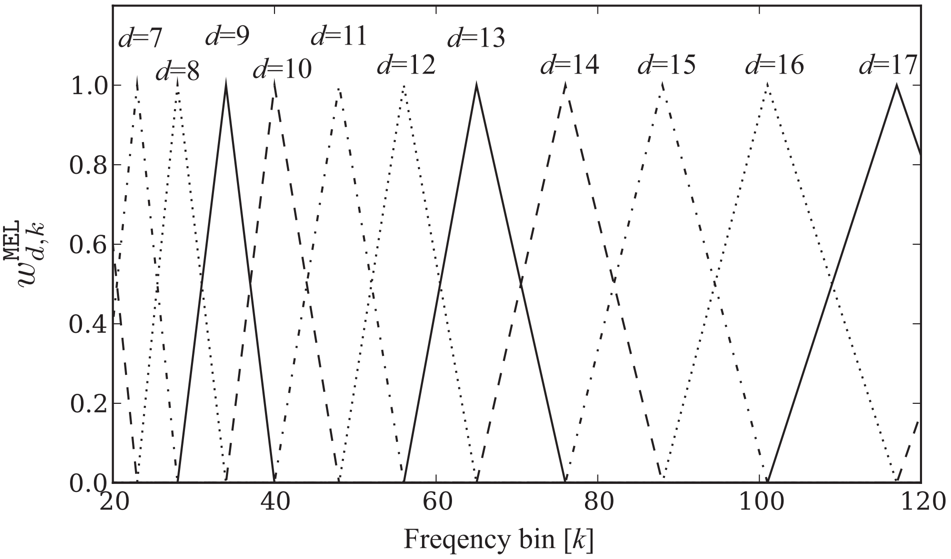


Figure 2.1 Frequency characteristics of Mel-filterbank $w^{\texttt{MEL}}_{d,k}$ as a function of a frequency bin k. (Sample rate = 16000 Hz, B = 20, upper cutoff frequency = 8000 Hz, lower cutoff frequency = 0 Hz)

and derivatives, as follows:

$$\begin{aligned}
\boldsymbol{x}_n^{\mathtt{MFCC\_E\_D\_A}} &\overset{\text{def}}{=} \left[\left(\boldsymbol{x}_n^{\mathtt{MFCC\_E}}\right)^{\mathsf{T}}, \left(\nabla_n \boldsymbol{x}_n^{\mathtt{MFCC\_E}}\right)^{\mathsf{T}}, \left(\nabla_n \nabla_n \boldsymbol{x}_n^{\mathtt{MFCC\_E}}\right)^{\mathsf{T}} \right]^{\mathsf{T}}, \\
\boldsymbol{x}_n^{\mathtt{MFCC\_E}} &\overset{\text{def}}{=} \left[\left(\boldsymbol{x}_n^{\mathtt{MFCC}}\right)^{\mathsf{T}}, \left(x_n^{\mathtt{E}}\right)^{\mathsf{T}} \right]^{\mathsf{T}}, \\
x_n^{\mathtt{E}} &= \log \left\{ \sum_k |x_{n,k}^{\mathtt{STFT}}|^2 \right\},
\end{aligned} \tag{2.6}$$

where ∇_n is the partial derivative operator with respect to the variable n. Typically, the derivatives are obtained by a numerical method, as follows:

$$\nabla_n f(n) = \frac{\sum_{\tau=1}^{T} f(n+\tau) - f(n-\tau)}{2 \sum_{\tau=1}^{T} \tau^2} \tag{2.7}$$

where T is the number of frames used to compute the derivatives, which is typically set at 1 or 2. A block diagram of an MFCC_E_D_A extraction process is presented in Figure 2.2.

It should be noted that ∇ is a linear convolution operator of sequence. A derivative operator $\nabla_n x_{n,d}$ of a feature $x_{n,d}$ can be expressed as the following convolution form:

$$\begin{aligned}
\nabla_n x_{n,d} &= \sum_{\tau=-T}^{T} h_{\tau+T+1}^{\mathtt{DELTA}} + x_{n-\tau,d}, \\
\boldsymbol{h}^{\mathtt{DELTA}} &\overset{\text{def}}{=} \begin{cases} \{-\frac{1}{2}, 0, \frac{1}{2}\} & (T=1), \\ \{-\frac{2}{10}, -\frac{1}{10}, 0, \frac{1}{10}, \frac{2}{10}\} & (T=2), \\ \vdots \end{cases}
\end{aligned} \tag{2.8}$$

where $h_\tau^{\mathtt{DELTA}}$ is the τ^{th} element in the sequence $\boldsymbol{h}^{\mathtt{DELTA}}$. Thus, the derivative operations are equivalent to a time-domain filtering operation of feature trajectories. The frequency characteristics of this filter depend on the time window length T. The next subsection focused on filtering techniques of feature trajectories.

2.1.2 Amplitude modulation of speech

Although features based on the *source-filter* model, such as MFCC, realize accurate speech recognition in particular environments, speech recognition in realistic environments involves other problems due to room acoustics and transmission channels. Eq. (2.1) only considers the source signal and the vocal tract filter. However, filters due to characteristics of rooms and transmission channels can not be negligible in realistic environments. In order to avoid effects arose from rooms and transmission channels, feature compensation methods based on amplitude modulation (AM) of speech signals are proposed.

These AM methods focus on the speed of variations in power spectrum. Since the characteristics of rooms and transmission channels are varying very slowly or staying constant,

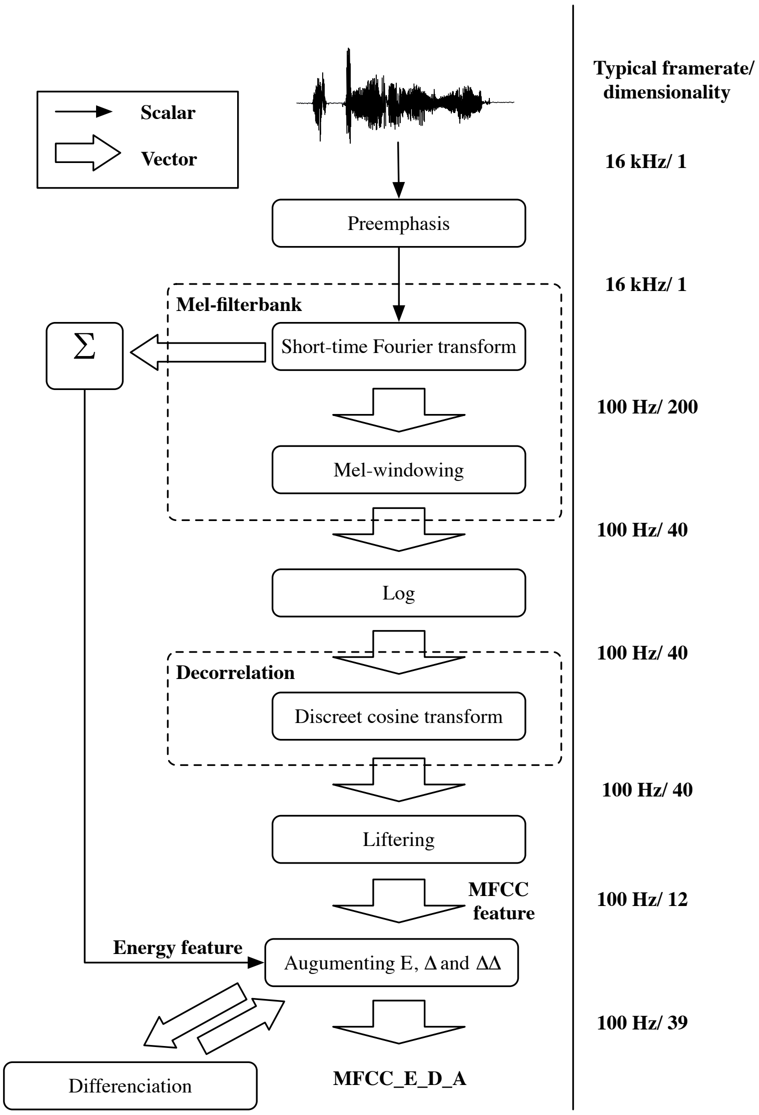


Figure 2.2 Block diagram of MFCC_E_D_A feature extraction

suppressing lower-frequency variations in feature sequences might be effective. Further, since vocal tracts cannot move so fast, higher-frequency components are considered as a non-informative part due to additive noises. Thus, band-pass filtering in feature trajectory domain is considered as an essential operation for feature extraction.

In the **rel**ative **spe**c**tra** (RASTA) technique [Hermansky and Morgan, 1994], which is inspired by the human hearing experiments [Green, 1976, Houtgast and Steeneken, 1985], feature variations around 4 Hz are emphasized. In order to perform this emphasis in feature sequence, RASTA applies the filter corresponding to the transfer function $H^{\texttt{RASTA}}(z)$ to feature trajectory. The transfer function $H^{\texttt{RASTA}}(z)$ is defined as follows:

$$H^{\texttt{RASTA}}(z) = 0.1z^4 \frac{2 + z^{-1} - z^{-3} - 2z^{-4}}{1 - 0.98z^{-1}}. \tag{2.9}$$

The RASTA technique is performed by applying the autoregressive moving-average (ARMA) filter defined by this equation to each component in feature vectors. Figure 2.3 shows the block diagram of ARMA filter used to perform the RASTA technique.

Figure 2.4 shows the comparison of frequency characteristics of time-domain feature filtering methods. From this figure, it is confirmed that the differentiation operation only emphasizes higher-frequency movements. Contrastingly, RASTA method emphasize the components around 4 Hz. Since the efficiency of the RASTA operation is confirmed by conducting

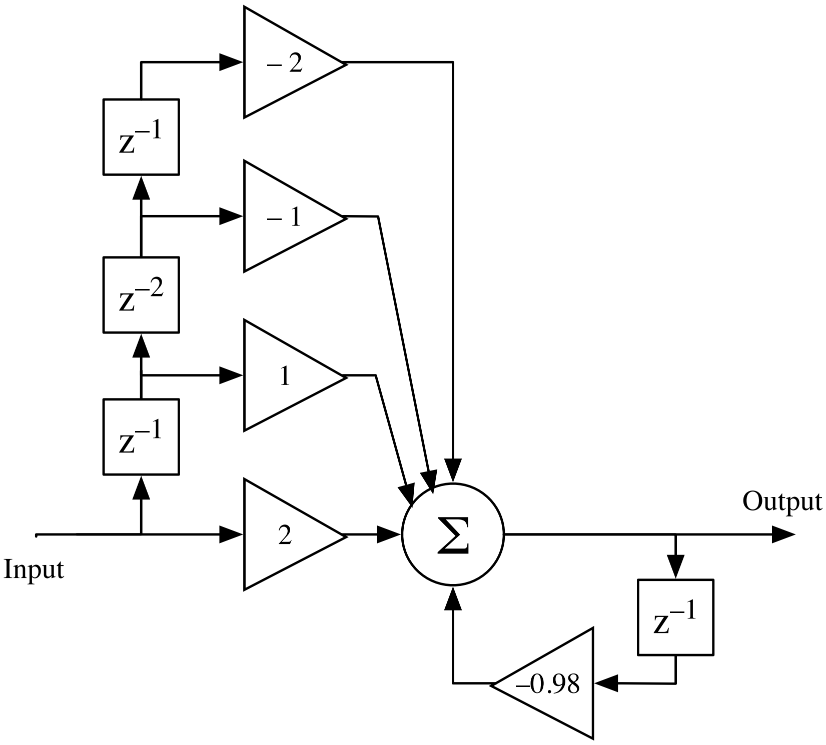


Figure 2.3 Block diagram of the ARMA filter that realizes the RASTA filtering

several experiments, the direct handling of feature trajectory is now considered as an important operation for feature extraction [Vuuren and Hermansky, 1997].

2.2 Acoustic models and training

In this section, a definition of acoustic models ($P(\boldsymbol{X}|\boldsymbol{l})$) is described. First, continuous-density hidden Markov models (CD-HMMs) are described, and then conventional training methods for CD-HMMs are described.

2.2.1 Continuous-density hidden Markov models

CD-HMM is one of the most successful models for ASR since it is suitable for handling variable length sequence in a stochastic way. In CD-HMMs, as the name implies, a hidden state sequence, which is assumed as a first-order Markov chain, is used as a hidden variable. Emission probabilities of a feature sequence $\boldsymbol{X}$, given the label sequence $\boldsymbol{l}$, are obtained by

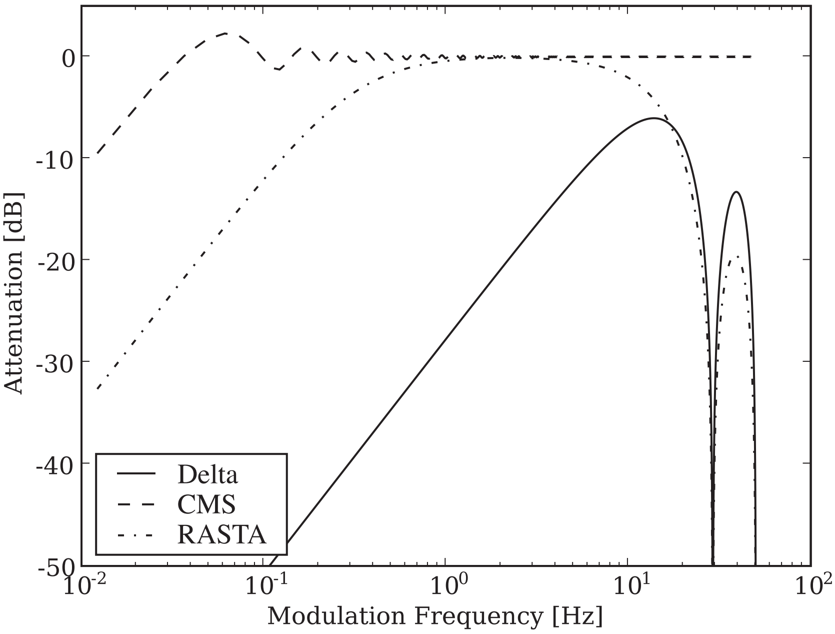


Figure 2.4 Modulation frequency characteristics of a numerical differentiation operator and the RASTA operator. CMS: Cepstral mean subtraction technique [Atal, 1974]; Delta: Differentiation operator ($T = 2$)

marginalizing out the hidden state sequence $\boldsymbol{q}$, as follows:

$$P(\boldsymbol{X}|\boldsymbol{l}) = \sum_{\boldsymbol{q}} P(\boldsymbol{q}, \boldsymbol{X}|\boldsymbol{l}) \overset{\text{def}}{=} \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} P(\boldsymbol{X}|\boldsymbol{q})P(\boldsymbol{q}) \overset{\text{def}}{=} \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \prod_n P(\boldsymbol{x}_n|q_n)P(\boldsymbol{q}). \tag{2.10}$$

Here, $\mathcal{S}(\boldsymbol{l})$ denote a set of all possible state sequences determined from the given the label sequence $\boldsymbol{l}$. Further, the hidden state sequence $\boldsymbol{q}$ is assumed as a first-order Markov chain. Therefore, $P(\boldsymbol{q})$ is defined as follows:

$$\begin{aligned} P(\boldsymbol{q}) =& \prod_n P(q_n|q_{n-1}) = \prod_n \mathcal{P}_{q_{n-1},q_n}, \\ q_0 =& s^{\texttt{BEGIN}}, q_{N(q)+1} = s^{\texttt{END}}, \end{aligned} \tag{2.11}$$

where $s^{\texttt{BEGIN}}$ and $s^{\texttt{END}}$ are constants that denote the initial state and the final state, respectively; $N(\boldsymbol{q})$, the number of elements in the state sequence $\boldsymbol{q}$; $\mathcal{P}$, a transition probability matrix. Figure 2.5 illustrates the dependency between state sequences $\boldsymbol{q}$ and a label sequence $\boldsymbol{l}$, and state sequences $\boldsymbol{q}$ and a feature sequence $\boldsymbol{X}$.

In CD-HMMs, the emission probability for each state is modeled by Gaussian mixture models (GMMs), as follows:

$$P(\boldsymbol{x}_n|q_n) \overset{\text{def}}{=} \sum_m \rho_{q_n,m}\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_{G(q_n,m)}, \boldsymbol{R}_{G(q_n,m)}). \tag{2.12}$$

Here, since the same Gaussian pdf (and its parameters) might be shared by several states and mixture components, a *many-to-one* mapping function $G(s,m)$ is introduced in order to map a state index s and a mixture component index m to a Gaussian pdf index g; $\rho_{q_n,m}$ is a mixture proportion that satisfy $\sum_m \rho_{q_n,m} = 1$; $\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_g, \boldsymbol{R}_g)$ denotes the Gaussian probability density function (pdf) parametrized by a mean vector $\boldsymbol{\mu}_g$ and a precision matrix $\boldsymbol{R}_g$ as follows:

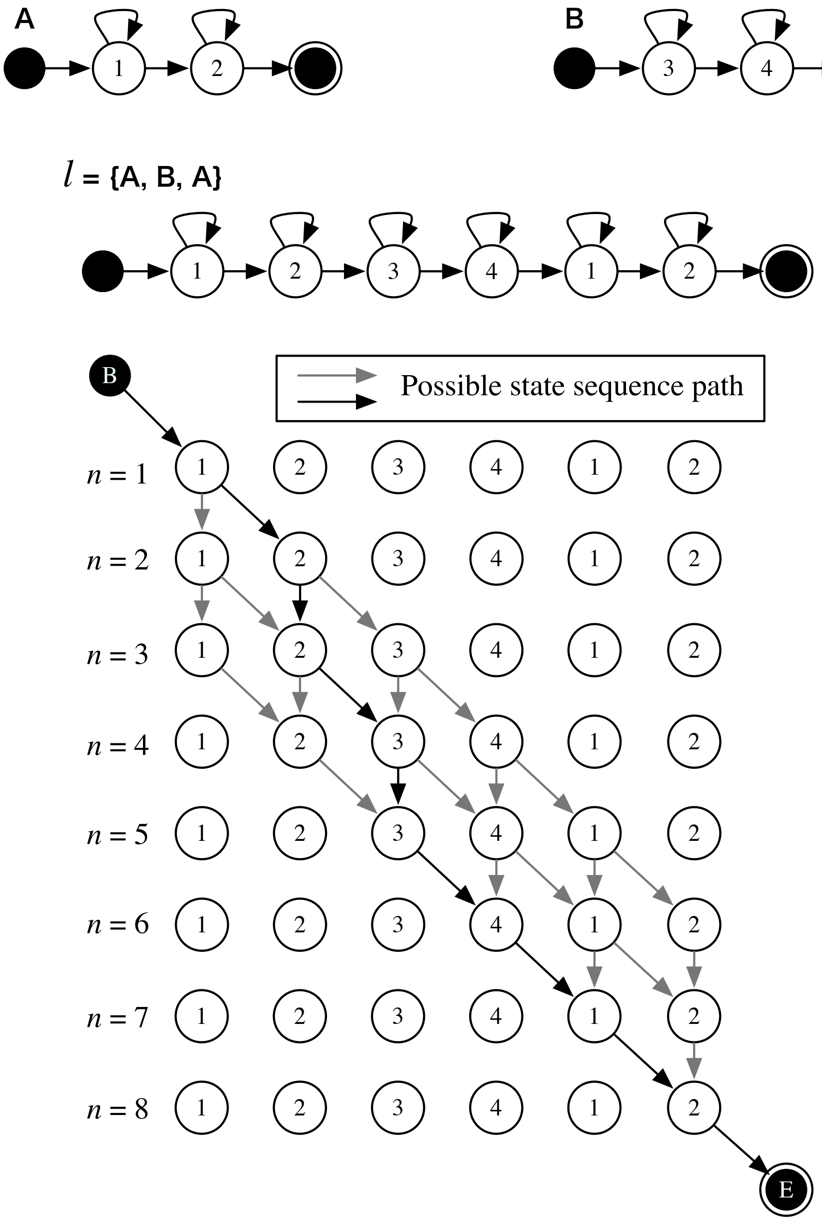
$$\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_g, \boldsymbol{R}_g) = \frac{\sqrt{|\boldsymbol{R}_g|}}{(2\pi)^{D/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}_n - \boldsymbol{\mu}_g)^{\mathsf{T}}\boldsymbol{R}_g(\boldsymbol{x}_n - \boldsymbol{\mu}_g)\right\}, \tag{2.13}$$

where D is the dimensionality of $\boldsymbol{x}_n$.

By substituting Eqs. (2.11), (2.12) and (2.13) into Eq. (2.10), the emission probability of a feature sequence $\boldsymbol{X}$ is derived as follows:

$$P(\boldsymbol{X}|\boldsymbol{l}) = \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \prod_n \mathcal{P}_{q_{n-1},q_n} \sum_m \rho_{q_n,m}\mathcal{N}(\boldsymbol{x}_n^i|\boldsymbol{\mu}_{G(q_n,m)}, \boldsymbol{R}_{G(q_n,m)}). \tag{2.14}$$

Here, in order to simplify the equation, a sequence of mixture component indices $\boldsymbol{m} =$



Example: $\boldsymbol{q} = \{1, 2, 2, 3, 3, 4, 1, 2\}$

$$P(\boldsymbol{q}) = \mathcal{P}_{\mathrm{B},1}\mathcal{P}_{1,2}\mathcal{P}_{2,2}\mathcal{P}_{2,3}\mathcal{P}_{3,3}\mathcal{P}_{3,4}\mathcal{P}_{4,1}\mathcal{P}_{1,2}\mathcal{P}_{2,\mathrm{E}}$$

$$P(\boldsymbol{X}|\boldsymbol{q}) = P(\boldsymbol{x}_1|q_1 = 1)\, P(\boldsymbol{x}_2|q_2 = 2)\, P(\boldsymbol{x}_3|q_3 = 2)\, P(\boldsymbol{x}_4|\mathrm{q}_4 = 3)\, P(\boldsymbol{x}_5|q_5 = 3) \times P(\boldsymbol{x}_6|q_6 = 4)\, P(\boldsymbol{x}_7|\mathrm{q}_7 = 1)\, P(\boldsymbol{x}_8|\mathrm{q}_8 = 1)$$

Figure 2.5 Basic idea of hidden Markov models

Table. 2.1 Notations for parameters of CD-HMMs (S: the number of all states; D: the dimensionality of feature vectors)

Notation	Description
Θ	Set of all parameters; $\Theta \overset{\text{def}}{=} \{\Lambda, \boldsymbol{\mathcal{P}}\}$
$\boldsymbol{\mathcal{P}}$	State transition probability matrix; $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{S\times S}$
Λ	Set of parameters for all emission pdfs; $\Lambda \overset{\text{def}}{=} \{\lambda_s \| \forall s\}$
λ_s	Set of parameters for the emission pdf at state s; $\lambda_s \overset{\text{def}}{=} \{\rho_{s,m}, \boldsymbol{\mu}_{G(s,m)}, \boldsymbol{R}_{G(s,m)} \| \forall s, \forall m\}$
$\rho_{s,m}$	Mixture proportion; s.t. $\rho_{s,m} > 0, \sum_m \rho_{s,m} = 1$
$\boldsymbol{\mu}_g$	Mean vector for the g^{th} Gaussian pdf,$\boldsymbol{\mu}_g \in \mathbb{R}^D$
$\boldsymbol{R}_g$	Precision matrix for the g^{th} Gaussian pdf, $\boldsymbol{R}_g \in \mathbb{R}^{D\times D}$

$\{m_1, m_2, \cdots, m_n, \cdots\}$ is introduced as follows:

$$\begin{aligned}P(\boldsymbol{X}|\boldsymbol{l}) =& \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} \prod_n \mathcal{P}_{q_{n-1},q_n} \rho_{q_n,m_n} \mathcal{N}(\boldsymbol{x}_n^i | \boldsymbol{\mu}_{G(q_n,m_n)}, \boldsymbol{R}_{G(q_n,m_n)})\\ =&P(\boldsymbol{q}, \boldsymbol{m}|\boldsymbol{l}) \underbrace{\prod_n \mathcal{N}(\boldsymbol{x}_n^i | \boldsymbol{\mu}_{G(q_n,m_n)}, \boldsymbol{R}_{G(q_n,m_n)})}_{P(\boldsymbol{X}|\boldsymbol{q},\boldsymbol{m},\boldsymbol{l})}\\ =& \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} P(\boldsymbol{X}, \boldsymbol{q}, \boldsymbol{m}|\boldsymbol{l}).\end{aligned} \tag{2.15}$$

This expression explicitly shows that CD-HMMs involve two hidden variables, i.e., state sequence $\boldsymbol{q}$ and mixture component sequence $\boldsymbol{m}$. The parameters of CD-HMMs are listed in Table 2.1. The following subsections describe the estimation methods for these parameters.

2.2.2 Maximum likelihood estimation

Hereinafter, let $\mathcal{X} = \{\boldsymbol{X}^1, \boldsymbol{X}^2, \cdots, \boldsymbol{X}^i \cdots\}$ be a set of feature sequences in a training dataset, and $\mathcal{L} = \{\boldsymbol{l}^1, \boldsymbol{l}^2, \cdots, \boldsymbol{l}^i \cdots\}$ be a set of label sequences in the training dataset.

The most widely accepted method to estimate parameters is the maximum likelihood estimation (MLE). In the MLE, a parameter set $\hat{\Theta}$ that maximizes the log-likelihood $\mathcal{F}^{\texttt{MLE}}(\Theta; \mathcal{X}, \mathcal{L})$ of the training dataset $\mathcal{X}$, $\mathcal{L}$, is estimated as follows:

$$\begin{aligned}\hat{\Theta} =&\underset{\Theta}{\operatorname{argmax}} \log P(\mathcal{X}, \mathcal{L}|\Theta) = \underset{\Theta}{\operatorname{argmax}} \log \prod_i P(\boldsymbol{X}^i, \boldsymbol{l}^i|\Theta)\\ =&\underset{\Theta}{\operatorname{argmax}} \underbrace{\sum_i \log P(\boldsymbol{X}^i|\boldsymbol{l}^i, \Theta)}_{\mathcal{F}^{\texttt{MLE}}(\Theta;\mathcal{X},\mathcal{L})} + \underbrace{\sum_i \log P(\boldsymbol{l}^i)}_{\text{constant}}\end{aligned} \tag{2.16}$$

Because the optimal of the objective function $\mathcal{F}^{\mathtt{MLE}}(\Theta;\mathcal{X},\mathcal{L})$ cannot be derived analytically where the CD-HMMs are used as acoustic models, lower bounds of the objective function, called auxiliary function, are introduced in order to perform the optimization. As defined in Eq. (2.15), the emission probability of sequences is defined by using hidden variables; $\boldsymbol{q}$ and $\boldsymbol{m}$. Substituting Eq. (2.15) into Eq. (2.16) yields the following objective function.

$$\begin{aligned}
&\mathcal{F}^{\mathtt{MLE}}(\Theta;\mathcal{X},\mathcal{L})\\
=&\sum_i \log \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} P(\boldsymbol{X}^i,\boldsymbol{q},\boldsymbol{m}|\boldsymbol{l}^i,\Lambda) + \text{constant}\\
=&\sum_i \log \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} \prod_n \mathcal{P}_{q_{n-1},q_n} \rho_{q_n,m_n} \mathcal{N}(\boldsymbol{x}^i_n|\boldsymbol{\mu}_{G(q_n,m_n)},\boldsymbol{R}_{G(q_n,m_n)}) + \text{constant}.
\end{aligned} \tag{2.17}$$

In order to obtain a lower bound of the objective function, a probabilistic distribution over hidden variables $Q(\boldsymbol{q},\boldsymbol{m};\Theta')$ parametrized by Θ' is introduced, and then Jensen's inequality is applied as follows:

$$\begin{aligned}
&\mathcal{F}^{\mathtt{MLE}}(\Theta;\mathcal{X},\mathcal{L})\\
=&\sum_i \log \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m};\Theta') \frac{\prod_n \rho_{q_n,m_n}\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_{G(q_n,m_n)},\boldsymbol{R}_{G(q_n,m_n)})\mathcal{P}_{q_{n-1},q_n}}{Q(\boldsymbol{q},\boldsymbol{m};\Theta')}\\
&\quad + \text{constant}\\
\geq&\sum_i \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m};\Theta') \log \frac{\prod_n \rho_{q_n,m_n}\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_{G(q_n,m_n)},\boldsymbol{R}_{G(q_n,m_n)})\mathcal{P}_{q_{n-1},q_n}}{Q(\boldsymbol{q},\boldsymbol{m};\Theta')}\\
&\quad + \text{constant}\\
=&\sum_i \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m};\Theta') \log \prod_n \rho_{q_n,m_n}\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_{q_n m_n},\boldsymbol{R}_{q_n m_n})\mathcal{P}_{q_{n-1},q_n}\\
&\quad \underbrace{- \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m};\Theta') \log Q(\boldsymbol{q},\boldsymbol{m};\Theta')}_{\text{constant}} + \text{constant}\\
=&\sum_i \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} \sum_n\\
&\quad Q(\boldsymbol{q},\boldsymbol{m};\Theta')\left(\log\rho_{q_n,m_n} + \log\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_{G(q_n,m_n)},\boldsymbol{R}_{G(q_n,m_n)}) + \log\mathcal{P}_{q_{n-1},q_n}\right)\\
&\qquad + \text{constant}\\
\overset{\text{def}}{=}&\tilde{\mathcal{F}}^{\mathtt{MLE}}(\Theta;\mathcal{X},\mathcal{L},\Theta').
\end{aligned} \tag{2.18}$$

The optimum of the auxiliary function $\tilde{\mathcal{F}}^{\mathtt{MLE}}(\Theta;\mathcal{X},\mathcal{L},\Theta')$ with given Θ' can be solved analytically.

It should be noted that the auxiliary function $\tilde{\mathcal{F}}^{\text{MLE}}$ touches the original function $\mathcal{F}^{\text{MLE}}$ at Θ', where Q is set as the posterior pdf of hidden variables, i.e. $Q(\boldsymbol{q},\boldsymbol{m};\Theta') \overset{\text{def}}{=} P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')$, as follows:

$$
\begin{aligned}
&\tilde{\mathcal{F}}^{\text{MLE}}(\Theta;\mathcal{X},\mathcal{L},\Theta')\mid_{\Theta=\Theta'}\\
=&\sum_i\sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)}\sum_{\boldsymbol{m}}P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')\log\frac{P(\boldsymbol{X}^i,\boldsymbol{q},\boldsymbol{m}|\boldsymbol{l}^i,\Theta')}{P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')}+\sum_i\log P(\boldsymbol{l}^i)\\
=&\sum_i\sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)}\sum_{\boldsymbol{m}}P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')\log\frac{P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')P(\boldsymbol{X}^i|\boldsymbol{l}^i,\Theta')}{P(\boldsymbol{q},\boldsymbol{m}|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')}+\sum_i\log P(\boldsymbol{l}^i)\\
=&\sum_i\log P(\boldsymbol{X}^i|\boldsymbol{l}^i,\Theta')+\sum_i\log P(\boldsymbol{l}^i)=\mathcal{F}^{\text{MLE}}(\Theta;\mathcal{X},\mathcal{L}).
\end{aligned}
\tag{2.19}
$$

Since this lower bound touches the original objective function $\mathcal{F}^{\text{MLE}}(\Theta;\mathcal{X},\mathcal{L})$, iterative update of Θ converges a local optimum of the original optimization function. Specifically, the current estimate of Θ is used to determine $Q(\boldsymbol{q},\boldsymbol{m}|\Theta)$, and then Θ is updated by using the current setting of Q. The iterative optimization algorithm based on this setting is termed *Expectation-Maximization* (EM) algorithm. By using this Q function, the auxiliary function is decomposed into each frame, as follows:

$$
\begin{aligned}
\tilde{\mathcal{F}}(\Theta;\mathcal{X},\mathcal{L},\Theta')=&\sum_i\sum_n\sum_{s,m}P(q_n=s,m_n=m|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')\log\mathcal{N}(X_n|\boldsymbol{\mu}_{G(s,m)},\boldsymbol{R}_{G(s,m)})\\
&+\sum_i\sum_n\sum_s P(q_n=s,m_n=m|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')\log\rho_{q_n,m_n}\\
&+\sum_i\sum_n\sum_{s,s'}P(q_{n-1}=s,q_n=s'|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')\log\mathcal{P}_{s,s'},
\end{aligned}
\tag{2.20}
$$

where

$$
\begin{aligned}
P(q_n=s,m_n=m|\boldsymbol{X},\boldsymbol{l},\Theta')&=\sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})}\sum_{\boldsymbol{m}}P(q_n=s,m_n=m|\boldsymbol{X},\boldsymbol{l},\Theta'),\\
P(q_{n-1}=s,q_n=s'|\boldsymbol{X},\boldsymbol{l},\Theta')&=\sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})}P(q_{n-1}=s',q_n=s|\boldsymbol{X},\boldsymbol{l},\Theta').
\end{aligned}
\tag{2.21}
$$

In order to efficiently perform the EM algorithm, the forward-backward algorithm is used in general. In this algorithm, each HMM state occupation probability $P(q_n=s|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')$ is computed as a product of a forward probability $(\exp\alpha^i_{n,s})$ and a backward probability $(\exp\beta^i_{n,s})$, as follows:

$$
P(q_n=s|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')=\exp(\alpha^i_{n,s}+\beta^i_{n,s}-\beta^i_{0,s^{\text{BEGIN}}})
\tag{2.22}
$$

where

$$\begin{aligned}
\alpha^i_{n,s} =& \log\left\{\sum_{s'} \exp\left\{\alpha_{(n-1),s'}\right\} P(\boldsymbol{x}^i_{n-1}|q_{n-1}=s',\Theta')\mathcal{P}'_{s',s}\right\}, \\
\beta^i_{n,s} =& \log\left\{\sum_{s'} \exp\left\{\beta^i_{(n+1),s'}\right\} P(\boldsymbol{x}^i_{n}|q_{n}=s,\Theta')\mathcal{P}'_{s,s'}\right\}, \\
\alpha^i_{1,s^{\texttt{BEGIN}}} =& 0.0, \\
\beta^i_{N(\boldsymbol{X}^i)+1,s^{\texttt{END}}} =& 0.0,
\end{aligned} \tag{2.23}$$

where $\mathcal{P}'_{s',s}$ is a state transition probability matrix in the parameter set Θ'.

From the definition of CD-HMMs, pdfs depending on Θ' in Eq. (2.20) can be computed as follows:

$$\begin{aligned}
P(q_n=s,m_n=m|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta') =& P(m_n=m|\boldsymbol{X}^i,\boldsymbol{l}^i,q_n=s)P(q_n=s|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta') \\
=& \frac{\mathcal{N}(\boldsymbol{x}^i_n|\boldsymbol{\mu}'_{G(s,m)},\boldsymbol{R}'_{G(s,m)})}{\sum_{m'}\mathcal{N}(\boldsymbol{x}^i_n|\boldsymbol{\mu}'_{G(s,m')},\boldsymbol{R}'_{G(s,m')})}\exp\left\{\alpha^i_{n,s}+\beta^i_{n,s}-\beta^i_{0,s^{\texttt{BEGIN}}}\right\} \\
P(q_{n-1}=s,q_n=s'|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta') =& P(q_{n-1}=s'|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')P(q_n=s|\boldsymbol{X}^i,\boldsymbol{l}^i,\Theta')
\end{aligned} \tag{2.24}$$

where $\boldsymbol{\mu}'_g$ and $\boldsymbol{R}'_g$ are parameters in the parameter set Θ'. Note that these variables ($\alpha^i_{s,m}$, $\beta^i_{s,m}$) can be computed efficiently by using a recursive procedure. By using these variables (α,β), the optimal point of the auxiliary function $\mathcal{F}^{\text{MLE}}(\Theta;\mathcal{X},\mathcal{L},\Theta')$ can be expressed as follows:

$$\begin{aligned}
\hat{\boldsymbol{\mu}}_g =& \frac{\sum_i \boldsymbol{\chi}^1_g(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}{\sum_i \chi^0_g(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}, \\
\hat{\boldsymbol{R}}_g =& \hat{\boldsymbol{\Sigma}}_g^{-1}, \\
\hat{\boldsymbol{\Sigma}}_n =& \frac{\sum_i \boldsymbol{\chi}^2_g(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}{\sum_i \chi^0_g(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')} - \hat{\boldsymbol{\mu}}_g\hat{\boldsymbol{\mu}}_g^{\mathsf{T}}, \\
\hat{\boldsymbol{\rho}}_{s,m} =& \frac{\sum_i \chi^0_{G(s,m)}(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}{\sum_i\sum_{m'} \chi^0_{G(s,m')}(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}, \\
\hat{\mathcal{P}}_{s,s'} =& \frac{\sum_i \chi^{\text{TR}}_{s,s'}(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')}{\sum_{\tilde{s}}\sum_i \chi^{\text{TR}}_{s,\tilde{s}}(\boldsymbol{X}^i,\boldsymbol{l}^i;\Theta')},
\end{aligned} \tag{2.25}$$

where the functions χ^0_g, $\boldsymbol{\chi}^1_g$, $\boldsymbol{\chi}^2_g$, $\boldsymbol{\chi}^{\text{TR}}_{s,s'}$ are termed as “sufficient statistics function” in this thesis, and expressed as follows:

$$\begin{aligned}
\chi_g^0(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta') &\stackrel{\text{def}}{=} \sum_{n=1}^{N(\boldsymbol{X}^i)} P(q_n = s, m_n = m | \boldsymbol{X}^i, \boldsymbol{l}^i, \Theta'), \\
\boldsymbol{\chi}_g^1(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta') &\stackrel{\text{def}}{=} \sum_{n=1}^{N(\boldsymbol{X}^i)} P(q_n = s, m_n = m | \boldsymbol{X}^i, \boldsymbol{l}^i, \Theta') \boldsymbol{x}_n^i, \\
\boldsymbol{\chi}_g^2(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta') &\stackrel{\text{def}}{=} \sum_{n=1}^{N(\boldsymbol{X}^i)} P(q_n = s, m_n = m | \boldsymbol{X}^i, \boldsymbol{l}^i, \Theta') \boldsymbol{x}_n^i (\boldsymbol{x}_n^i)^{\mathsf{T}}, \\
\chi_{s,s'}^{\mathtt{TR}}(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta') &\stackrel{\text{def}}{=} \sum_{n=2}^{N(\boldsymbol{X}^i)} P(q_n = s, q_{n-1} = s' | \boldsymbol{X}^i, \boldsymbol{l}^i, \Theta').
\end{aligned} \tag{2.26}$$

Since all probability $P(.)$ used in the above definition of sufficient statistics functions can be computed by using the forward-backward algorithm as shown in Eq. (2.24), the computation of sufficient statistics functions is tractable.

2.2.3 Bayesian inference

Recently, Bayesian inference is introduced as a generalization of MLE that enables diverse extensions. In Bayesian inference, the optimal parameters are not determined but probabilistic distributions of parameters are inferred. This thesis avoids detailed discussions about Bayesian inference, and just introduces a basic idea of distribution-based expression of parameters.

By using Bayes' law, a distribution (posterior pdf) of parameter set Θ, given the obtained dataset $\mathcal{X}, \mathcal{L}$, is written as follows:

$$P(\Theta | \mathcal{X}, \mathcal{L}) = \frac{P(\mathcal{X}, \mathcal{L} | \Theta) P^0(\Theta)}{P(\mathcal{X}, \mathcal{L})} \tag{2.27}$$

where $P^0(\Theta)$ is a prior pdf that reflects the prior belief of the parameters [*1]. Here, the *maximum-a-posteriori* (MAP) estimation is derived by using the mode-value of the posterior pdf as a representative parameter, which is obtained as follows:

$$\hat{\Theta}^{\mathtt{MAP}} = \underset{\Theta}{\text{argmax}} P(\Theta | \mathcal{X}, \mathcal{L}) = \underset{\Theta}{\text{argmax}} P(\mathcal{X}, \mathcal{L} | \Theta) P^0(\Theta). \tag{2.28}$$

Further, the MLE is derived by introducing a uniform pdf to MAP estimation as follows:

$$\hat{\Theta}^{\mathtt{MLE}} = \underset{\Theta}{\text{argmax}} P(\mathcal{X}, \mathcal{L} | \Theta). \tag{2.29}$$

[*1] In this thesis, prior pdfs are distinguished by using P^0 notation.

Thus, the Bayesian inference is considered as a generalization of these estimation methods.

Advantages of the Bayesian inference are listed as follows:

- Utilization of a prior belief of the parameters, and
- Robust classification performed by marginalizing (integrating) all possible parameters over a posterior pdf (a.k.a. Bayesian predictive classification).

In the case of CD-HMM parameter inference, the posterior pdf $P(\Theta|\mathcal{X}, \mathcal{L})$ is not tractable. The variational approximation methods are often introduced to realize Bayesian methods in such cases.

In the variational Bayesian methods [Attias, 2000, Watanabe, 2006], an approximated posterior pdf $\tilde{P}(\Theta)$, which is restricted to a specific pdf family, is obtained by minimizing the Kullback-Leibler divergence (KL divergence) of an approximated posterior pdf $\tilde{P}(\Theta)$ from the true posterior pdf $P(\Theta|\mathcal{X}, \mathcal{L})$. Thus, the approximated posterior pdf is obtained as follows:

$$\tilde{P}(\Theta) = \underset{P(\Theta)}{\operatorname{argmax}} \underbrace{\mathrm{KL}[P(\Theta)||P(\Theta|\mathcal{X}, \mathcal{L})]}_{\mathcal{F}^{\mathrm{VB}}[\tilde{P}(\Theta)]} \tag{2.30}$$

where $P(\Theta|\mathcal{X}, \mathcal{L})$ is the true posterior pdf defined in Eq. (2.27), and the KL divergence is defined as follows:

$$\mathrm{KL}[\tilde{P}(\Theta)||P(\Theta|\mathcal{X}, \mathcal{L})] = \left\langle \log \tilde{P}(\Theta) - P(\Theta|\mathcal{X}, \mathcal{L}) \right\rangle_{\tilde{P}(\Theta)}. \tag{2.31}$$

It is known that the Bayesian inference is advantageous, even if the training data is limited.

2.2.4 Discriminative training

The models obtained by the Bayesian inference or its specialized methods (MLE/ MAP) are accurate in the sense of statistic generative models. However, because the ultimate objective of model estimation for ASR is construction of classifiers, several discriminative training methods are proposed in order to optimize classification performance of constructed classifiers.

In discriminative training methods, model parameters are estimated by optimizing a discriminative criterion function. Several criteria for discriminative training are proposed.

Minimum classification error (MCE) attempts to minimize the number of misclassification label sequences [Juang and Katagiri, 1992]. Because the number of misclassification is denoted by discrete value, a smoothed misclassification number function is often used as an optimization criterion.

Ideally, the number of misclassification (error) of a classifier $\mathcal{E}(\Theta)$ defined by a parameter

Θ is represented as follows:

$$\begin{aligned}\mathcal{E}(\Theta) &= \sum_i \left(1 - \mathcal{H}\left(\log \frac{P(\boldsymbol{X}^i, \boldsymbol{l}^i|\Theta)}{\max_{\boldsymbol{l}} P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)}\right)\right) \\ &= \underbrace{N(\mathcal{X})}_{\text{constant}} - \underbrace{\sum_i \mathcal{H}\left(\log P(\boldsymbol{X}^i, \boldsymbol{l}^i|\Theta) - \max_{\boldsymbol{l}} \log P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)\right)}_{\mathcal{F}^{\texttt{MCE}}(\Theta;\mathcal{X},\mathcal{L})}\end{aligned} \tag{2.32}$$

where $\mathcal{H}(x)$ is the Heaviside-step function that returns 1 when $x > 0$ and 0 otherwise, $N(\mathcal{X})$ is the number of training sequences.

MCE training is aiming for minimizing this error function by maximizing the objective function $\mathcal{F}^{\texttt{MCE}}(\Theta; \mathcal{X}, \mathcal{L})$. The objective function is discontinuous because it includes Heaviside-step function $\mathcal{H}$ and $\max$ function. Hence, in order to perform optimization by using gradient-based optimization methods, a smoothed objective function $\tilde{\mathcal{F}}^{\texttt{MCE}}(\Theta; \mathcal{X}, \mathcal{L})$ is introduced by using a smoothed Heaviside-step function $\tilde{\mathcal{H}}$ and a $\mathrm{softmax}$ function, as follows:

$$\tilde{\mathcal{F}}^{\texttt{MCE}}(\Theta; \mathcal{X}, \mathcal{L}) = \sum_i \tilde{\mathcal{H}}\left(\log P(\boldsymbol{X}^i, \boldsymbol{l}^i|\Theta) - \mathrm{softmax}_{\boldsymbol{l} \neq \boldsymbol{l}^i} \log\left(P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)\right)\right) \tag{2.33}$$

where $\mathrm{softmax}_{\boldsymbol{l} \neq \boldsymbol{l}^i} \left\{P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)\right\}$ is often defined as follows:

$$\mathrm{softmax}_{\boldsymbol{l} \neq \boldsymbol{l}^i} \left\{\log P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)\right\} \overset{\text{def}}{=} \frac{1}{\eta} \log \sum_{\boldsymbol{l} \neq \boldsymbol{l}^i} \left(P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)\right)^{\eta}, \tag{2.34}$$

where η is a hyper-parameter that controls the approximate accuracy of the $\mathrm{softmax}$ function. Several functions are used as smoothed Heaviside-step functions $\tilde{\mathcal{H}}$. For example, linear function, sigmoid function, or piecewise linear function are used. A smoothed Heaviside-step function is termed as "loss function."

Maximum mutual information estimation (MMIE) is performed by maximizing mutual information between the feature vector sequence variable $\boldsymbol{X}$ and the label sequence variable $\boldsymbol{l}$ [Bahl et al., 1986]. Mutual information to be maximized is defined as follows:

$$\begin{aligned}I[\boldsymbol{X}; \boldsymbol{l}] &\overset{\text{def}}{=} \underbrace{H[P(\boldsymbol{l})]}_{\text{constant}} - H[P(\boldsymbol{l}|\boldsymbol{X}, \Theta)] \\ &= \text{constant} - \underbrace{\langle -\log P(\boldsymbol{l}|\boldsymbol{X}, \Theta)\rangle_{P(\boldsymbol{l}|\boldsymbol{X},\Theta)}}_{\text{expectation}}\end{aligned} \tag{2.35}$$

Here, by approximating the expectation operator by the empirical average computed from the training dataset, mutual information is approximated as follows:

$$\begin{aligned}
I[\boldsymbol{X};\boldsymbol{l}] \approx & \text{constant} + \underbrace{\frac{1}{N(\mathcal{L})}}_{\text{constant}} \sum_i \log P(\boldsymbol{l}^i|\boldsymbol{X}^i) \\
= & \text{constant} + \text{constant} \times \sum_i \log \frac{P(\boldsymbol{l}^i, \boldsymbol{X}^i)}{P(\boldsymbol{X}^i)} \\
= & \text{constant} + \text{constant} \times \sum_i \log \frac{P(\boldsymbol{l}^i, \boldsymbol{X}^i)}{\sum_{\boldsymbol{l}} P(\boldsymbol{l}, \boldsymbol{X}^i)} \\
= & \text{constant} + \underbrace{\text{constant} \times \sum_i \log \frac{P(\boldsymbol{X}^i|\boldsymbol{l}^i)P(\boldsymbol{l}^i)}{\sum_{\boldsymbol{l}} P(\boldsymbol{X}^i|\boldsymbol{l})P(\boldsymbol{l})}}_{\mathcal{F}^{\texttt{MMI}}(\Theta;\mathcal{X},\mathcal{L})}
\end{aligned} \tag{2.36}$$

Although MMIE is not explicitly aiming for reduction of classification errors, MMIE is also considered as a discriminative training method since error hypotheses $\boldsymbol{l} \neq \boldsymbol{l}^i$ are also considered in the objective function.

Minimum phone error (MPE) is performed by maximizing the expectation of a phone accuracy function [Povey and Woodland, 2002]. As contrasted to the MCE training that is aiming for minimization of sequence errors, MPE training is aiming for minimization of phone errors.

MPE training estimates the parameters so that the expectation of phone accuracy function $A(\boldsymbol{l}, \boldsymbol{l}^i)$ over the sequence posterior pdf $P(\boldsymbol{l}|\boldsymbol{X}, \Theta)$ is maximized. The expectation to be maximized is expanded as follows:

$$\begin{aligned}
\sum_i \langle A(\boldsymbol{l}, \boldsymbol{l}^i) \rangle_{P(\boldsymbol{l}|\boldsymbol{X}^i,\Theta)} = & \sum_i \sum_{\boldsymbol{l}} P(\boldsymbol{l}|\boldsymbol{X}^i, \Theta) A(\boldsymbol{l}, \boldsymbol{l}^i) \\
= & \sum_i \sum_{\boldsymbol{l}} \frac{P(\boldsymbol{l}, \boldsymbol{X}^i|\Theta)}{P(\boldsymbol{X}^i|\Theta)} A(\boldsymbol{l}, \boldsymbol{l}^i) \\
= & \sum_i \sum_{\boldsymbol{l}} \frac{P(\boldsymbol{l}, \boldsymbol{X}^i|\Theta)}{\sum_{\boldsymbol{l}'} P(\boldsymbol{X}^i, \boldsymbol{l}'|\Theta)} A(\boldsymbol{l}, \boldsymbol{l}^i) \\
= & \underbrace{\sum_i \frac{\sum_{\boldsymbol{l}} P(\boldsymbol{l}, \boldsymbol{X}^i|\Theta) A(\boldsymbol{l}, \boldsymbol{l}^i)}{\sum_{\boldsymbol{l}} P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)}}_{\mathcal{F}^{\texttt{MPE}}(\Theta;\mathcal{X},\mathcal{L})}
\end{aligned} \tag{2.37}$$

By using this expanded objective function $\mathcal{F}^{\texttt{MPE}}(\Theta; \mathcal{X}, \mathcal{L})$ and an appropriate approximation method of phone accuracy function $A(\boldsymbol{l}, \boldsymbol{l}^i)$ described in [Povey and Woodland, 2002], the objective function can be maximized by using the extended Baum-Welch algorithm [Woodland, 2002].

2.3 Hidden Markov models/ multi-layer perceptron tandem approach

Another approach to improve classification performance of HMMs is HMM/MLP-tandem approach [Hermansky et al., 2000]. Because the discriminative training methods are based on standard CD-HMMs, nonlinearities of classification are limited by a choice of emission pdf family.

In tandem-approach, nonlinearities in feature vectors are resolved by using MLP-based monophone classifiers, and then classification result sequences are decoded by using standard CD-HMMs. Because MLPs can efficiently represent nonlinear classification boundaries when compared with GMM-based emission pdfs, the use of MLPs is effective. Figure 2.6 shows a block diagram of HMM/MLP-tandem systems. Recently, HMM/MLP-tandem approaches are used in order to cope with high-dimensional features [Morgan et al., 2005, Chen et al., 2005].

By introducing the frame-level label sequence l_n^i $(n = [1..N(\boldsymbol{X}^i)])$, the optimal trans-

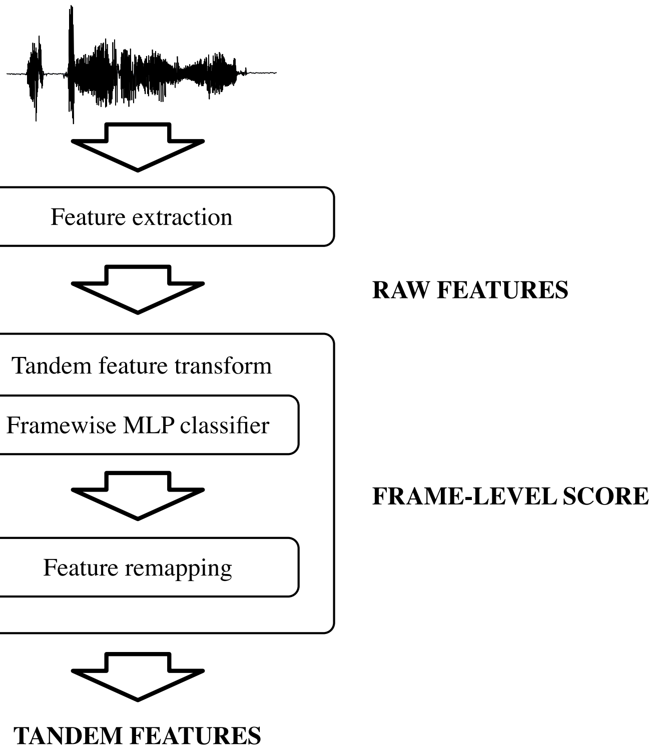


Figure 2.6 The block diagram of HMM/MLP-tandem based frontend

formed features $\hat{\boldsymbol{x}}_n^i$ are defined as follows:

$$\hat{\boldsymbol{x}}_n^i \overset{\text{def}}{=} \text{vec}\{\mathbb{1}(l_n^i, l)|\forall l\} \tag{2.38}$$

where l_n^i is a element in the set of monophones, vec is the vectorizing function which construct a vector from the given predicate, $\mathbb{1}$ is the Kronecker delta function denoted as follows:

$$\mathbb{1}(x, y) = \begin{cases} 1 & x = y, \\ 0 & \text{otherwise.} \end{cases} \tag{2.39}$$

In general, frame-level labels l_n^i are obtained by an HMM force alignment method.

In order to realize the above transformation for an input raw feature vector $\boldsymbol{x}$, a transform function $\boldsymbol{\mathcal{T}}$ parametrized by $\hat{\Xi}$ is used as follows:

$$\boldsymbol{x}^{\texttt{MLP}} \overset{\text{def}}{=} \boldsymbol{\mathcal{T}}(\boldsymbol{x}; \hat{\Xi}) \tag{2.40}$$

where $\hat{\Xi}$ is estimated so that the squared error over the training dataset is minimized, as follows:

$$\hat{\Xi} = \underset{\Xi}{\text{argmin}} \sum_i \sum_{n=1}^{N(X^i)} ||\boldsymbol{\mathcal{T}}(\boldsymbol{x}_n^i; \Xi) - \hat{\boldsymbol{x}}_n^i||^2 \tag{2.41}$$

Here, a multi-layer perceptron (MLP) is introduced as a transform function in Eqs. (2.40) and (2.41). The transform function defined by an MLP is expressed as follows:

$$\begin{aligned} \boldsymbol{\mathcal{T}}(\boldsymbol{x}; \Xi) =& \boldsymbol{a}_U(\boldsymbol{x}; \Xi) \\ \boldsymbol{a}_0(\boldsymbol{x}; \Xi) =& \boldsymbol{x} \\ \boldsymbol{a}_u(\boldsymbol{x}; \Xi) =& \boldsymbol{\zeta}_u(\boldsymbol{C}_u \boldsymbol{a}_{u-1}(\boldsymbol{x}) + \boldsymbol{b}_u) \end{aligned} \tag{2.42}$$

where $\Xi = \{\boldsymbol{C}_u, \boldsymbol{b}_u | u = [1..U]\}$, U is the number of layers in MLP as a hyper parameter, and $\boldsymbol{\zeta}_u$ is the nonlinear vector warping function at u^{th} layer. The optimization is efficiently solved by using the back-propagation algorithm [Rumelhart and McClelland, 1986].

In general, element-wise sigmoid function $\boldsymbol{\varsigma}$ is used as $\boldsymbol{\zeta}_u$ for all u, denoted as follows:

$$\begin{aligned} \boldsymbol{\zeta}_u(\boldsymbol{x}) \overset{\text{def}}{=}& \boldsymbol{\varsigma}(\boldsymbol{x}) \quad \forall u, \\ \boldsymbol{\varsigma}(\boldsymbol{x}) =& [\varsigma(x_1), \varsigma(x_2), \cdots]^{\mathsf{T}}, \\ \varsigma(x_d) =& \frac{1}{1 + \exp(-x_d)}, \end{aligned} \tag{2.43}$$

where d is an index of dimensionality.

Because the optimal transformed feature vector $\hat{\boldsymbol{x}}_n^i$ is a binary vector, the distribution of $\boldsymbol{x}^{\texttt{MLP}}$, which is trained to approximate $\hat{\boldsymbol{x}}_n^i$, is also skew. Therefore, GMMs used in conventional CD-HMMs are not suitable as models of $\boldsymbol{x}^{\texttt{MLP}}$. In order to adapt the features $\boldsymbol{x}^{\texttt{MLP}}$ to the

GMMs, feature remapping operations are used. Conventionally, elementwise logarithm functions followed by Karhunen-Loeve transformation (KLT) is used to adapt the MLP-output vectors into GMMs. Hence tandem features $\boldsymbol{x}^{\texttt{TANDEM}}$ are typically obtained as follows:

$$\boldsymbol{x}_n^{\texttt{TANDEM}} = \mathsf{K}^{\texttt{MLP}} \cdot \left(\mathbf{log}\left(\boldsymbol{x}_n^{\texttt{MLP}}\right)\right) \tag{2.44}$$

where $\mathbf{log}$ denotes a multivariate function that applies logarithm for each element, and $\mathsf{K}^{\texttt{MLP}}$ is a KLT projection matrix, which is obtained as follows:

$$\begin{aligned} \mathsf{K}^{\texttt{MLP}} &\stackrel{\text{def}}{=} \underset{\boldsymbol{K}}{\operatorname{argmax}} \left| \sum_i \sum_n (\boldsymbol{K}\mathbf{log}\left\{\boldsymbol{x}_n^{i,\texttt{MLP}}\right\} - \boldsymbol{K}\bar{\boldsymbol{x}})(\boldsymbol{K}\mathbf{log}\left\{\boldsymbol{x}_n^{i,\texttt{MLP}}\right\} - \boldsymbol{K}\bar{\boldsymbol{x}})^T \right|, \\ \bar{\boldsymbol{x}} &\stackrel{\text{def}}{=} \frac{\sum_i \sum_n \mathbf{log}\left\{\boldsymbol{x}_n^{i,\texttt{MLP}}\right\}}{\sum_i N(\boldsymbol{X}^i)}, \end{aligned} \tag{2.45}$$

where K is restricted to be a orthonormal matrix.

Finally, the tandem feature vectors $\boldsymbol{x}_n^{\texttt{TANDEM}}$ are obtained from input vectors $\boldsymbol{x}$ and the training dataset $\mathcal{X}, \mathcal{L}$. Although the tandem methods include MLP-classifiers in their formulation, the tandem-approach is also considered as a feature extraction process that can effectively deal with high-dimensional features.

Chapter 3

High-dimensional features based on frequency modulation of speech

In order to construct high-dimensional representations of speech signals, this chapter describes novel speech features based on frequency modulation of speech signals. The efficiency of the proposed speech features is confirmed by carrying out reverberant speech recognition experiments and noisy speech recognition experiments. The motivation, approach, and previous studies are discussed in Section 3.1. The proposed method is presented in Section 3.2. The conventional ensemble classification method of high-dimensional features is introduced and described in Section 3.3. The experimental setup and experimental results are presented in Section 3.4.

3.1 Introduction

To utilize automatic speech recognition (ASR) systems in realistic environments, their robustness to environmental effects, including the presence of additive noise and/or multiplicative noise, is important. Although these effects certainly damage acoustical features, the accuracy of human speech recognition is not degraded as much as that of ASR [Lippmann, 1997]. This deficit in speech recognition can be prevented by enhancing acoustical features by using multiple feature streams and temporal information.

In general, it is considered that in human speech recognition, multiple acoustical cues are extracted and the available cues (feature streams) are selected adaptively to recognize speech robustly [Allen and Li, 2009, Zeng et al., 2005]. The use of multiple feature streams is considered to be an efficient technique because most environmental effects damage limited properties of the signal. For example, locations of zero-crossing points in signals are stable even if the signal is corrupted by low-energy additive noise, since these locations are determined by dominant spectral peaks. In machine recognition, dynamic integration of multiple feature streams is utilized by employing multistream speech recognizers, which estimate speech recognition results from multiple heterogeneous features [Janin et al., 1999]. In order to exploit the multistream speech recognizers, each stream should compensate for the shortcoming of the other streams. Therefore, complementarity of features is important [Sharma, 1999].

On the other hand, several studies on human speech perception have suggested the importance of temporal speech features [Green, 1976]. A strong evidence of the use of features derived by temporal processing is that speech intelligibility can be estimated by measuring amplitude modulation (AM) degradation caused by room acoustics [Houtgast and Steeneken, 1985]. Furthermore, the frequency modulation (FM), which can be treated as the residue of AM, is also considered to be an important acoustical property used in human speech perception [Paliwal and Alsteris, 2003]. [Yoshida et al., 2002] confirmed that the reconstructed signal that preserves the locations of the zero-crossing points of narrow-band waveforms is perceivable by human speech recognition although the AM information in the reconstructed signal is heavily corrupted.

In conventional recognizers, the dynamics of speech are represented by using time-series derivative of features as augmented features. In addition, as in perceptual studies, the importance of AM processing is discussed. As in human speech perception, RASTA (relative spectra) processing of speech proposed by [Hermansky and Morgan, 1994] emphasize the temporal dynamics of narrow-band envelope around 4 Hz in order to achieve robust speech recognition. Recently, data-driven temporal filtering techniques are applied to narrow-band envelopes as extensions of RASTA [Vuuren and Hermansky, 1997]. TRAPS (temporal patterns) [Hermansky and Sharma, 1998], HATS (hidden activation TRAPS) [Chen et al., 2004a], and tonotopic multilayer perceptrons (TMLP) [Chen et al., 2005] are introduced as an integration of two state-of-the-art technologies: data-driven AM filtering and hidden Markov model/multilayer perceptron (HMM/MLP)-tandem approach [Hermansky et al., 2000]. These techniques can extract efficient modulation from narrow-band envelopes of speech signals and are applied to large-vocabulary continuous speech recognition (LVCSR) tasks [Morgan et al., 2005]. Although hidden Markov models (HMMs) are capable of representing temporal changes in acoustical features by transition of the hidden states, this capability is rather poor. This is because the representation of continuous movements of acoustical properties is not accurate since the states in HMMs have discrete values. Furthermore, HMMs cannot represent the long-span dependency because state transitions are assumed to be first-order Markov chains. Therefore, the use of dynamic features that represent continuous long-range movements of acoustical properties is necessary.

Features based on the FM in speech signals have been investigated as complementarity features of AM. Complementarity is an important factor for construction of multistream speech recognizers. Several methods are used to extract phonetic information from the FM in speech signals. For example, [Wang et al., 2003] employed the segmental average instantaneous frequencies of signals. [Chen et al., 2004b] proposed a method based on spectral centroids, which depends on FM of signals. [Dimitriadis et al., 2005] employed the average of instantaneous frequencies weighted by amplitudes. However, these methods are not competitive in performance when FM features are used separately, because these features are proposed as alternatives in combination with amplitude-based features.

However, the results of the perceptual experiments [Yoshida et al., 2002] encouraged the author to consider that FM in speech signals certainly contains phonetic information. In these experiments, it is confirmed that the reconstructed signal that preserves the zero-crossing points of narrowband waveforms are perceivable through human speech recognition. In this thesis, it is considered that FM features can be used as independent features as well as complemental features, if appropriate temporal analyses are carried out. In order to confirm this hypothesis, the data-driven modulation filtering technique are applied to instantaneous frequency trajectories, and then the speech recognition performance is verified when the FM features are used individually and in combination.

3.2 Classification of frequency modulation patterns

In this section, the proposed FM feature extraction system, which can be regarded as a frequency modulation variant of HATS [Chen et al., 2004a], is described. Figure 3.1 shows the block diagram of the proposed system. As shown in the figure, input signals are separated into narrow-band signals by using a filterbank. Then, the pseudo instantaneous frequencies (PIFs) are extracted for each channel in the filterbank. Temporal filters are applied to emphasize essential modulation in the input trajectory of instantaneous frequency. Each component shown in the diagram is described in the following subsections.

3.2.1 Filterbank

In this study, an equal-bark-filterbank defined in [Hermansky, 1990] is used to simulate the frequency responses of the basal membrane of the lining of the inner ear.

The frequency response of k^{th} frequency bin of d^{th} channel ($w^{\mathtt{EBF}}_{d,k}$) is defined by the band-pass filter with exponential slope $\Psi(z)$ in the Bark frequency scale $\Omega^{\mathtt{BARK}}(\omega)$ as follows:

$$\begin{aligned}
w_{d,k}^{\mathtt{EBF}} =& \Psi\left(\Omega^{\mathtt{BARK}}\left(\frac{k}{K}\omega^{\mathtt{RATE}}\right)-d\right),\\
\Omega^{\mathtt{BARK}}(\omega) =& 6\log\left(\frac{\omega}{1200\pi}+\sqrt{\left(\frac{\omega}{1200\pi}\right)^2+1}\right),\\
\Psi(z) =& \begin{cases} 0 & z<-1.3,\\ 10^{2.5(z+0.5)} & -1.3\le z\le -0.5,\\ 1 & -0.5<z<0.5,\\ 10^{-1.0(z-0.5)} & 0.5\le z\le 2.5,\\ 0 & 2.5<z, \end{cases}
\end{aligned} \tag{3.1}$$

where K is the number of frequency bins, which determines the number of samples used in the following frequency sampling method, $\omega^{\mathtt{RATE}}$ is the sampling rate of the raw input signal $\boldsymbol{r}$ in angular frequency (rad/s). Figure 3.2 shows the frequency responses of the filterbank $w_{d,k}^{\mathtt{EBF}}$ as functions of frequency.

The filters in the filterbank are implemented as finite-impulse-response (FIR) filters because it is important to maintain the time-series waveform of signals and linear phase characteristics of filters in this study. Here, impulse responses $\boldsymbol{h}_b^{\mathtt{EBF}}$ of the FIR filters is obtained

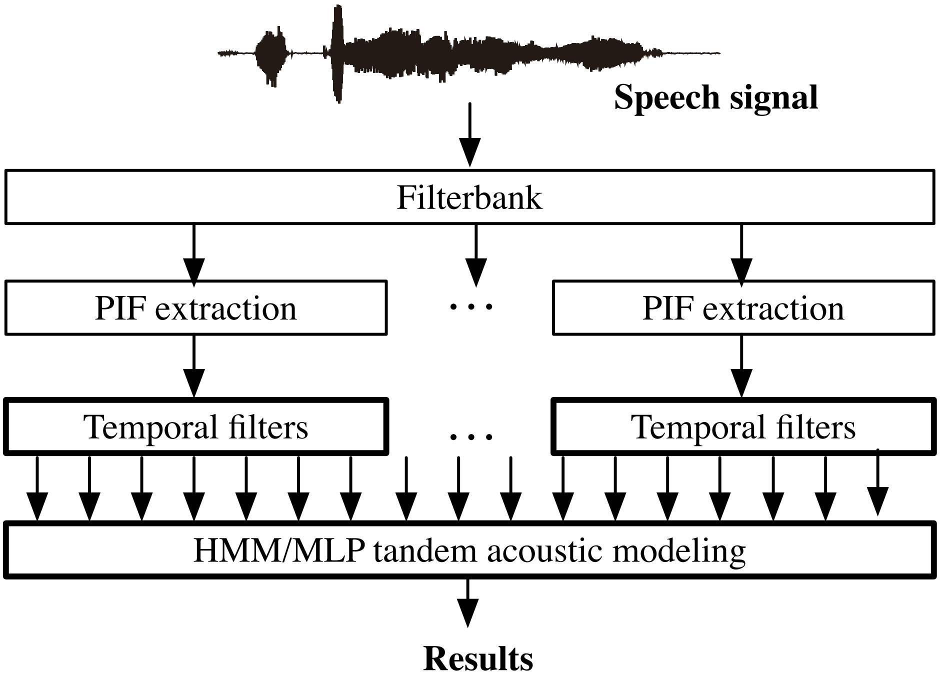


Figure 3.1 Block diagram of proposed FM processing system. The components that require training session are depicted with a thick border

by applying frequency sampling method to $\boldsymbol{w}^{\texttt{EBF}}$ as follows:

$$\boldsymbol{h}_d^{\texttt{EBF}} \overset{\text{def}}{=} \text{Re}[\mathsf{F}^{-1}\left(\boldsymbol{W}^{\texttt{EBF}}\right)_{\boldsymbol{d}}^{\mathsf{T}}], \tag{3.2}$$

where F^{-1} is the matrix representation of the inverse Fourier transform.

Then, a narrow band signal $\boldsymbol{r}_d \overset{\text{def}}{=} \{r_{d,1}, r_{d,2}, \cdots, r_{d,t}, \cdots\}$ of the speech signal $\boldsymbol{r} \overset{\text{def}}{=} \{r_1, r_2, \cdots, r_t, \cdots\}$ is obtained by using convolution, as follows:

$$\boldsymbol{r}_{d,t} \overset{\text{def}}{=} \sum_{\tau=1}^{N(\boldsymbol{h}_d^{\texttt{EBF}})} h_{d,\tau}^{\texttt{EBF}} \cdot r_{t-\tau} \tag{3.3}$$

It should be noted that $\boldsymbol{r}_d$ is sampled at the same sample rate with original speech signal $\boldsymbol{r}$, and is also intractable in statistical models.

3.2.2 Pseudo instantaneous frequency extraction

Several methods have been proposed for AM-FM decomposition, such as the Teager energy operator (TEO) method [Kaiser, 1993] and the method based on the Hilbert transform [Boashash, 1992, Suzuki et al., 2006]. Since the primary motivation behind this study is

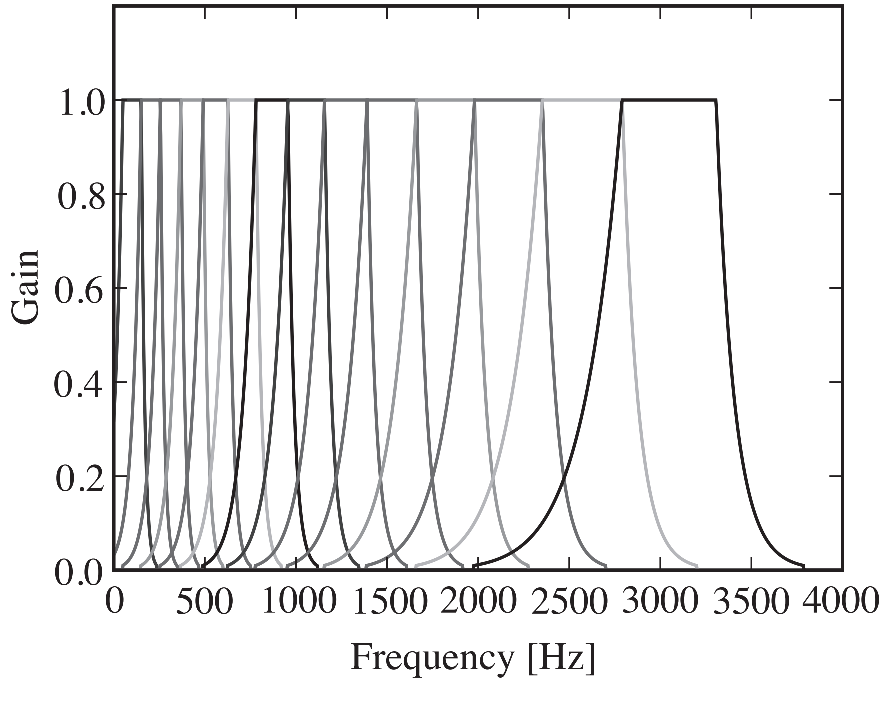


Figure 3.2 Frequency response of filterbank $w_{b,k}^{\texttt{EBF}}$

based on the human perception of the zero-crossing points in signals, a instantaneous frequency (IF) of speech signals are defined by using their zero-crossing points.

The PIF used in this paper is obtained by the following procedure:

1. Measure the time interval $\mathbb{O}_{d,t}$ between the preceding and the following zero-crossing points for the sample at time t in d^{th} narrowband signal.
2. The d^{th} -channel logarithmic PIF ($\varpi_{d,t}$) at time t is defined by $\varpi_{d,t} \overset{\mathrm{def}}{=} \log \frac{\pi}{\mathbb{O}_{d,t}}$.

Figure 3.3 shows the trajectories of PIF and IF obtained by a numerical method [Suzuki et al., 2006]. As shown in the figure, the PIF is correlated to the IF derived by Hilbert transform method. However, PIF has some practical advantages compared to the IF obtained Hilbert transform method;

- PIF can be defined in silent segments of signal,
- PIF value is ensured to be positive.

PIFs can be considered as variants of zero crossing with peak amplitude (ZCPA) features [Kim et al., 1999, Gajic and Paliwal, 2003], in which amplitude weighting is omitted. While

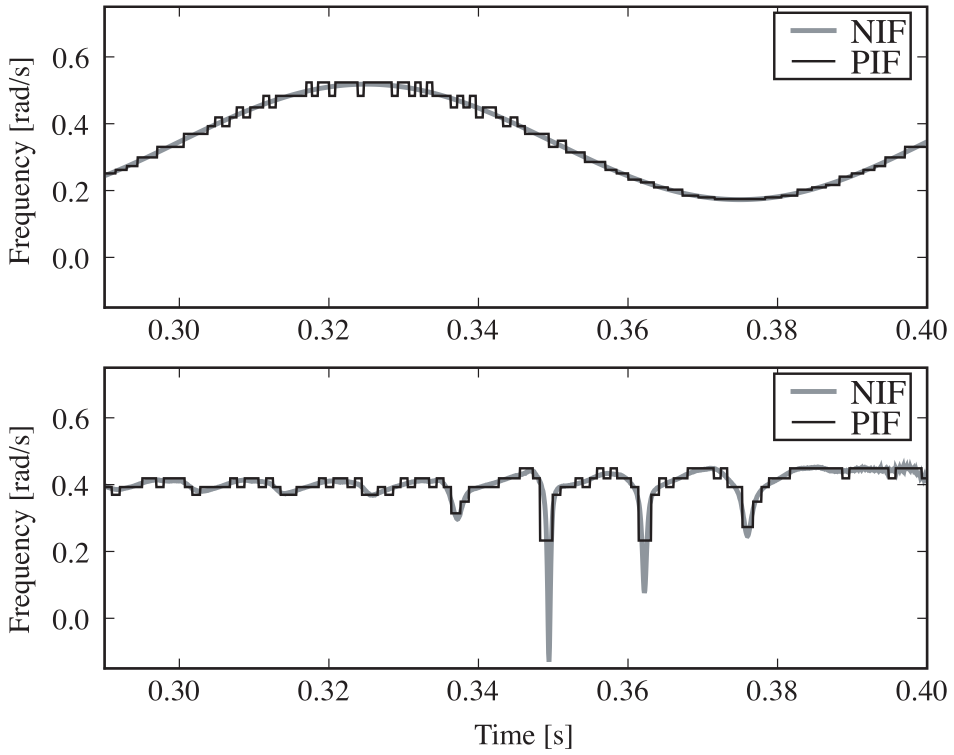


Figure 3.3 Instantaneous frequency obtained by the numerical approach (NIF) [Suzuki et al., 2006] and the PIF of a single sinusoid with FM (top) and a narrow-band speech signal (bottom)

amplitude weighting can improve performance, it can also make features dependent on AM information. This dependency leads to losses in complementarity properties.

The average of the PIF signal is computed for each 25 ms window (10 ms shift) in order to achieve equivalence between the frame rate of proposed and conventional features. Figure 3.4 shows an example of the trajectory of logarithmic envelope and logarithmic PIF. The behaviors of the trajectories of PIF appear disordered and chaotic as compared to envelopes. In order to efficiently handle such complicated trajectories, MLP-based temporal filters and MLP-based acoustic modeling are employed in the system.

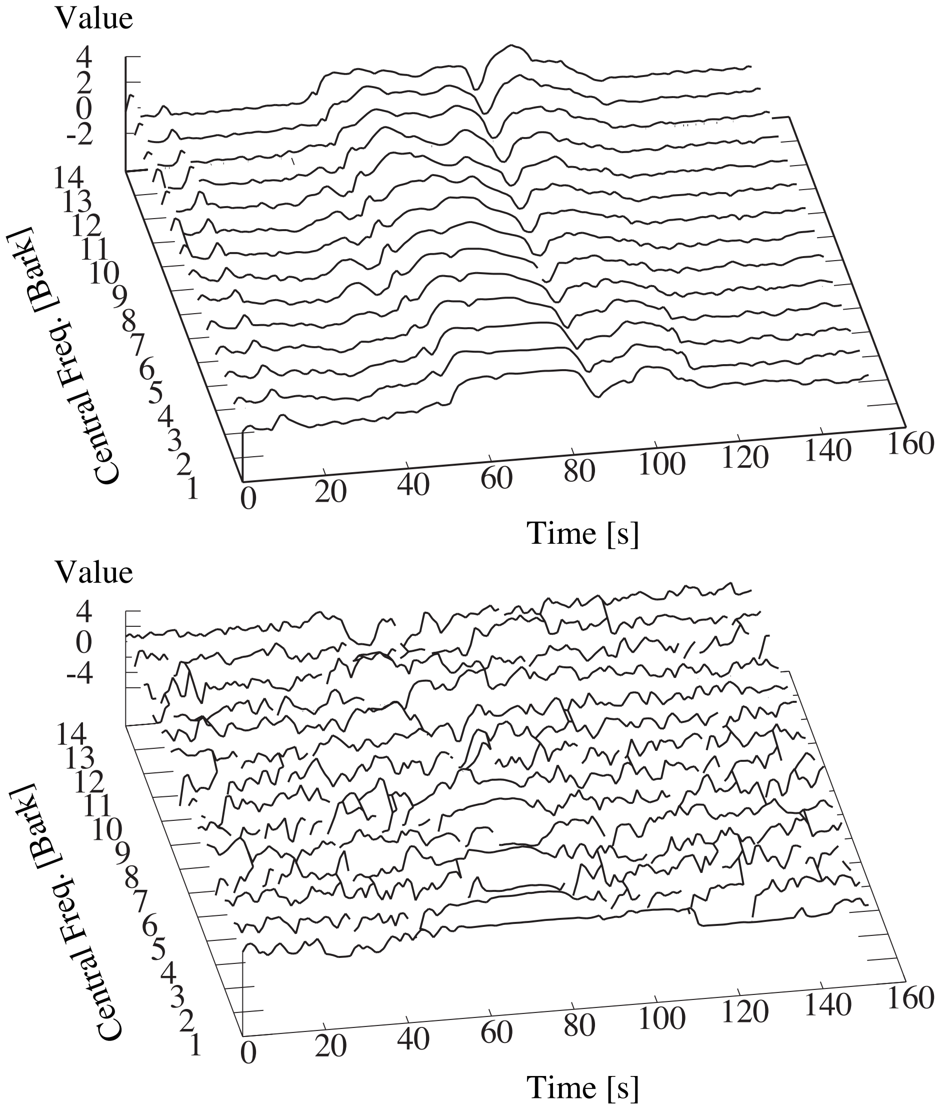


Figure 3.4 Narrow-Band logarithmic envelope (top) and logarithmic PIF (bottom) of speech. (The mean and variance are normalized to 0.5 and 0.25, respectively, for visualization)

3.2.3 Temporal filters

In this section, temporal filtering is applied to Log-PIF sequences $\boldsymbol{\varpi}_d = \{\varpi_{d,1}, \cdots, \varpi_{d,n}, \cdots\}$ in order to emphasize modulation components, which contribute to improve discriminativity of phoneme categories in the Log-PIF. The emphasis technique used in this chapter is based on multilayer-perceptron, which is conventionally used to extract the significant amplitude modulation (AM) from envelopes [Chen et al., 2004a]. In this chapter, this technique is applied to the trajectory of Log-PIF to extract frequency modulation.

First, in order to reduce the sampling rate, the low-pass filtering and resampling to Log-PIF sequence are applied. Then, fixed-length subsequence are defined as a features as follows:

$$\boldsymbol{x}^{\mathtt{IFS}}_{d,n} \overset{\text{def}}{=} \left[\tilde{\varpi}_{d,(n-(D-1)/2)}, \cdots \tilde{\varpi}_{d,n}, \cdots \tilde{\varpi}_{d,(n+(D-1)/2)}\right]^{\mathsf{T}} \tag{3.4}$$

where an odd number D is the number of dimensionality of subsequence $\boldsymbol{x}^{\mathtt{IFS}}_{d,n}$, which is typically set at 51, and $\tilde{\varpi}_{d,n}$ is n^{th} frame of resampled Log-PIF sequence obtained from d^{th} channel.

Then, similar to the tandem-approach described in Section 2.3, an MLP-based classifier for each subsequence that attempts to classify subsequence to a frame-level phoneme label is introduced. Here, a 2-layer perceptron and sigmoid-type nonlinear warping functions are used as follows:

$$\boldsymbol{\mathcal{T}}(\boldsymbol{x};\Xi_d) = \varsigma\left(\boldsymbol{C}_{d,2}\underbrace{\varsigma\left(\overbrace{\boldsymbol{C}_{d,1}\boldsymbol{x}}^{\text{Convolution}} + \boldsymbol{b}_{d,1}\right)}_{\text{Nonlinear compression}} + \boldsymbol{b}_{d,2}\right), \tag{3.5}$$

where $\Xi_d \overset{\text{def}}{=} \{\boldsymbol{C}_{d,1} \in \mathbb{R}^{D\times M}, \boldsymbol{C}_{d,2} \in \mathbb{R}^{M\times|\mathbb{P}|}, \boldsymbol{b}_{d,1} \in \mathbb{R}^{M}, \boldsymbol{b}_{d,2} \in \mathbb{R}^{|\mathbb{P}|}\}$ is the parameter of the MLPs corresponding to d^{th} channel, $|\mathbb{P}|$ is the number of elements in a set of monophones, M is a hyper-parameter to determine the number of feature components extracted from instantaneous frequency trajectory. Reconsidering that $\boldsymbol{x}$ is taken from subsequences of $\boldsymbol{\varpi}_d$, the linear transformation of $\boldsymbol{x}$ can be assumed as convolution of $\boldsymbol{\varpi}_d$. Therefore, each element in the transformed vector $\boldsymbol{C}_{d,1}\boldsymbol{x}$ can be assumed as an output of linear convolution filters. The sigmoid function can be assumed as nonlinear compression applied to filtered trajectory $\boldsymbol{C}_{d,1}\boldsymbol{x}$.

In this formulation, optimal filter coefficients $\boldsymbol{C}_{d,1}$ can be obtained by standard the Back-Propagation Algorithm. Similar to tandem-approach, by introducing the teaching signal Eq.

(2.38), the optimization with respect to phoneme classification can be expressed as follows:

$$\hat{\Xi}_d = \mathop{\mathrm{argmin}}_{\Xi_d} \sum_i \sum_{n=1}^{N(X^i)} ||\boldsymbol{\mathcal{T}}(\boldsymbol{x}_n^{i\mathtt{IFS}}; \Xi_d) - \hat{\boldsymbol{x}}_n^i||^2 \tag{3.6}$$

Further, the emphasized features can be defined by using an incomplete MLP-function $\tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}; \hat{\Xi}_d)$, as follows:

$$\begin{aligned} \boldsymbol{x}_n^{\mathtt{FM}} &\stackrel{\mathrm{def}}{=} \left[\tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}_{1,n}^{\mathtt{IFS}}; \hat{\Xi}_1)^{\mathsf{T}}, \tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}_{2,n}^{\mathtt{IFS}}; \hat{\Xi}_2)^{\mathsf{T}}, \cdots, \right]^{\mathsf{T}} \\ \tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}; \hat{\Xi}_d) &= \varsigma\left(\boldsymbol{C}_{d,1} \cdot \boldsymbol{x} + \boldsymbol{b}_{d,1}\right) \end{aligned} \tag{3.7}$$

The average modulation frequency response of the filters, obtained by the data set used in Section 3.4, is shown in Figure 3.5. Interestingly, similar to the studies on AM features [Hermansky, 1998], the modulation around 4 Hz is important even in the discrimination of FM features.

3.3 Combination of AM and FM classifiers

This section describes a combination method of AM and FM classifiers derived from an inverse-entropy-based combination of the tandem acoustic models, as introduced by [Okawa et al., 1998, Ikbal et al., 2004].

The AM processing method used in this study is the HATS system proposed by [Chen et al., 2004a]. Since the proposed system can be assumed to be an extension of HATS, most of the fundamental frameworks can be commonly used. Figure 3.6 shows a block diagram of the combination systems.

By applying this technique, the feature vector of the combined system (*Amplitude and*

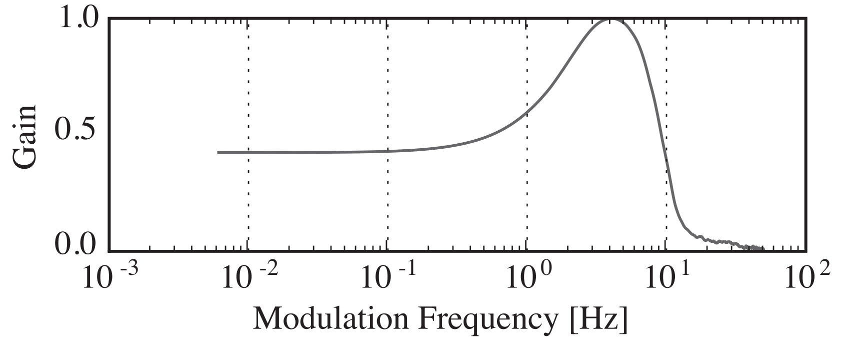


Figure 3.5 Average of modulation frequency responses $\hat{w}_{nb}$ for all channels and all modulation components

Frequency Modulation Classifiers; AFMC) can be defined as follows:

$$\boldsymbol{x}_n^{\mathtt{AFMC}} \overset{\mathrm{def}}{=} \nu_n \boldsymbol{x}_n^{\mathtt{FMMLP}} + (1-\nu_n)\boldsymbol{x}_n^{\mathtt{AMMLP}}, \tag{3.8}$$

where ν_n is the dynamic weighting coefficient, and $\boldsymbol{x}^{\mathtt{FMMLP}}$ and $\boldsymbol{x}^{\mathtt{AMMLP}}$ are the output vectors of FM-MLP and AM-MLP obtained by using the MLP function $\mathcal{T}$, as follows:

$$\begin{aligned} \boldsymbol{x}_n^{\mathtt{FMMLP}} &\overset{\mathrm{def}}{=} \mathcal{T}\left(\boldsymbol{x}_n^{\mathtt{FM}}; \hat{\Xi}^{\mathtt{FMMLP}}\right), \\ \boldsymbol{x}_n^{\mathtt{AMMLP}} &\overset{\mathrm{def}}{=} \mathcal{T}\left(\boldsymbol{x}_n^{\mathtt{AM}}; \hat{\Xi}^{\mathtt{AMMLP}}\right). \end{aligned} \tag{3.9}$$

Here, the parameter sets, $\hat{\Xi}^{\mathtt{FMMLP}}$ and $\hat{\Xi}^{\mathtt{AMMLP}}$ are obtained by the standard back propagation algorithm. $\boldsymbol{x}^{\mathtt{AM}}$ is the feature vector obtained from the HATS system.

The HATS feature vector $\boldsymbol{x}^{\mathtt{AM}}$ is obtained by emphasizing energy trajectory of narrow-band signals as follows:

$$\begin{aligned} \boldsymbol{x}_n^{\mathtt{AM}} &\overset{\mathrm{def}}{=} \left[\tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}_{1,n}^{\mathtt{IAS}}; \hat{\Xi}_1)^{\mathsf{T}}, \tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}_{2,n}^{\mathtt{IAS}}; \hat{\Xi}_2)^{\mathsf{T}}, \cdots, \tilde{\boldsymbol{\mathcal{T}}}(\boldsymbol{x}_{B,n}^{\mathtt{IAS}}; \hat{\Xi}_B)^{\mathsf{T}}, \right]^{\mathsf{T}} \\ \boldsymbol{x}_{b,n}^{\mathtt{IAS}} &\overset{\mathrm{def}}{=} [\tilde{e}_{b,(n-(D-1)/2)}, \cdots \tilde{e}_{b,n}, \cdots \tilde{e}_{b,(n+(D-1)/2)},]^{\mathsf{T}} \end{aligned} \tag{3.10}$$

where $\tilde{e}_{b,n}$ is the energy of n^{th} frame in b^{th} channel obtained as follows:

$$\tilde{e}_{b,n} = \sum_{t=T_n^{\mathtt{START}}}^{T_n^{\mathtt{STOP}}} ||r_{b,t}||^2. \tag{3.11}$$

Here, $T_n^{\mathtt{START}}$ and $T_n^{\mathtt{STOP}}$ are the indices of the first sample and the last sample in the n^{th} frame.

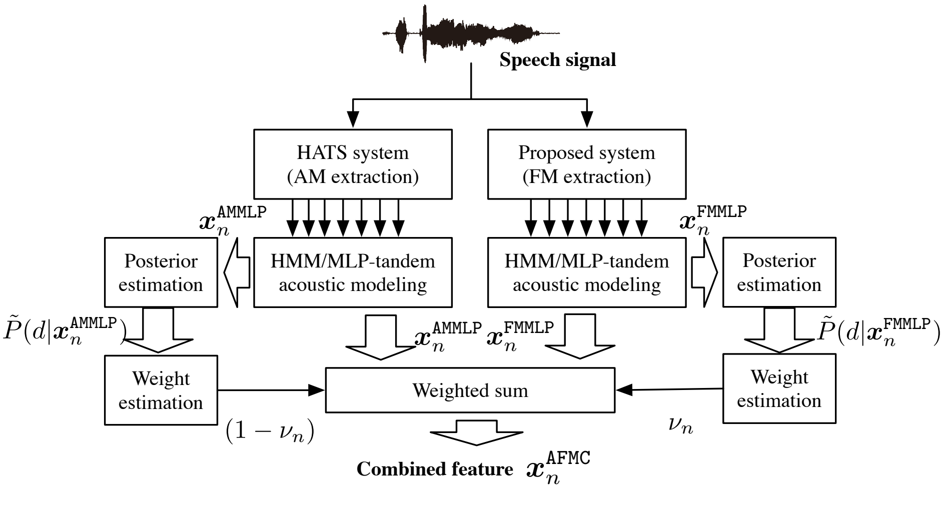

Speech signal
HATS system (AM extraction)
Proposed system (FM extraction)
$\boldsymbol{x}_n^{\mathtt{AMMLP}}$
$\boldsymbol{x}_n^{\mathtt{FMMLP}}$
Posterior estimation
HMM/MLP-tandem acoustic modeling
HMM/MLP-tandem acoustic modeling
Posterior estimation
$\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{AMMLP}})$
$\boldsymbol{x}_n^{\mathtt{AMMLP}}$
$\boldsymbol{x}_n^{\mathtt{FMMLP}}$
$\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{FMMLP}})$
Weight estimation
Weighted sum
Weight estimation
$(1-\nu_n)$
ν_n
Combined feature $\boldsymbol{x}_n^{\mathtt{AFMC}}$

Figure 3.6 Block diagram of combination of proposed system and HATS system. The variable names in the figure correspond to the variables in Eq. (3.12)

The weight coefficient ν_n is determined by inverse entropy $\{H[.]\}^{-1}$ of monophone posterior pdfs ($\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{AMMLP}})$ and $\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{FMMLP}})$) estimated from the output vectors ($\boldsymbol{x}_n^{\mathtt{FMMLP}}$ and $\boldsymbol{x}_n^{\mathtt{AMMLP}}$), as follows:

$$\begin{aligned}
\nu_n &= \frac{\left\{H[\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{AMMLP}})]\right\}^{-1}}{\left\{H[\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{FMMLP}})]\right\}^{-1} + \left\{H[\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{AMMLP}})]\right\}^{-1}}, \\
\tilde{P}(d|\boldsymbol{x}_n) &\overset{\mathrm{def}}{=} \frac{\left(\exp\left(x_{n,d}\right) - 1\right)^{-1}}{\sum_{d'} \left(\exp\left(x_{n,d'}\right) - 1\right)^{-1}} \\
H[\tilde{P}(d|.)] &= -\sum_{d} \tilde{P}(d|.) \log \tilde{P}(d|.),
\end{aligned} \tag{3.12}$$

where $\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{FMMLP}})$ and $\tilde{P}(d|\boldsymbol{x}_n^{\mathtt{AMMLP}})$ are the estimated posterior probability calculated by the cancellation of the sigmoid function and application of softmax transfer function to the output of FM-MLP and AM-MLP, respectively.

Figures 3.7, 3.8 and 3.9 show examples of trajectories of $\boldsymbol{x}^{\mathtt{AFMC}}$, $\boldsymbol{x}^{\mathtt{FMMLP}}$ and $\boldsymbol{x}^{\mathtt{AMMLP}}$, respectively. The utterance used in these examples is "/sil i ch i sil/" (clean speech). As shown in Figure 3.8 and 3.9, monophone classification can be performed by both of the FM classifier and the AM classifier even though FM patterns appear disordered and chaotic (cf. Figure 3.4). Furthermore, the accurate recognition can be done by using combination of classifier as shown in Figure 3.7.

3.4 Experiments and discussions

In order to evaluate the performance of the proposed system, noisy digit recognition experiments are performed. In this section, the efficiency of FM features used individually is evaluated at first. Then, the performance of the system with FM features used in combination

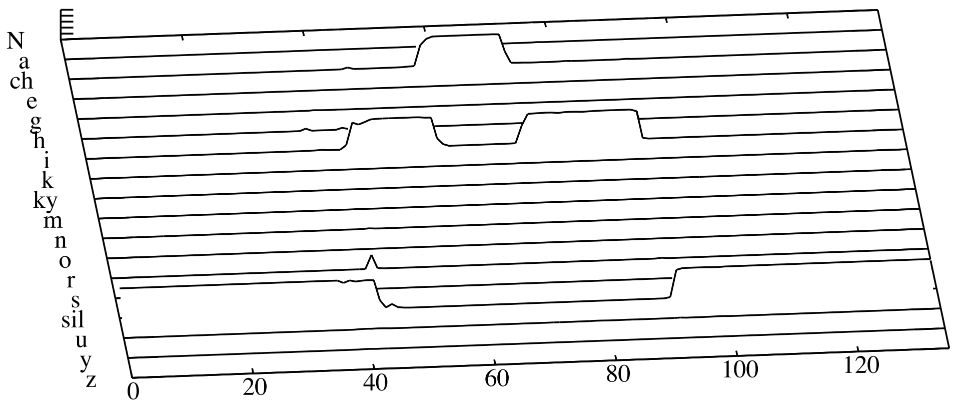


Figure 3.7 Trajectories of $\boldsymbol{x}^{\mathtt{AFMC}}$ as functions of frame n

is evaluated by multistream speech recognition experiments. Furthermore, to investigate the advantages of the FM analysis, the noisy speech recognition experiments in artificial noisy environments are performed.

3.4.1 Dataset and model description

The training set and the test set are taken from CENSREC-1 (a.k.a. AURORA-2J) [Nakamura et al., 2005], which is the Japanese translation of the dataset AURORA-2 [Pearce and Hirsh, 2000]. The training set used for both MLP and HMM comprises 8,440 utterances of clean speech obtained from 110 speakers. In these experiments, the sample rate of speech signals is fixed to 8,000 Hz. Therefore, the Bark filterbank splits the signals into 14 filtered signals.

In the experiments, 20 frequency modulation components are extracted for each narrowband signal. Therefore, the number of features used in acoustic models is the product of the number of bands (14) and 20 (i.e. $M = 20$ in Eq. (3.5)). The number of hidden neurons for

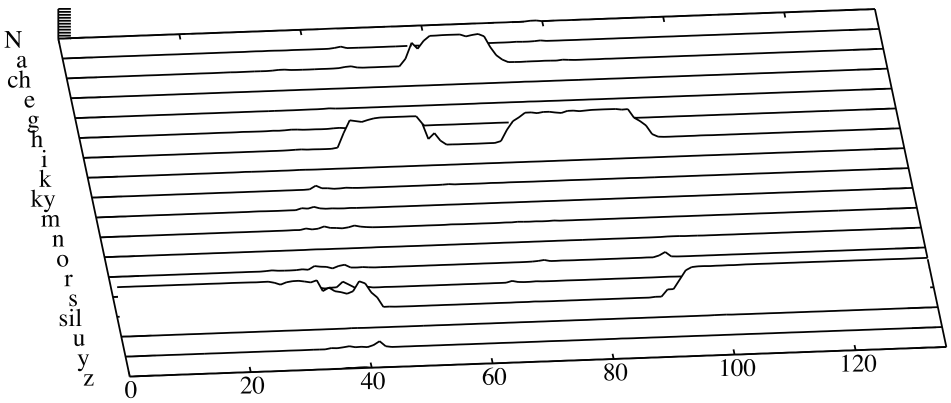


Figure 3.8 Trajectories of $\boldsymbol{x}^{\mathtt{FMMLP}}$ as functions of frame n

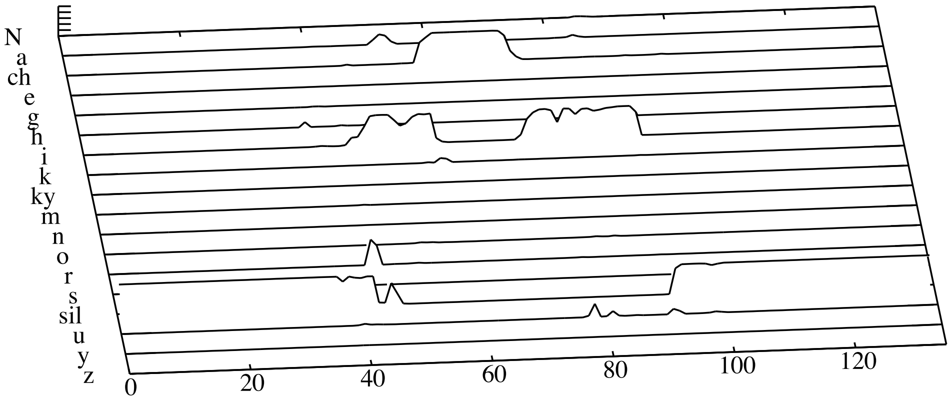


Figure 3.9 Trajectories of $\boldsymbol{x}^{\mathtt{AMMLP}}$ as functions of frame n

MLPs in tandem acoustic models is fixed to 200.

The HMM configurations in the experiments are same as those in CENSREC-1 baseline systems and similar to those in standard AURORA-2 baseline systems; each digit is modeled by CD-HMM with 16 states, silence is modeled by 5-state CD-HMM, and short-pause is modeled by 3-state CD-HMM. The number of mixture components is fixed to 20 for digit HMMs and 36 for silence or short-pause HMMs. Only the variances of features are modeled by the Gaussian distributions. In other words, all covariance matrices in the models are assumed to be diagonal.

3.4.2 Noisy speech recognition experiments

In order to clarify the individual performance of FM features, a noisy speech recognition task is carried out. Four noise environments from CENSREC-1 (restaurant, street, station, and airport) are selected for the test. The test set comprises 1,001 utterances for each noise environment and each signal-noise-ratio (SNR) condition.

Following speech recognition systems are compared:

- MFCC
  This is the standard MFCC speech recognition system. MFCC features are augmented by energy (E), Δ MFCC, ΔE, $\Delta\Delta$ MFCC, and $\Delta\Delta$ E.
- MFCC (CMS)
  Speech recognizer is constructed by incorporating utterance-level cepstral mean subtraction (CMS) techniques.
- AIF
  The recognizer is based on average instantaneous frequency (AIF) features that are defined by removing the average log-envelope (ALE) features from the AIF/ALE systems [Wang et al., 2003] (augmented by its derivations and accelerations; 42 dimensions.)
- HATS (AM)
  The recognizer is based on HATS system [Chen et al., 2001, Chen et al., 2004a]. AM features are extracted by using data-driven temporal filtering method. The number of features is 280. The acoustic model is based on the HMM/MLP tandem approach.
- FM
  This is the proposed FM processing systems. The number of features is 280. The acoustic model is based on the HMM/MLP-tandem approach.

Figure 3.10 shows the word error rates of the methods being compared, as a function of SNR.

From the figure, it is confirmed that the performance of the FM speech recognizers is comparable to that of the conventional MFCC speech recognition systems. Although the proposed method is inferior to the AM systems, it is confirmed that FM does contain phonetic information; further, data-driven temporal filtering technique and the HMM/MLP-tandem approach enabled the use of phonetic information obtained from FM. The substantial improvement in the performance of FM from that of AIF as compared to the performance improvement from MFCC to AM indicate that nonlinear processing is intrinsic to FM features.

3.4.3 Multistream speech recognition experiments

In this section, the performance of the combination of AM and FM processing is evaluated by performing multistream speech recognition experiments. The combination method of AM and FM features is described in Section 3.3.

The test set used in this experiment is the same as that used in experiments described in Section 3.4.2. The performance of the combination method is compared with the performances of the AM and FM systems, described in the previous section.

Figure 3.11 shows the word error rates of singlestream speech recognition systems (AM and FM) and their combination (AFMC).

From the figure, it is confirmed that the combination of AM and FM is efficient for achieving noise robustness. It should be noted that the combination method outperforms MFCC systems and conventional AM systems, even in clean environments. Therefore, the various techniques for speech emphasis should lead to performance improvements.

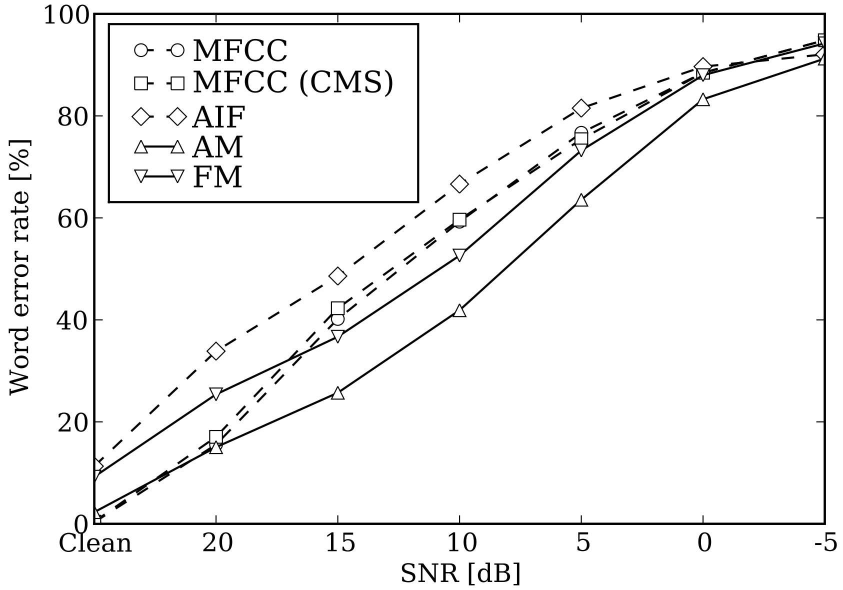


Figure 3.10 Word error rate of singlestream speech recognizers as a function of SNR

For comparison, optimal static weights, $\bar{\nu}$, which are independent of time n, are obtained by minimizing the squared error as follows:

$$\bar{\nu} = \underset{\nu}{\operatorname{argmin}} \sum_i \sum_n \left(\hat{\boldsymbol{x}}_n^i - \left(\nu \boldsymbol{x}_n^{i,\mathtt{FMMLP}} + (1-\nu) \boldsymbol{x}_n^{i,\mathtt{AMMLP}} \right) \right)^2, \tag{3.13}$$

where $\hat{\boldsymbol{x}}_n^i$ is the teaching signals, as defined in Section 2.3.

Table 3.1 shows the performance of the static combination method ($\bar{w}_m$) and proposed dynamic combination method ($w_m(t)$). It is confirmed that the dynamic combination is more effective, especially in noisy environments. In clean environments, it is observed that the static weights determined by optimization in Eq. (3.13) give satisfactory results. However, in realistic environments, the weights of these analyzers change. Therefore, the dynamic determination of weights is a critical step in achieving noise robustness.

The results suggest that the advantages of each feature stream are dependent on time. It is considered that dynamic integration is required because noise properties of realistic noise are varied, and hence, the advantageous feature streams are varied. In the following section, the relation between noise property and robustness of each stream are examined and discussed.

3.4.4 Complementarity evaluation

In order to investigate the difference in robustness of AM and FM analysis systems, various artificial noises are defined and used to evaluate the systems.

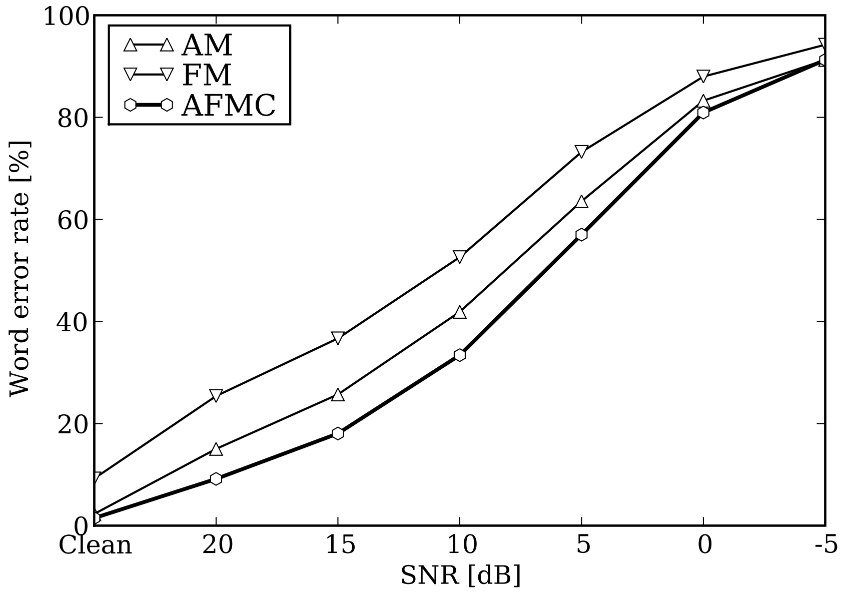


Figure 3.11 Word error rate of multistream speech recognizers for noisy speech as a function of SNR

Table. 3.1 Word error rates of the multistream speech recognition systems that use static stream weighting (Static; $\bar{\nu}$ in Eq. (3.13)) and dynamic stream weighting (Dynamic; ν_n in Eq. (3.12))

	Static	Dynamic
Clean	1.62	1.48
10 dB	39.45	33.43

Table. 3.2 Word error rates of compared methods in noisy environments (10 dB)

	AM	FM
wn	54.5	50.5
bpf_wn	64.7	97.1
burst_wn	50.9	37.1
burst_bpf_wn	37.1	65.3

The properties of noise considered in this section are listed below.

- Stationary noise or burst noise
- Narrow-Band noise or wide-band noise (white noise)

The following different noise patterns are created by combining these properties;

- White noise (wn)
  Full-range white noise.
- Band-pass filtered white noise (bpf_wn)
  This noise is obtained by applying band-pass filter to the noise "wn." The central frequency of a band-pass filter is obtained from uniform random values ranging from 1,000 Hz to 3,000 Hz, and the bandwidth is obtained from uniform random values ranging from 100 Hz to 2000 Hz.
- Burst noise (burst_wn)
  White noise of with a duration 250 ms and silence with a duration of 250 ms are connected alternately.
- Band-pass filtered burst noise (burst_bpf_wn)
  This is obtained by applying band-pass filter (the parameters for filters are same as in "bpf_wn") to the noise "burst_wn."

The spectrograms of these noises are depicted in Fig. 3.12. Those four noises are added at 10 dB SNR to clean speech data in the test set of CENSREC-1.

Table 3.2 lists the word accuracies of the AM and FM methods in the tested environments. From the results, it is observed that the FM analysis has certain disadvantages in the case of narrow-band noises. However, it is advantageous for full-range noises. In contrast, the AM analysis is often observed to be degraded under full-range burst noise.

The error rates of each classifier differ significantly, depending on the noise characteristics. Therefore, it is confirmed that the two classifiers share a complementary relation. Because burst noise degrades the modulation information of in envelope, the performance of AM recognizers in the “burst_wn” environment was not sufficiently high. FM speech recognizers work robustly, even when AM features are degraded. It is considered that these complementarity characteristics of FM features make it possible to achieve robustness in realistic noise environment.

3.4.5 Reverberant speech recognition experiments

In this section, the performance of the proposed system is evaluated by conducting reverberant digit recognition experiments.

As similar to the previous experiments, the training set used for both the MLP and HMM was taken from CENSREC-1. Every test set that was used for both the clean environment test and reverberant environment tests comprised 2002 utterances from 104 speakers. The sampling rate for all the input signals was fixed at 8000 Hz; hence, the number of filter bank channels was 14.

Four impulse responses for producing reverberant speech is prepared in order to simulate the reverb at the following locations:

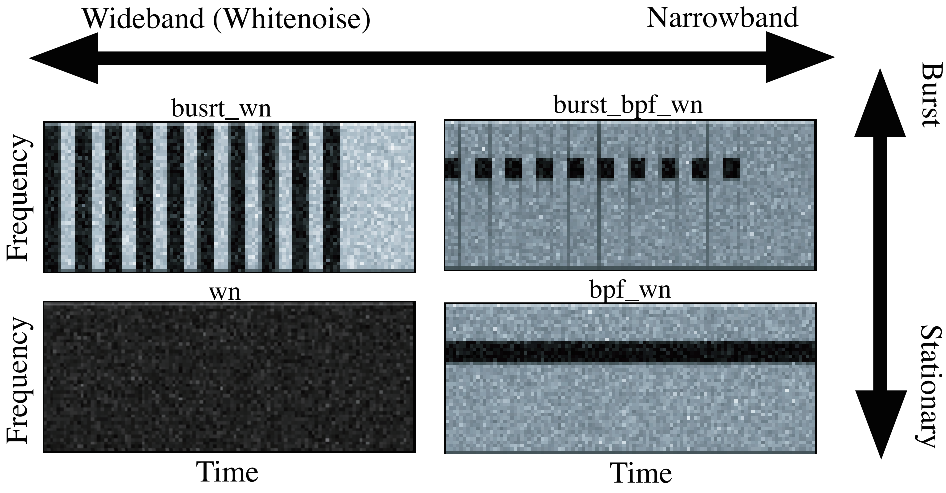


Figure 3.12 Spectrograms of selected noise patterns

- Room (small reverberant room)
- Meeting room
- Silo
- Theater
- Cathedral

Figure 3.13 shows the time characteristics of the reverb calculated by using the expression

$$R(n) = 10\log_{10}\left\{\frac{\sum_{\tau=n}^{\infty}(h_\tau^{\mathtt{REV}})^2}{\sum_{\tau=0}^{\infty}(h_\tau^{\mathtt{REV}})^2}\right\}, \tag{3.14}$$

where $h_\tau^{\mathtt{REV}}$ is the impulse response of the reverb.

The reverberation test set was generated by convoluting the impulse responses to signals in clean environments. Figure 3.14 shows the spectrograms for clean speech and reproduced reverberant speech using the impulse response of the silo.

The baseline is the MFCC and energy feature extraction system that is augmented by the derivation and acceleration of the MFCC and energy. (MFCC_E_D_A; 39 dims.)

All the HMMs and MLPs are trained to be independent of the gender and speaker.

Table 3.3 shows the word error rates of the compared methods in the test set.

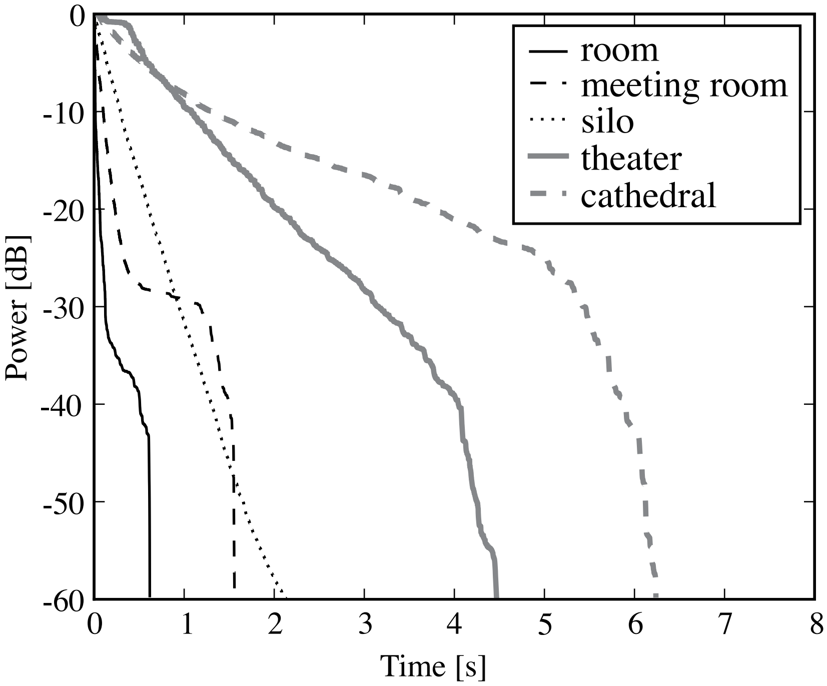


Figure 3.13 Time characteristics of reverb

Table. 3.3 Word error rates of compared methods as percentages

	MFCC	AIF	AM	FM	AFMC
Clean	2.1	10.3	2.5	9.1	1.6
Room	21.1	62.9	13.4	27.8	7.9
Meeting room	65.7	72.3	58.2	55.5	48.0
Silo	73.8	86.5	74.2	73.6	68.6
Theater	76.9	85.9	68.8	78.6	65.0
Cathedral	84.2	98.8	83.2	86.3	82.1
Avg. reverb	64.3	81.3	59.6	64.3	54.3

3.5 Conclusion

In this chapter, the speech recognition system based on data-driven temporal filtering techniques is presented. By performing the speech recognition experiments, it is confirmed that

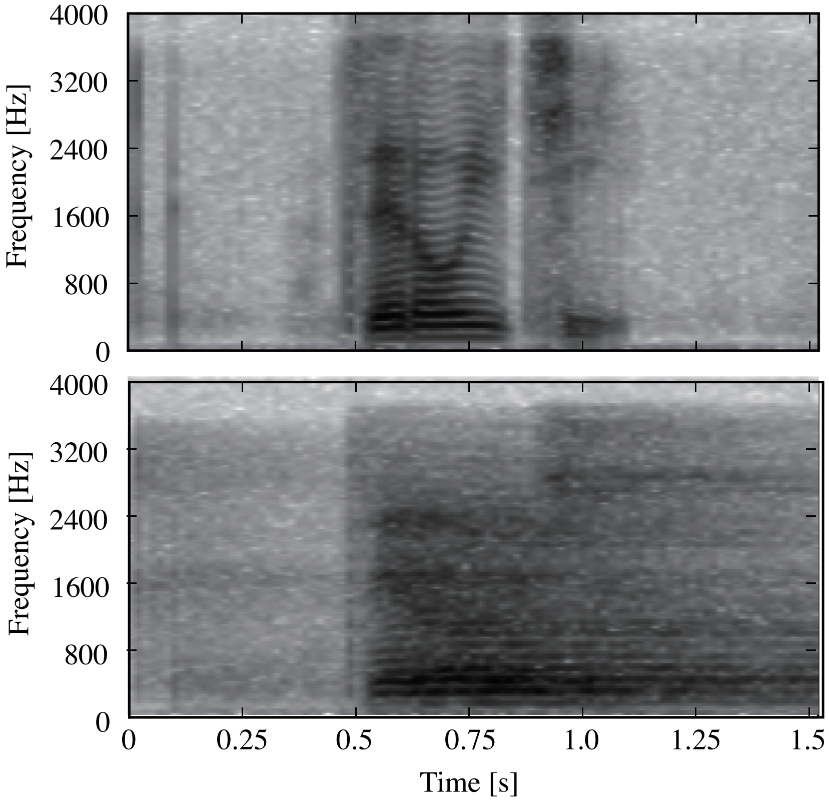


Figure 3.14 Spectrograms for clean speech (top) and reverberant speech in silo (bottom)

frequency modulation (FM) in speech signals contains phonetic information. In the proposed systems, only the density of zero-crossing points in the signal is analyzed. Therefore, the features used in the proposed system do not contain information on amplitude. The evaluation results show that FM in speech signals also contains phonetic information and that the FM features can be treated as independent features as well as complemental features.

Furthermore, the efficiency of the combination of AM and FM systems are verified. It is confirmed that the combination system outperformed all the conventional singlestream recognizers. The combination system reduced word error by 43.6% at 10 dB SNR.

To evaluate the complementarity of AM and FM features, their performance under artificial noisy environments are evaluated. The results show that the characteristics of FM features and AM features are completely different. FM features are considerably robust to wide-band noise, where AM features are not.

Furthermore, reverberant speech recognition experiments are carried out in order to verify advantage of the proposed system. It is confirmed by the experiments that the FM analysis and AM/FM combination system are advantageous for reverberant speech recognition.

Through series of experiments, it is demonstrated that the proposed FM features can achieve sufficiently high performance when used in singlestream speech recognizers and can outperform conventional recognizers when used in combination with AM features.

Finally, it is confirmed in this chapter that further improvement in performance of ASR systems can be achieved by employing high-dimensional signal representations. Although, in this chapter, the analysis method based on FM is presented, combining other analysis methods would also be effective.

Chapter 4

Regularized discriminative models based on continuous-density hidden Markov models

In the previous chapter, the MLP-ensemble classifier is introduced in order to prevent the *curse of dimensionality*. In this chapter, as another method to construct efficient classifiers, an HMM-based classification derived by the minimum relative entropy discrimination framework is proposed. Since this chapter only focuses on an acoustic model estimation method, experiments are carried out by using conventional MFCC features (MFCC_E_D_A described in Section 2.1.1).

4.1 Introduction

Recently, discriminative training methods for probabilistic models have achieved higher performance than conventional maximum likelihood training methods, even in large vocabulary continuous speech recognition tasks [Woodland, 2002, McDermott and Katagiri, 2005]. In the discriminative training methods, probabilistic model parameters are estimated by optimizing a discriminative criterion function. Several methods for discriminative training have been identified along with choices of performance functions as described in Section 2.2.4. Although discriminative training methods significantly outperform conventional maximum likelihood training methods, the training processes still include the risk of overfitting due to a shortage of available training data.

On the other hand, in discriminative non-probabilistic models, several regularization techniques are employed in order to avoid overfitting. The support vector machine (SVM) is one of the most successful discriminative models that utilize regularization techniques

[Boser et al., 1992, Vapnik, 1999]. The regularization techniques introduce additional terms that represent how desirable the model parameters are, for the performance function in order to prevent overfitting. In SVMs, large-margin linear classifiers are obtained by introducing regularization terms that minimize the L2-norm of weight vectors used in linear classifiers. Further, it is well known that SVM can prevent the *curse of dimensionality*.

Regularization techniques are also imported into discriminative training methods for probabilistic models. I-smoothing technique can be interpreted as a regularization technique that controls the estimated model parameters so that higher likelihood as well as discriminative performance is ensured [Povey and Woodland, 2002]. Large-margin hidden Markov Models (LM-HMMs) introduced by Sha *et al.* [Sha and Saul, 2007] include additional regularization terms that lead to a decrease in the L2-norm of *natural parameters* of each Gaussian pdf in HMMs.

In order to provide probabilistic interpretations of existing regularized discriminative training methods, the minimum relative entropy discrimination (MRED) framework proposed by Jebara *et al.* [Jebara, 2001, Jaakkola et al., 2000] [*1] is applied to discriminative training of CD-HMMs. Although MRED has already been applied in regression problems involving one-class HMMs [Jebara, 2001] and feature selection problems for HMMs [Valente and Wellekens, 2003], the application in the sequential pattern recognition problems, i.e., classification problems that handle sequences of continuous vectors as inputs and sequences of labels as outputs, are not discussed.

In this study, MRED is applied to continuous density HMMs (CD-HMMs). Because the MRED framework is a Bayesian framework, the author intends to provide novel discriminative perspectives for various problems in speech recognition, including model selection, adaptation, and feature extraction, by applying MRED to CD-HMMs. In this thesis, as a first step, an MRED-based discriminative training method is discussed.

The rest of this chapter is organized as follows. In Section 4.2, the MRED framework is described. In Section 4.3, an approximation method for MRED is proposed in order to apply MRED to CD-HMMs. An example of optimization method is presented in Section 4.4. Experimental setup is presented in Section 4.5, and the results are discussed in Section 4.6.

4.2 Minimum relative entropy discrimination (MRED)

In this section, at first, a general formulation of regularized discrimination is introduced by considering constrained convex optimization. Then, MRED is described as a natural exten-

[*1] Conventionally, this framework is also known as “maximum entropy discrimination.” However, the author uses a more specific notation, i.e., “minimum relative entropy discrimination,” because the maximum entropy property can only be acquired when specific prior pdfs are used. The author of [Jebara, 2001] also uses this specific notation in several papers.

sion of the general formulation.

This section begins with a definition of a parametric discriminant function $\mathcal{D}(\boldsymbol{X}^i;\Theta)$, which indicate the discriminative performance of given parameter Θ with respect to an input example $\boldsymbol{X}^i$. By introducing a performance threshold ξ^i (a.k.a. functional margin) with respect to i^{th} input example, the focus of the problem is to estimate Θ so that $\mathcal{D}(\boldsymbol{X}^i;\Theta) \geq \xi^i$.

Using the discriminant function $\mathcal{D}(\boldsymbol{X}^i;\Theta)$, all regularized discrimination problems can be written under a convex optimization framework in the following manner:

$$\begin{aligned} &\underset{\Theta,\xi}{\text{minimize}}\ R(\Theta) + \sum_i L(\xi^i), \\ &\text{subject to}\ \mathcal{D}(\boldsymbol{X}^i;\Theta) - \xi^i \geq 0, \quad \forall i. \end{aligned} \tag{4.1}$$

Here, $R(\Theta)$ is a regularization function, and $L(\xi^i)$ is a loss function that returns a positive value when the performance threshold ξ^i is lower than a desirable performance. The regularization function $R(\Theta)$ is designed to represent the illness of the parameter Θ.

In this formulation, when $\boldsymbol{X}^i$ is misclassified, $\mathcal{D}(\boldsymbol{X}^i;\Theta)$ is less than 0, and ξ^i is set at $\xi^i = \mathcal{D}(\boldsymbol{X}^i;\Theta)$ in order to satisfy the constraint by slacking the constraint by ξ^i. Therefore, ξ^i is termed "slack variable." Because the discriminant function $\mathcal{D}(\boldsymbol{X}^i;\Theta)$ is completely ignored if ξ^i is determined freely, ξ^i is controlled by monotonically increasing loss function $L(\xi^i)$.

Although, the composition of the loss function and the discriminant function $L(\mathcal{D}(\boldsymbol{X}^i;\Theta))$ is treated as a cost function to be minimized in conventional discriminative training methods, the discriminant function and the loss function are treated separately in the convex optimization scheme. Further, in order to maintain the convexity, L, $\mathcal{D}$, and R must be chosen from convex functions.

Recently, Jebara introduced MRED as a probabilistic interpretation of the previous formulation in order to properly model the variations in the estimation results, similar to that achieved in Bayesian inference methods [Jebara, 2001]. In MRED, by considering all variables as random variables, the effect of regularization and loss can be interpreted by using prior probability distribution function (pdf) of model parameters and slack variables. Further, the regularization function $R(\Theta)$ and the loss function $L(\xi^i)$ in Eq. (4.1) are represented by the Kullback-Leibler divergence (KL divergence or relative entropy) between the prior pdf $P^0(\Theta,\xi)$ and a posterior pdf $P(\Theta,\xi)$, as follows:

$$\begin{aligned} &\underset{P(\Theta,\xi)}{\text{minimize}}\ \mathrm{KL}[P(\Theta,\xi)||P^0(\Theta,\xi)], \\ &\text{subject to}\ \left\langle \mathcal{D}(\boldsymbol{X}^i;\Theta) - \xi^i \right\rangle_{P(\Theta,\xi)} \geq 0, \quad \forall i, \end{aligned} \tag{4.2}$$

where $\langle f(x)\rangle_{P(x)}$ denotes the expectation of $f(x)$ with respect to $P(x)$, i.e. $\langle f(x)\rangle_{P(x)} \overset{\mathrm{def}}{=} \int_x P(x)f(x)dx$; $\mathrm{KL}[f(x)||g(x)]$, the KL divergence of $g(x)$ from $f(x)$, i.e. $\mathrm{KL}[f(x)||g(x)] \overset{\mathrm{def}}{=} \langle \log f(x) - \log g(x)\rangle_{f(x)}$.

A comparison with the preceding formulation (Eq. (4.1)) shows that posterior pdfs of model parameters $P(\Theta)$ and slack variables $P(\xi)$ are optimized so that the expectations of discriminant functions are larger than the expectations of the corresponding slack variable.

Figure 4.1 shows a geometric interpretation of the MRED optimization. As shown in Figure 4.1, all possible pdfs can be embedded into a simplex in a Hilbert space where inner products (f, g) are defined as $\int_x f(x)g(x)dx$. Since the discriminative constraints are formulated by using expectation, these constraints can be regarded as linear constraints, regardless of a choice of a discriminant function, in the Hilbert space. Further, due to the convexity of the KL divergence, the optimization can be assumed as a convex optimization over the Hilbert space. One advantage of the convex optimization scheme is that the MRED solution can also be obtained by solving a Wolfe dual problem. The Wolfe dual problem of the primary problem (Eq. 4.2) is expressed as follows:

$$\underset{\alpha}{\text{maximize}}\ \ J(\alpha), \quad \text{subject to}\ \ \alpha^i \geq 0 \quad \forall i, \tag{4.3}$$

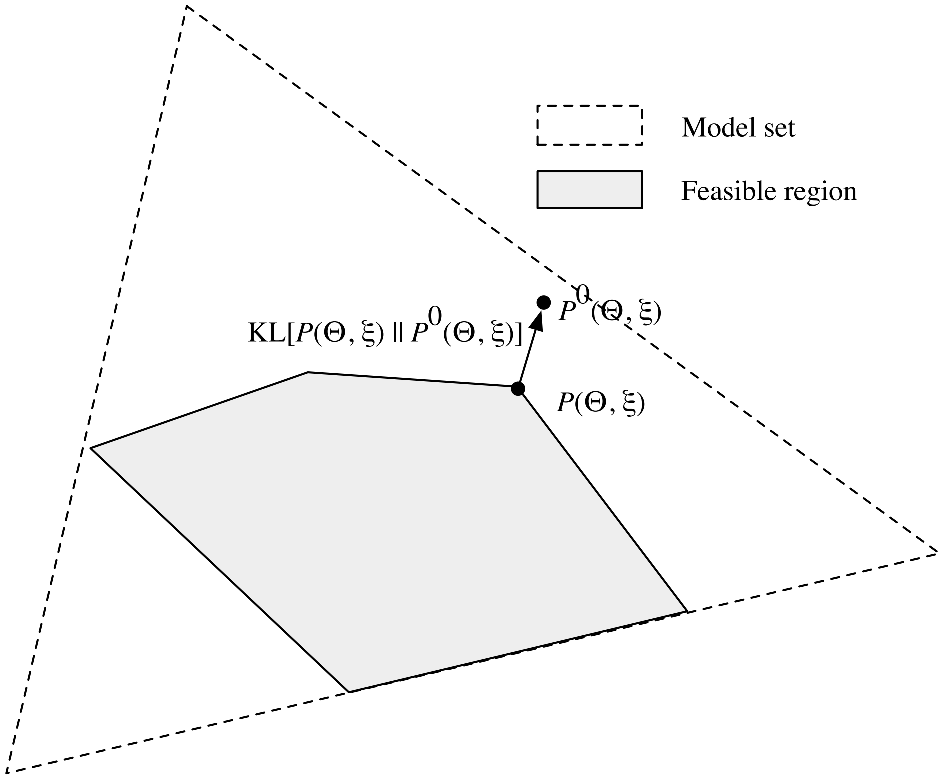


Figure 4.1 Geometric interpretation of the MRED optimization in a Hilbert space

where

$$\begin{aligned} J(\alpha) =& -\log Z(\alpha), \\ Z(\alpha) =& \left\langle \exp\left[\sum_i \alpha^i \left(\mathcal{D}(\boldsymbol{X}^i;\Theta) - \xi^i\right)\right]\right\rangle_{P^0(\Theta,\xi)}. \end{aligned} \tag{4.4}$$

Further, the optimal posterior pdf can be expressed by using the Lagrange multipliers $\alpha \overset{\text{def}}{=} \{\alpha^i|\forall i\}$ as follows:

$$P(\Theta,\xi|\alpha) = \frac{1}{Z(\alpha)} P^0(\Theta,\xi) \exp\left[\sum_i \alpha^i \left(\mathcal{D}(\boldsymbol{X}^i;\Theta) - \xi^i\right)\right]. \tag{4.5}$$

Detailed derivations of the posterior pdf and the dual-objective function are described in Appendix A.

The optimal Lagrange multipliers $\hat{\alpha}$ can be obtained as a solution of the optimization problem (Eq. (5.11)), and then The optimal $P(\Theta,\xi|\hat{\alpha},\hat{Q})$ is obtained by substituting α with the optimal Lagrange multipliers.

4.3 MRED for speech recognition

In order to apply MRED to speech recognition problems, discussions on the discriminant function $\mathcal{D}$ are presented at first. Then, the closed-form expression of the performance function $J(\alpha)$ in Eq. (5.11) is obtained by finding integrals with conjugate prior pdfs.

4.3.1 Discriminant function

Although there are many choices for the discriminant functions (e.g., MMI [Bahl et al., 1986], MCE [McDermott and Katagiri, 1997], or MPE [Povey and Woodland, 2002]), the following MCE-type discriminant function is chosen:

$$\mathcal{D}(\boldsymbol{X}^i;\Theta) = \log \frac{P(\boldsymbol{X}^i, \boldsymbol{l}^i|\Theta)}{\max_{\boldsymbol{l}\neq\boldsymbol{l}^i} P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta)}, \tag{4.6}$$

where

$$\begin{aligned} P(\boldsymbol{X}^i, \boldsymbol{l}|\Theta) =& P(\boldsymbol{l}) \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} \prod_n \mathcal{P}_{q_{n-1},q_n} \rho_{q_n,m_n} \mathcal{N}(\boldsymbol{x}_n^i | \boldsymbol{\mu}_{G(q_n,m_n)}, \boldsymbol{R}_{G(q_n,m_n)}) \\ =& P(\boldsymbol{l}) \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i|\boldsymbol{l},\Theta). \end{aligned} \tag{4.7}$$

Here, $\boldsymbol{q} = \{q_1, q_2, \cdots\}$ is a state sequence, $\boldsymbol{m} = \{m_1, m_2, \cdots\}$ is a mixture component sequence, and $G(s,m)$ is the numbering function that indicate the index of the Gaussian associated with m^{th} mixture component in s^{th} HMM state. The independency between parameters Θ and the label sequence $\boldsymbol{l}$ is assumed since the objective of this chapter is to estimate acoustic model parameters.

Here, the softmax function defined in Eq. (2.34) is introduced with $\eta = 1$ in order to make tractable $\max$ function in the previous discriminant function. The approximated discriminant function is as follows:

$$\begin{aligned}\mathcal{D}(\boldsymbol{X}^i;\Theta) &\approx \log P(\boldsymbol{X}^i,\boldsymbol{l}^i|\Theta) - \log \sum_{\boldsymbol{l}\neq\boldsymbol{l}^i} P(\boldsymbol{X}^i,\boldsymbol{l}|\Theta) \\ &= \log \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\boldsymbol{l}^i|\Theta) - \log \sum_{\boldsymbol{l}\neq\boldsymbol{l}^i} \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\boldsymbol{l}|\Theta).\end{aligned} \tag{4.8}$$

Since the summation over all possible erroneous label sequences $\boldsymbol{l} \neq \boldsymbol{l}^i$ is intractable in many cases, the lattice-based representations of error hypothesis are introduced. Examples of lattices are illustrated in Figure 4.2. Since lattices restrict possible word sequences, lattices can also be used to restrict possible state sequences [*2].

By introducing the lattice-based representations of the correct label sequence A^i and incor-

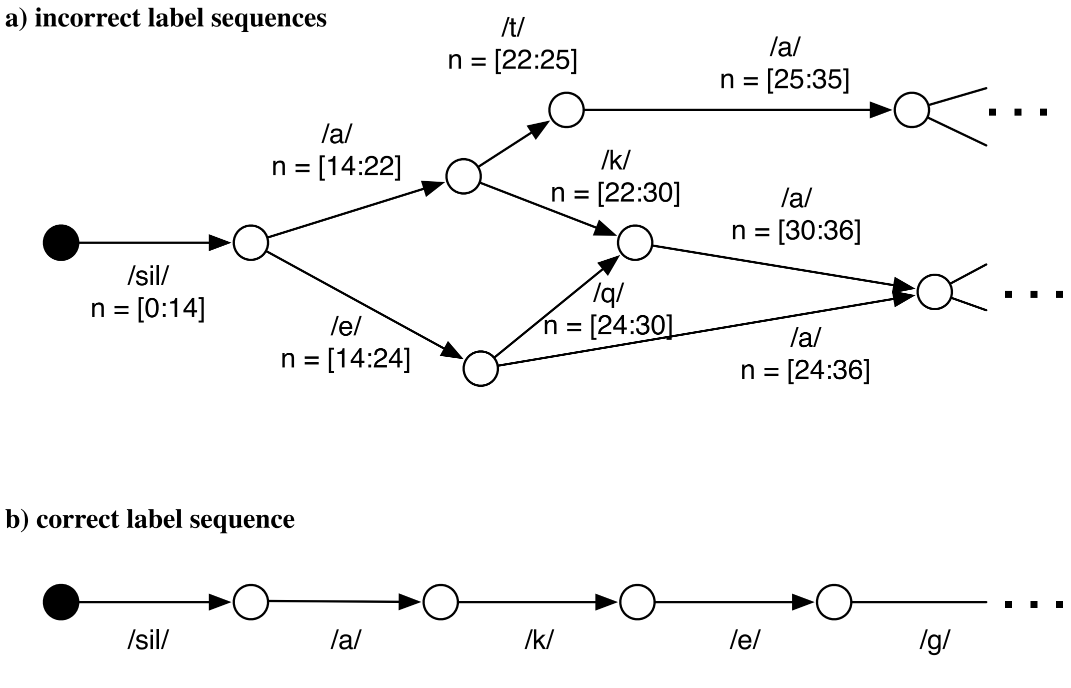


Figure 4.2 Examples of lattice-based representations of (a) incorrect label sequences, and (b) correct label sequence

[*2] In these examples, time alignment information is specified in the lattice (a). In such cases, possible state sequences are more restricted.

rect label sequences $\tilde{A}^i$, the approximated discriminant function can be expressed as follows:

$$\begin{aligned}
&\log \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l}^i)} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\boldsymbol{l}^i|\Theta) - \log \sum_{\boldsymbol{l}\neq\boldsymbol{l}^i} \sum_{\boldsymbol{q}\in\mathcal{S}(\boldsymbol{l})} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\boldsymbol{l}|\Theta)\\
\approx &\underbrace{\log \sum_{\boldsymbol{q}\in\mathcal{S}(A^i)} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta)}_{\mathcal{L}(X^i,A^i;\Theta)} - \underbrace{\log \sum_{\boldsymbol{q}\in\mathcal{S}(\tilde{A}^i)} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta)}_{\mathcal{L}(X^i,\tilde{A}^i;\Theta)} \\
\overset{\text{def}}{=} &\tilde{\mathcal{D}}(X^i,A^i,\tilde{A}^i;\Theta)
\end{aligned} \tag{4.9}$$

where $\mathcal{S}(A)$ is another parametrization of the $\mathcal{S}(\boldsymbol{l})$ that returns a set of possible state sequences with respect to the given lattice A, and $\mathbb{L}(\boldsymbol{q})$ is a label sequence corresponding to the given state sequence $\boldsymbol{q}$.

Conventionally, lattices are obtained as interim results of decoders. However, since the lattices obtained from decoders represent hypothesis label sequences, these lattices often include the correct label sequence. Therefore, the lattice-based representation of incorrect label sequences $\tilde{A}^i$ is obtained by removing the correct label sequence from these lattices. The removing operation can be performed by using difference operation for finite-state transducers [Allauzen et al., 2007].

By exploiting the Jensen's inequality, the lattice-based log-likelihood function $\mathcal{L}$ can be expressed as follows:

$$\begin{aligned}
\mathcal{L}(X^i,A;\Theta) &= \log \sum_{\boldsymbol{q}\in\mathcal{S}(A)} \sum_{\boldsymbol{m}} P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta)\\
&= \log \sum_{\boldsymbol{q}\in\mathcal{S}(A)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m}) \frac{P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta)}{Q(\boldsymbol{q},\boldsymbol{m})}\\
&= \max_Q \sum_{\boldsymbol{q}\in\mathcal{S}(A)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m}) \log \frac{P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta)}{Q(\boldsymbol{q},\boldsymbol{m})}\\
&= \max_Q \sum_{\boldsymbol{q}\in\mathcal{S}(A)} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m}) \log P(\boldsymbol{q},\boldsymbol{m},\boldsymbol{X}^i,\mathbb{L}(\boldsymbol{q})|\Theta) - H[Q(\boldsymbol{q},\boldsymbol{m})]\\
&\overset{\text{def}}{=} \max_Q \tilde{\mathcal{L}}(\boldsymbol{X}^i,A;\Theta,Q)
\end{aligned} \tag{4.10}$$

where $H[.]$ is the entropy functional, defined as follows:

$$H[Q(\boldsymbol{q},\boldsymbol{m})] = -\sum_{\boldsymbol{q}} \sum_{\boldsymbol{m}} Q(\boldsymbol{q},\boldsymbol{m}) \log Q(\boldsymbol{q},\boldsymbol{m}). \tag{4.11}$$

By substituting this representation of log-likelihood (Eq. (4.10)) and the lattice-based discriminant function (Eq. (4.9)) into the primary problem (Eq. (4.2)), the following expression of the primary problem is obtained.

$$\begin{aligned}&\underset{P(\Theta,\xi)}{\text{minimize}}\ \mathrm{KL}[P(\Theta,\xi)||P^0(\Theta,\xi)],\\ &\text{subject to } \left\langle \max_{Q} \tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, Q) - \max_{\tilde{Q}} \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, \tilde{Q}) - \xi^i \right\rangle_{P(\Theta,\xi)} \geq 0 \quad \forall i.\end{aligned} \tag{4.12}$$

In order to make integration tractable, the expectation of maxima is approximated by the maximum of the expectation, as follows:

$$\begin{aligned}&\underset{P(\Theta,\xi)}{\text{minimize}}\ \mathrm{KL}[P(\Theta,\xi)||P^0(\Theta,\xi)],\\ &\text{subject to } \max_{Q} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, Q) \right\rangle_{P(\Theta,\xi)} - \max_{\tilde{Q}} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, \tilde{Q}) \right\rangle_{P(\Theta,\xi)} - \left\langle \xi^i \right\rangle_{P(\Theta,\xi)}\\ &\qquad \geq 0, \quad \forall i.\end{aligned} \tag{4.13}$$

Here, by introducing the infinite set $\mathcal{Q}^i \overset{\mathrm{def}}{=} \{\tilde{Q}^i_1, \tilde{Q}^i_2, \cdots, \tilde{Q}^i_o, \cdots\}$ of all possible $\tilde{Q}$ with respect to i^{th} training datum, the above primary problem can be expressed as follows:

$$\begin{aligned}&\underset{P(\Theta,\xi)}{\text{minimize}}\ \mathrm{KL}[P(\Theta,\xi)||P^0(\Theta,\xi)],\\ &\text{subject to } \max_{Q} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, Q) \right\rangle_{P(\Theta,\xi)} - \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, \tilde{Q}^i_o) \right\rangle_{P(\Theta,\xi)} - \left\langle \xi^i \right\rangle_{P(\Theta,\xi)} \geq 0,\\ &\qquad \forall i, \forall o.\end{aligned} \tag{4.14}$$

The optimal point of this primary problem is defined by introducing a set of Lagrange multiplier $\alpha \overset{\mathrm{def}}{=} \{\alpha^i_o | \forall i, \forall o\}$, as follows:

$$\begin{aligned}P(\Theta,\xi|\alpha,\hat{Q}) \propto& P^0(\Theta,\xi) \exp\left\{ \sum_i \sum_o \alpha^i_o \left(\tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, \hat{Q}^i) - \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, \tilde{Q}^i_o) - \xi^i \right) \right\},\\ \hat{Q}^i =& \underset{Q}{\mathrm{argmax}} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, Q) \right\rangle_{P(\Theta,\xi)}.\end{aligned} \tag{4.15}$$

Further, the dual-objective function of the above primary function is as follows:

$$\begin{aligned}&\underset{\alpha}{\text{maximize}}\ J(\alpha,\hat{Q}) = -\log Z(\alpha,\hat{Q}),\\ &\text{subject to } \alpha^i_o \geq 0 \quad \forall i\ \forall o,\end{aligned} \tag{4.16}$$

where

$$Z(\alpha,\hat{Q}) = \left\langle \exp\left\{ \sum_i \sum_o \alpha^i_o \left(\tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, \hat{Q}^i) - \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, \tilde{Q}^i_o) - \xi^i \right) \right\} \right\rangle_{P^0(\Theta,\xi)}. \tag{4.17}$$

Here, $\hat{Q}$ is defined as $\hat{Q} = \{\hat{Q}^1, \hat{Q}^2, \cdots, \hat{Q}^i, \cdots\}$. The infinite constraints appeared in the dual optimization formulation can be handled by employing a cutting plane method (described in Section 4.4). Finally, the tractable dual-objective function is obtained. The next section derives a closed-form expression of the objective function by introducing conjugate prior pdfs.

4.3.2 Prior pdfs and a closed-form expression of the objective function

Here, in order to derive a closed-form expression, the independence of each parameter is assumed as follows:

$$P^0(\Theta, \xi) \stackrel{\text{def}}{=} \prod_i P^0(\xi^i) \times \prod_g P^0(\boldsymbol{\mu}_g, \boldsymbol{R}_g) \times \prod_s P^0(\boldsymbol{\rho}_s) \times \prod_s P^0(\boldsymbol{\mathcal{P}}_s), \tag{4.18}$$

where $\boldsymbol{\mathcal{P}}_s$ denotes the s^{th} row in a transition matrix $\mathcal{P}$. By substituting Eqs. (4.10) and (4.18) into Eq. (4.17), the following decomposed objective function is obtained.

$$\begin{aligned} Z(\alpha, \hat{Q}) =& \prod_g Z_g^{\texttt{EMIS}}(\alpha, \hat{Q}) \times \prod_s Z_s^{\texttt{MIX}}(\alpha, \hat{Q}) \times \prod_s Z_s^{\texttt{TR}}(\alpha, \hat{Q}) \times \prod_i J_i^{\texttt{SLACK}}(\alpha, \hat{Q}), \\ J(\alpha, \hat{Q}) =& -\log Z(\alpha, \hat{Q}) \\ =& \sum_g J_g^{\texttt{EMIS}}(\alpha, \hat{Q}) + \sum_s J_s^{\texttt{MIX}}(\alpha, \hat{Q}) + \sum_s J_s^{\texttt{TR}}(\alpha, \hat{Q}) + \sum_i J_i^{\texttt{SLACK}}(\alpha, \hat{Q}). \end{aligned} \tag{4.19}$$

The following paragraphs find the integral for each term in the above decomposed objective function.

■**Gaussian parameters ($J^{\texttt{EMIS}}$)** In order to obtain a closed-form expression of the objective function, conjugate pdfs are used as prior pdfs of the model parameters. The conjugate pdfs for the parameters of Gaussian pdfs (with diagonal covariance matrices [*3]) are represented by the normal-gamma distribution $\mathcal{N} \circ \mathcal{G}(.)$ as follows:

$$P^0(\mu_{g,d}, r_{g,d}) = \mathcal{N} \circ \mathcal{G}(\mu_{g,d}, r_{g,d} | \mu_{g,d}^0, \gamma_{g,d}^0, \eta_g^0, \beta_{g,d}^0), \tag{4.20}$$

where $\mu_{g,d}$ and $r_{g,d}$ are the mean and the variance of d^{th} dimension of g^{th} Gaussian distribution. The normal-gamma distribution is defined as follows:

$$\mathcal{N} \circ \mathcal{G}(\mu, r | \mu^0, \gamma^0, \eta^0, \beta^0) \propto \frac{(\beta^0)^{\eta^0}}{\Gamma(\eta^0)} (r)^{\eta^0 - 1/2} \exp\left\{ -\beta^0 r - \frac{r\gamma^0}{2} (\mu^0 - \mu)^2 \right\}. \tag{4.21}$$

[*3] By using the normal-Wishart distribution [Bishop, 2006] as a prior pdf, full covariance Gaussian pdfs are also tractable in this method.

By using this conjugate prior pdf, the integration over the model parameters in $J^{\mathtt{EMIS}}(\alpha, \hat{Q})$ in Eq. (4.19) can be solved as follows:

$$\begin{aligned}
J_g^{\mathtt{EMIS}}(\alpha, \hat{Q}) =& \sum_d J_{g,d}^{\mathtt{EMIS}}(\alpha, \hat{Q}) \\
J_{g,d}^{\mathtt{EMIS}}(\alpha, \hat{Q}) =& \frac{\gamma_g(\alpha, \hat{Q})}{2} \log\{2\pi\} - \log\left\{\Gamma(\eta_g(\alpha, \hat{Q}))\right\} \\
&+ \frac{1}{2} \log\left\{\xi_g(\alpha, \hat{Q})\right\} + \eta_g(\alpha, \hat{Q}) \log\left\{r_{g,d}(\alpha, \hat{Q})\right\},
\end{aligned} \tag{4.22}$$

where the followings are parameters of posteriors, as functions of α and $\hat{Q}$, as follows:

$$\begin{aligned}
\eta_g(\alpha, \hat{Q}) =& \eta_g^0 + \frac{\Delta_g^0(\alpha, \hat{Q})}{2}, \\
\gamma_g(\alpha, \hat{Q}) =& \gamma_n^0 + \Delta_g^0(\alpha, \hat{Q}), \\
\mu_{g,d}(\alpha, \hat{Q}) =& \frac{\gamma_g^0 \mu_{g,d}^0 + \Delta_{g,d}^1}{\gamma_g^0 + \gamma_g(\alpha, \hat{Q})}, \\
\beta_{g,d}(\alpha, \hat{Q}) =& \beta_{g,d}^0 + \frac{1}{2}\left(\gamma_g^0 \left(\mu_{g,d}^0\right)^2 + \Delta_{g,d}^2 - \gamma_g(\alpha, \hat{Q}) \mu_{g,d}(\alpha, \hat{Q})^2\right),
\end{aligned} \tag{4.23}$$

Here, the followings are difference between statistics obtained from Q^i and $\tilde{Q}_o^i$, as follows:

$$\begin{aligned}
\Delta_g^0(\alpha, \hat{Q}) =& \sum_i \sum_o \alpha_o^i \left(\chi_g^0(\boldsymbol{X}^i; \hat{Q}^i) - \chi_g^0(\boldsymbol{X}^i; \tilde{Q}_o^i)\right), \\
\Delta_{g,d}^1(\alpha, \hat{Q}) =& \sum_i \sum_o \alpha_o^i \left(\chi_{g,d}^1(\boldsymbol{X}^i; \hat{Q}^i) - \chi_{g,d}^1(\boldsymbol{X}^i; \tilde{Q}_o^i)\right), \\
\Delta_{g,d}^2(\alpha, \hat{Q}) =& \sum_i \sum_o \alpha_o^i \left(\chi_{g,d}^2(\boldsymbol{X}^i; \hat{Q}^i) - \chi_{g,d}^2(\boldsymbol{X}^i; \tilde{Q}_o^i)\right),
\end{aligned} \tag{4.24}$$

where g is an index for Gaussian distributions, d is a dimensionality index, and $\chi^0(\boldsymbol{X}^i; Q), \chi^1(\boldsymbol{X}^i; Q)$, and $\chi^2(\boldsymbol{X}^i; Q)$ are occupancy, 1st-order statistics, and 2nd-order statistics of the feature vector sequence $\boldsymbol{X}^i$ and the hidden variable distribution Q with respect to d^{th} dimension of the n^{th} Gaussian pdf [*4], defined as follows:

$$\begin{aligned}
\chi_g^0(\boldsymbol{X}^i; Q) =& \sum_{\boldsymbol{q}} \sum_{\boldsymbol{m}} Q(\boldsymbol{q}, \boldsymbol{m}) \sum_n \mathbb{1}(q_n, g), \\
\chi_{g,d}^1(\boldsymbol{X}^i; Q) =& \sum_{\boldsymbol{q}} \sum_{\boldsymbol{m}} Q(\boldsymbol{q}, \boldsymbol{m}) \sum_n \mathbb{1}(q_n, g) x_{n,d}^i, \\
\chi_{g,d}^2(\boldsymbol{X}^i; Q) =& \sum_{\boldsymbol{q}} \sum_{\boldsymbol{m}} Q(\boldsymbol{q}, \boldsymbol{m}) \sum_n \mathbb{1}(q_n, g) \left(x_{n,d}^{(i)}\right)^2
\end{aligned} \tag{4.25}$$

[*4] Although this definition of the sufficient statistics functions is different from the definition in Chapter 2, both of the definitions are compatible, and calculated by the forward-backward algorithm.

Δ^0, Δ^1, and Δ^2 are the weighted sums of the differences between the sufficient statistics obtained from the reference lattice A^i and the sufficient statistics obtained from the competitor lattice $\tilde{A}^i$. The differences in the sufficient statistics are also used in conventional discriminative training methods in which all the difference statistics are accumulated with the same weight.

■**Mixture weights ($J^{\texttt{MIX}}$) / transition probability ($J^{\texttt{TR}}$)** Similar to the case of the Gaussian parameters, the conjugate prior pdfs are introduced as $P^0(\rho_s)$ and $P^0(\mathcal{P}_s)$. Since mixture weight vectors and rows in transition probability matrices can be assumed as discrete pdfs, the Dirichlet distribution is suitable as a conjugate prior pdf of these parameters. Therefore, the Dirichlet pdfs are introduced as the prior pdfs for $\mathcal{P}_s \stackrel{\text{def}}{=} \{\mathcal{P}_{s,s'}|\forall s'\}$ and $\boldsymbol{\rho}_s \stackrel{\text{def}}{=} \{\rho_{s,m}|\forall m\}$ as follows:

$$\begin{aligned} P^0(\boldsymbol{\rho}_s|\boldsymbol{\phi}^0_s) =& \frac{\Gamma(\sum_m \phi^0_{s,m})}{\prod_n \Gamma(\phi^0_{s,m})} \prod_m (\rho_{s,m})^{\phi^0_{s,m}}, \\ P^0(\mathcal{P}_s|\boldsymbol{\varphi}^0_s) =& \frac{\Gamma(\sum_{s'} \varphi^0_{s,s'})}{\prod_n \Gamma(\varphi^0_{s,s'})} \prod_m (\pi_{s,s'})^{\varphi^0_{s,s'}}. \end{aligned} \tag{4.26}$$

By substituting (4.26) into (4.17), the following closed-form expressions are obtained:

$$\begin{aligned} Z^{\texttt{MIX}}_s(\alpha, \hat{Q}) \propto& \frac{\Gamma\left(\sum_m \phi_{s,m}(\alpha, \hat{Q})\right)}{\prod_m \Gamma\left(\phi_{s,m}(\alpha, \hat{Q})\right)}, \\ Z^{\texttt{TR}}_s(\alpha, \hat{Q}) \propto& \frac{\Gamma\left(\sum_{s'} \varphi_{s,s'}(\alpha, \hat{Q})\right)}{\prod_{s'} \Gamma\left(\varphi_{s,s'}(\alpha, \hat{Q})\right)}. \end{aligned} \tag{4.27}$$

Here, $\phi_{s,m}(\alpha, \hat{Q})$ and $\varphi_{s,s'}(\alpha, \hat{Q})$ are parameters of the posterior pdfs, obtained as follows:

$$\begin{aligned} \phi_{s,m}(\alpha, \hat{Q}) =& \phi^0_{s,m} + \Delta^{\texttt{MIX}}_{s,m}(\alpha, \hat{Q}) \\ \varphi_{s,s'}(\alpha, \hat{Q}) =& \varphi^0_{s,s'} + \Delta^{\texttt{TR}}_{s,s'}(\alpha, \hat{Q}) \end{aligned} \tag{4.28}$$

where

$$\begin{aligned} \Delta^{\texttt{MIX}}_{s,m}(\alpha, \hat{Q}) =& \Delta^0_{G(s,m)}(\alpha, \hat{Q}), \\ \Delta^{\texttt{TR}}_{s,s'}(\alpha, \hat{Q}) =& \sum_i \sum_o \alpha^i_o \left(\chi^{\texttt{TR}}_{s,s'}(\boldsymbol{X}^i; \hat{Q}^i) - \chi^{\texttt{TR}}_{s,s'}(\boldsymbol{X}^i, \tilde{Q}^i)\right) \end{aligned} \tag{4.29}$$

By using the above expressions, the corresponding terms of the objective function can be expressed as follows:

$$
\begin{aligned}
J_s^{\mathtt{MIX}}(\alpha, \hat{Q}) &= -\log\Gamma\left(\sum_m \phi_{s,m}(\alpha,\hat{Q})\right) + \sum_m \log\Gamma\left(\phi_{s,m}(\alpha,\hat{Q})\right), \\
J_s^{\mathtt{TR}}(\alpha, \hat{Q}) &= -\log\Gamma\left(\sum_{s'} \varphi_{s,s'}(\alpha,\hat{Q})\right) + \sum_{s'} \log\Gamma\left(\varphi_{s,s'}(\alpha,\hat{Q})\right).
\end{aligned}
\tag{4.30}
$$

■**Slack variables ($J^{\mathtt{SLACK}}$)** In order to utilize a hinged linear penalty function as used in soft-margin SVMs, an exponential distribution is used as a prior pdf of slack variables, as follows:

$$
P^0(\xi^i) = \frac{1}{c^0}\exp\left\{-c^0\left(\xi^i - \delta^i\right)\right\}, \quad (\xi^i \le l^i).
\tag{4.31}
$$

Here, c^0 is a hyper parameter that adjusts a proportion of penalty. δ^i is a hyper parameter that represents a threshold of the hinged linear penalty function. Figure 4.3 shows the likelihood function and the log-likelihood function of this prior pdf ($\delta^i = 1$, $c^0 = 2.0$).

Similar to the cases of the model parameters, the integration in $Z_i^{\mathtt{SLACK}}(\alpha, \hat{Q}) = \exp(-J_i^{\mathtt{SLACK}}(\alpha, \hat{Q}))$ in Eq. (4.19) is analytically solved by using this prior distribution, as follows:

$$
\begin{aligned}
J_i^{\mathtt{SLACK}}(\alpha, \hat{Q}) &= -\sum_o \alpha_o^i\left(\Delta^{\mathtt{SHIFT}}(\hat{Q}, i, o) - \delta^i\right) + \log\left(c^0 - \sum_o \alpha_o^i\right), \\
\Delta^{\mathtt{SHIFT}}(\hat{Q}, i, o) &= H[\hat{Q}^i] - H_q[\tilde{Q}_o^i] + \sum_{\boldsymbol{q}\in\mathcal{S}(A^i)} \log P(\mathbb{L}(\boldsymbol{q})) - \sum_{\boldsymbol{q}\in\mathcal{S}(\tilde{A}^i)} \log P(\mathbb{L}(\boldsymbol{q})).
\end{aligned}
\tag{4.32}
$$

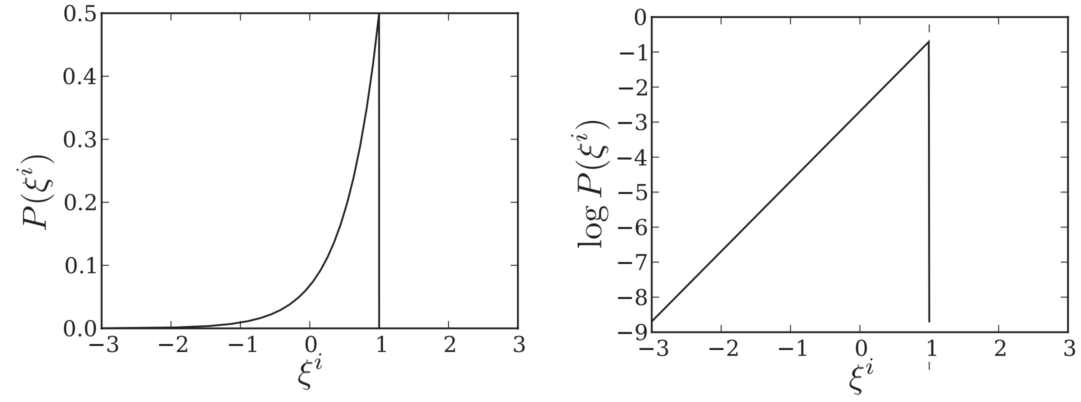


Figure 4.3 Probability density function defined in Eq. (4.31) ($\delta^i = 1$, $c^0 = 2.0$)

4.3.3 Parameter update

By substituting the above prior settings into Eq. (4.15), the following posterior pdf is obtained.

$$\begin{aligned} P(\Theta|\alpha, \hat{Q}) = &\prod_g \mathcal{N} \circ \mathcal{G}(\mu_{g,d}, r_{g,d}|\mu_{g,d}(\alpha, \hat{Q}), \gamma_{g,d}(\alpha, \hat{Q}), \eta_g(\alpha, \hat{Q}), \beta_{g,d}(\alpha, \hat{Q})) \\ &\times \prod_s \mathrm{Dir}(\boldsymbol{\rho}_s|\boldsymbol{\phi}_s(\alpha, \hat{Q})) \times \prod_s \mathrm{Dir}(\boldsymbol{\mathcal{P}}_s|\boldsymbol{\varphi}_s(\alpha, \hat{Q})) \end{aligned} \tag{4.33}$$

where

$$\begin{aligned} \boldsymbol{\phi}_s(\alpha, \hat{Q}) = &[\phi_{s,1}(\alpha, \hat{Q}), \cdots, \phi_{s,m}(\alpha, \hat{Q}), \cdots]^{\mathsf{T}}, \\ \boldsymbol{\varphi}_s(\alpha, \hat{Q}) = &[\varphi_{s,1}(\alpha, \hat{Q}), \cdots, \varphi_{s,s'}(\alpha, \hat{Q}), \cdots]^{\mathsf{T}}. \end{aligned} \tag{4.34}$$

Here, $\mathrm{Dir}(.|.)$ denotes the pdf of the Dirichlet distribution, defined as follows:

$$\mathrm{Dir}(\boldsymbol{\rho}|\boldsymbol{\phi}) = \frac{\Gamma\left(\sum_{d'} \phi_{d'}\right)}{\prod_{d'} \Gamma\left(\phi_{d'}\right)} \prod_d \left(\rho_d\right)^{\phi_d}. \tag{4.35}$$

As mentioned above, the posterior pdf $P(\Theta)$ is obtained by substituting α with the optimal $\hat{\alpha}$ into Eq. (4.33). It should be noted that this posterior pdf is determined so that the differences between the log-likelihoods of the correct label sequence and incorrect label sequences are sufficiently large. Although the Student's t-distribution is often used as the expectation of the likelihood function over the posterior pdf in Bayesian approach, Student's t-distribution may be inconsistent since the posterior pdf is obtained under the constraints with respect to log-likelihood function. Therefore, the maximum-a-posteriori parameters are used, in which the modes of posterior distributions serve as the estimated parameters.

4.3.4 Empirical prior setting

This section focuses on the determination of hyper parameters. In MRED, the empirically estimated hyper parameters are often used in conjugate priors. Here, in order to guarantee the performance of estimated models, hyper parameters are defined by using the sufficient

Algorithm 1 Iterative optimization algorithm of $\hat{Q}$ and α

```
$O \leftarrow 0$
loop
    optimize $\hat{Q}$ with the given $\alpha_o^i$ $(o < O)$ where $\alpha_o^i = 0$ $(o \geq O)$
    $O \leftarrow O + 1$
    determine $\tilde{Q}_{o=O}^i$
    optimize $\alpha_o^i (o \leq O)$ with the given $\hat{Q}$
end loop
```

statistics obtained by the maximum-likelihood parameters, as follows:

$$\begin{aligned}
\gamma_g^0 =& \sum_i \chi_g^0(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}}), \\
\mu_{g,d}^0 =& \frac{\sum_i \chi_{g,d}^1(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}})}{\gamma_g^0}, \\
\alpha_g =& \frac{1}{2}\gamma_g^0 \\
\beta_{g,d} =& \frac{1}{2}\sum_i \chi_{g,d}^2(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}}) - \alpha_n \left(\mu_{g,d}^0\right)^2,
\end{aligned} \tag{4.36}$$

where $\Theta^{\mathtt{MLE}}$ is a parameter set obtained by using the maximum likelihood procedure, the definitions of $\chi_g^0(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}})$, $\chi_{g,d}^1(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}})$, and $\chi_{g,d}^2(\boldsymbol{X}^i, \boldsymbol{l}^i; \Theta^{\mathtt{MLE}})$, are the same as those in Eq. (2.26).

4.4 Optimization

In this section, an optimization method for the objective function is discussed and proposed. Since the dual-optimization problem defined in Eq. (4.17) involves two different optimizations, that is, the optimization with respect to α and the optimization with respect to $\hat{Q}$, the optimization must be solved by using an iterative scheme. Further, due to the infinite constraints in the primary problem (Eq. (4.14)), the number of Lagrange multipliers α_o^i is infinite. In order to handle the infinite constraints, a cutting plane method [Tsochantaridis et al., 2005] is adapted to this iterative iteration scheme. The Algorithm 1 shows an iterative optimization algorithm based on a cutting plane method.

■**$\hat{Q}$-optimization** In order to prevent combinatorial explosion of the discrete pdf Q^i, sufficient statistics $\chi(\boldsymbol{X}^i; Q^i) \overset{\text{def}}{=} \{\chi_g^0(\boldsymbol{X}^i; Q^i), \chi_{g,d}^1(\boldsymbol{X}^i; Q^i), \chi_{g,d}^2(\boldsymbol{X}^i; Q^i) | \forall g, \forall d\}$ is used to represent Q^i. Because the $\hat{Q}$-optimization with fixed α is equivalent to the maximum likelihood optimization of $\hat{Q}$ with fixed Θ', the optimal sufficient statistics can be obtained by using the forward-backward algorithm as in the EM algorithm.

■**Determination of $\tilde{Q}_o^i$** A cutting plane method is performed by finding and considering a constraint that is considered as the most critical constraint [Tsochantaridis et al., 2005]. The most critical constraint can be obtained by using the current hypothesis of the model $(\alpha, \hat{Q})$, as follows:

$$\begin{aligned}\tilde{Q}_o^i =& \underset{Q'}{\text{argmin}} \max_{Q} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, A^i; \Theta, Q) \right\rangle_{P(\Theta,\xi|\alpha,\hat{Q})} - \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, Q') \right\rangle_{P(\Theta,\xi|\alpha,\hat{Q})} \\ & - \left\langle \xi^i \right\rangle_{P(\Theta,\xi|\alpha,\hat{Q})} \\ =& \underset{Q'}{\text{argmax}} \left\langle \tilde{\mathcal{L}}(\boldsymbol{X}^i, \tilde{A}^i; \Theta, Q') \right\rangle_{P(\Theta,\xi|\alpha,\hat{Q})} .\end{aligned} \tag{4.37}$$

As shown in the above equation, the optimal $\tilde{Q}_o^i$ corresponding to the most critical constraints can be obtained by maximizing log-likelihood with respect to the lattice $\tilde{A}^i$. Thus, the sufficient statistics, which represents $\tilde{Q}_o^i$, can be obtained by using the forward-backward algorithm.

■**α-optimization** The α optimization can be solved by using several optimization methods. For example, a gradient-based method can be used for this optimization. Because the α-optimization is a convex optimization when $\hat{Q}$ is fixed, this optimization is ensured to reach the global optimum.

The detailed implementation of this optimization is discussed in Appendix B.

4.5 Experimental setup

In the experiments, 3,696 sentences from the TIMIT database [Lamel et al., 1986] are used for model training, and 192 sentences are used for evaluation. All the training and test speeches are parametrized by Mel-frequency cepstral coefficients (MFCC) and its energy augmented by their derivatives and accelerations (MFCC_E_D_A; 39 dims.) computed at a 10 ms frame shift with 25 ms window size (cf. Section 2.1.1).

As described in [Lee and Hon, 1989], we used 48 phonetic classes for the training and decoding, and the phoneme accuracies were calculated by using 39 broader phonetic categories. A bi-gram (bi-phoneme) grammar model is applied during all decoding processes. Proportion for grammar models is set at 5.

For comparison, the discriminative training methods are performed by optimizing the linear-loss MCE criterion, and the MMIE criterion. In general, MCE is not optimized by the EBW method. However, because large number of similarities in the implementation of our method and those of EBW method, the MCE system is optimized by the EBW method. The baseline and initial models for MCE/EBW and MMIE were trained by a maximum likelihood Viterbi training procedure.

Table. 4.1 Phoneme error rates of the compared methods

Method	1 mix.	2 mix.
MLE	42.2	38.8
MCE/EBW (1-best)	40.4	–
MRED (1-best)	39.5	–
MMIE (Lattice)	39.2	36.7
MRED (Lattice)	38.9	36.0

4.6 Discussions

Table 4.1 shows the phoneme error rates of the compared methods. It is confirmed that MRED outperforms the conventional MCE/EBW method and the MMIE method. In the table, the best results obtained by varying the number of iterations are presented. However, it is confirmed that continuing iterations declines the performance of the MCE models. Contrastingly, it is confirmed that the performances of MRED models rarely decay with iterations.

It is considered that regularization techniques by employing empirical priors for model parameters lead to improvements in phoneme accuracy. Although, in general, the use of priors increases the number of hyper parameters to be tuned in advance, the use of empirical priors does not necessitate that kind of tunings.

Chapter 5

Feature augmentation based on hidden Markov kernel machines

This chapter discusses on regularized discrimination of high-dimensional features obtained by kernel methods. Recently, HMM/ MLP-tandem approaches are also regarded as a kernel-like transformation in several articles [Collobert and Bengio, 2004, Malkin et al., 2009]. Although the MLP-based approach is accepted widely in ASR, kernel based methods are not discussed enough since kernel based methods necessitate modifications in training procedures. In this chapter, the author pointed out that a kernel based method can be obtained by using the training method described in the previous chapter (Chapter 4). Furthermore, a simple Viterbi-path based approximation method of MRED is presented for computational efficiency.

5.1 Introduction

Hidden Markov models (HMMs) have been widely used in classification problems of sequential data, such as speech recognition, speaker recognition, handwriting recognition, and gesture recognition because of their extensibility. In such classification problems, nonlinear classification techniques are essential because feature vectors are not linearly separable in the natural feature space. To deal with such nonseparable sequences, kernel-based nonlinear classification techniques have been especially developed based on support vector machines (SVMs) [Boser et al., 1992, Vapnik, 1999].

Several approaches can be used for carrying out kernel-based classification of sequential data based on SVMs [Tsochantaridis et al., 2005, Ganapathiraju et al., 2004, Joder et al., 2008], such as an approach that involves the use of SVMs with a kernel function that directly handles sequential data (sequential kernel) [Joder et al., 2008, Shimodaira et al., 2002, Cuturi et al., 2007], and SVM/ HMM hybrid approaches

[Huang et al., 2006, Ganapathiraju et al., 2000] that involve the use of SVMs as static classifiers for the fixed alignment segments determined by HMMs. However, these approaches cannot explicitly hold the HMM representation, which makes it difficult to integrate them to large-scale systems straightforwardly. Because of the abovementioned lack of the conventional SVM-based approaches, HMM-based approaches are still used in some sequential pattern classification problems, especially in speech recognition. In order to carry out a nonlinear classification, the current state-of-the-art HMM-based sequential classifiers introduce several discriminative training methods to HMMs as described in Section 2.2.4, and use Gaussian mixture models (GMMs) with a large number of mixture components as emission probability density functions (pdfs) of HMMs. Since GMMs are capable of representing arbitrary pdfs, increasing the number of mixture components in GMMs could, in principle, lead to optimal nonlinear classification. However, the risk of local optima and overfitting also arises with an increase in the number of mixture components. The objective of this chapter is to prevent these risks by enhancing the emission pdfs of HMMs based on kernel methods.

In this chapter, a novel kernel machine is proposed for the classification of sequential data. Since the proposed method is formulated as a natural extension for conventional HMMs, our method can explicitly model the transition of hidden states behind the observed vectors. Therefore, the proposed method can be applied to many applications developed with conventional HMMs straightforwardly, especially for speech recognition. In addition, the proposed method can avoid the overfitting and local optima problems by using kernel-based nonlinear classification instead of mixture models.

Preliminary experiments that involved a phoneme classification task of speech data is performed to show the effectiveness of our proposed method. It should be noted that kernel-based methods for classification require, in principle, a computational cost of $O(p^3)$ for training and $O(pq)$ for evaluation, where p and q denote the numbers of frames in the training dataset and test dataset, respectively. Hence, evaluations on current standard corpora are prohibitive without any approximation, even if a small-sized corpus (e.g., TIMIT) is used for training and evaluation. In this chapter, as an initial attempt, the exact performance of the proposed kernel machines is focused by using a subset of the standard corpus (TIMIT). Since many approximation techniques aimed at the acceleration of kernel-based methods have been developed in the machine learning community [Kashima et al., 2009], it is considered that the proposed method can be scalably applied to large-scale corpora by applying appropriate approximation techniques. Therefore, a phoneme classification problem is chosen in our evaluations as a normal sequential pattern classification problem.

The remainder of this chapter is organized as follows. Section 5.2 briefly describes how kernel methods achieve nonlinear classification without mixture models. Section 5.3 defines the models used in the proposed method and a discriminant function that is the foundation for the proposed method. Section 5.4 describes a parameter estimation method and a method for introducing kernel techniques into the estimation process of the model parameters. In Section 5.5, the preliminary results of isolated phoneme classification experiments are presented and discussed.

5.2 Reproducing kernel Hilbert spaces

In this section, a method used to carry out nonlinear classification without using mixture models is described conceptually. The detailed formulations required for the application of this method to HMMs are described in Sections 5.3 and 5.4.

Conventional classifiers based on probabilistic models use pdfs $P(\boldsymbol{x})$ in the input feature space $\boldsymbol{x} \in \mathbb{R}^D$ as models of feature vectors. Although the classification boundaries constructed by Gaussian pdfs are quadratic surfaces in the input feature space $\mathbb{R}^D$, boundaries with higher-order nonlinearities are required in most applications. Therefore, to enhance the representation of emission pdfs, mixtures of Gaussian pdfs are often used to construct accurate boundaries. However, the use of mixtures might introduce the risk of local optima and overfitting.

The objective of this chapter is to construct classifiers in a higher-dimensional space $\mathcal{K} \overset{\text{def}}{=} \{\boldsymbol{\phi}(\boldsymbol{x})|\boldsymbol{x} \in \mathcal{X}\}$. It is well known that if an appropriate nonlinear warping function $\boldsymbol{\phi}$ is given, the optimal classification boundary can be represented as a linear function in a higher-dimensional space $\mathcal{K}$. Therefore, simple pdfs obtained without using mixture models (e.g., exponential distributions) can be used as models for warped feature vectors, as shown in Figure 5.1.

By using an appropriate kernel function $K : (\mathcal{X}, \mathcal{X}) \rightarrow \mathcal{R}$, there exists ϕ, which satisfies $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^\mathsf{T}\boldsymbol{\phi}(\boldsymbol{y})$. Therefore, $\boldsymbol{\phi}$ is not defined explicitly in general. If all operations in the higher-dimensional space $\mathcal{K}$ can be written by using inner products in $\mathcal{K}$, the explicit rep-

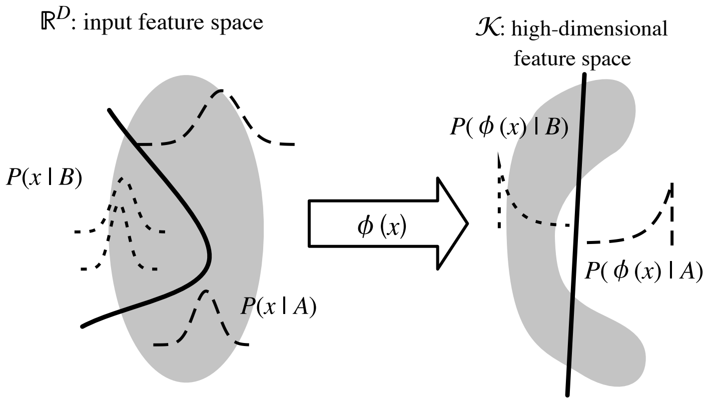

$\mathbb{R}^D$: input feature space
$\mathcal{K}$: high-dimensional feature space
$P(x \mid B)$
$P(x \mid A)$
$\phi(x)$
$P(\phi(x) \mid B)$
$P(\phi(x) \mid A)$

Figure 5.1 Basic concept of using probability density functions in reproducing kernel Hilbert space

resentation and computation of $\boldsymbol{\phi}$ can be omitted by substituting $\boldsymbol{\phi}(.)^{\mathsf{T}}\boldsymbol{\phi}(.)$ with $K(.,.)$. The higher-dimensional space $\mathcal{K}$ defined by the kernel function K is called the reproducing kernel Hilbert space (RKHS). SVMs, which are formulated as linear classifiers, achieve nonlinear classification by considering linear classification in an RKHS.

As an example, the average of warped feature vectors $\boldsymbol{\phi}(\boldsymbol{x}^i)$ in a dataset $\left\{\boldsymbol{\phi}(\boldsymbol{x}^i)|i \in [1..N]\right\}$ is discussed. The computation of the inner product between the average and an input vector $\boldsymbol{\phi}(\boldsymbol{x})$ can be expressed by using the kernel function K as follows:

$$\boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}} \left(\frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\phi}(\boldsymbol{x}^i) \right) = \frac{1}{N} \sum_{i=1}^{N} K(\boldsymbol{x}, \boldsymbol{x}^i). \tag{5.1}$$

Here, because of the summation ($\sum_{i=1}^{N}$) over K, the loop computation and storage attributed to all vectors $\boldsymbol{x}^i$ in the dataset are essential for kernel methods. This is the main cause of the computational complexity in kernel-based methods. However, several methods to eliminate this loop computation can be identified if an appropriate kernel function K is chosen [Kashima et al., 2009, Freitas et al., 2006].

5.3 Hidden Markov models with log-linear emission pdfs

In this section, a classifier is formulated by introducing HMMs as generative models and a discriminant function that indicates the classification performance of the models.

Model formulation described in this section includes explicit representation of feature warping function $\boldsymbol{\phi}$. Therefore, the straightforward implementation of the models described in this section might be impossible because the number of dimensions of $\boldsymbol{\phi}(\boldsymbol{x})$ might be infinite in general RKHSs. This problem is resolved by introducing a training method that can avoid the use of explicit representation of $\boldsymbol{\phi}(\boldsymbol{x})$; this method is described in Section 5.4.

5.3.1 Definition of discriminant function

Let $\mathcal{X} = \left\{\boldsymbol{X}^i|i \in [1..N(\mathcal{X})]\right\}$ and $\mathcal{L} = \left\{\boldsymbol{l}^i|i \in [1..N(\mathcal{L})]\right\}$ be sets of training data, where $N(\mathcal{X}) = N(\mathcal{L})$ is the number of examples in the training dataset. $\boldsymbol{X}^i$ is a sequence of D-dimensional feature vectors, i.e., $\boldsymbol{X}^i = \{\boldsymbol{x}_1^i, \cdots, \boldsymbol{x}_n^i, \cdots | \boldsymbol{x}_n^i \in \mathbb{R}^D\}$, and $\boldsymbol{l}^i$ is the corresponding label (phoneme or word) sequence (classifier outputs), i.e., $\boldsymbol{l}^i = \{l_1^i, l_2^i, \cdots\}$.

Conventional HMM-based sequential pattern classifiers can be used to obtain a classification result $\hat{\boldsymbol{l}}$ of an input feature sequence $\boldsymbol{X}$ by solving the following search problem:

$$\hat{\boldsymbol{l}} = \operatorname*{argmax}_{\boldsymbol{l}} \log P(\boldsymbol{l}|\boldsymbol{X}, \Theta), \tag{5.2}$$

where $\Theta \overset{\text{def}}{=} \{\lambda_s, \boldsymbol{\mathcal{P}}, \boldsymbol{\rho}_s | \forall s\}$ is a parameter of acoustic models. Since the objective of this study is to enhance the emission pdfs by kernel methods, the estimation of $\boldsymbol{\rho}$ and $\boldsymbol{\mathcal{P}}$ is not discussed here. Therefore, $\Lambda \stackrel{\text{def}}{=} \{\lambda_s | \forall s\}$ is used instead of θ for the sake of readability in the remainder of this chapter.

First, a parametric discriminant function is introduced in order to indicate the performance of the model parameter Λ. By considering $\boldsymbol{l} \neq \boldsymbol{l}^i$ as a possible sequence of labels that is different from the correct label sequence $\boldsymbol{l}^i$, the following pair-wise discriminant function $\mathcal{D}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda)$ is used:

$$\mathcal{D}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) \stackrel{\text{def}}{=} \log \frac{P(\boldsymbol{X}^i, \boldsymbol{l}^i | \Lambda)}{P(\boldsymbol{X}^i, \boldsymbol{l} | \Lambda)} \stackrel{\text{def}}{=} \log \frac{P(\boldsymbol{X}^i | \boldsymbol{l}^i, \Lambda) P(\boldsymbol{l}^i)}{P(\boldsymbol{X}^i | \boldsymbol{l}, \Lambda) P(\boldsymbol{l})}. \tag{5.3}$$

Here, it is assumed that the label sequences $(\boldsymbol{l}, \boldsymbol{l}^i)$ are independent of the parameters of emission pdfs Λ.

In this discriminant function, when $\boldsymbol{X}^i$ is misclassified into an incorrect word sequence $\boldsymbol{l} \neq \boldsymbol{l}^i$, it is found that $\mathcal{D}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda)$ is less than 0 (i.e., the denominator is greater than the numerator in Eq. (5.3)). Therefore, in order to eliminate misclassifications, Λ should be estimated such that $\mathcal{D}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) > 0$ for all possible $\boldsymbol{l} \neq \boldsymbol{l}^i$. It should be noted that it is also possible to provide an alternative definition of the discriminant function $\mathcal{D}$ (e.g., *maximum mutual information estimation* (MMIE) criterion [Bahl et al., 1986] and *minimum phone error* (MPE) criterion [Povey and Woodland, 2002]). In this chapter, a discriminant function that is similar to the one used in the *minimum classification error* (MCE) training of HMMs [McDermott and Katagiri, 1997] is used because the MCE-type discriminant function yields large-margin criterion when combined with the model training method described in Section 5.4. It should be noted that the discriminant function used in this chapter is defined as a pair-wise discriminant function as contrasted to the Eq. (4.6). i.e., the discriminant function takes an additional parameter $\boldsymbol{l}$ that is treated as an error label sequence, and evaluate the difference in the log-likelihoods between the correct label sequence and the given error label sequence.

5.3.2 Hidden Markov models with log-linear emission pdfs

As in the case of conventional HMMs, it is assumed that the n^{th} vector in an observed sequence $\boldsymbol{x}_n$ depends on the n^{th} HMM state q_n in a state sequence $\boldsymbol{q} = \{q_1, q_2, \cdots, q_n, \cdots\}$, and $\boldsymbol{q}$ depends on a given word sequence $\boldsymbol{l}$. Then, Eq. (5.3) is expressed as follows:

$$\mathcal{D}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) = \log \frac{\sum_{\boldsymbol{q} \in \mathcal{S}(\boldsymbol{l}^i)} \prod_n P(\boldsymbol{x}_n^i | \boldsymbol{\lambda}_{q_n}) P(\boldsymbol{q} | \boldsymbol{l}^i, \boldsymbol{X}^i)}{\sum_{\boldsymbol{q} \in \mathcal{S}(\boldsymbol{l})} \prod_n P(\boldsymbol{x}_n^i | \boldsymbol{\lambda}_{q_n}) P(\boldsymbol{q} | \boldsymbol{l}, \boldsymbol{X}^i)} + \log \frac{P(\boldsymbol{l}^i)}{P(\boldsymbol{l})}. \tag{5.4}$$

Most applications of HMMs approximate the sum of probabilities over every possible state sequence by a probability calculated from a single Viterbi (maximum likelihood) path. Therefore, the following Viterbi discriminant function $\tilde{\mathcal{D}}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda)$ is used instead of Eq. (5.4):

$$\tilde{\mathcal{D}}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) \stackrel{\text{def}}{=} \sum_n \log \frac{P(\boldsymbol{x}_n^i | \boldsymbol{\lambda}_{\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l}^i)})}{P(\boldsymbol{x}_n^i | \boldsymbol{\lambda}_{\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l})})} + \log \frac{P(\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)) P(\boldsymbol{l}^i)}{P(\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})) P(\boldsymbol{l})}, \tag{5.5}$$

where $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ denotes a Viterbi path for the correct word sequence $\boldsymbol{l}^i$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$ denotes a Viterbi path for an incorrect word sequence $\boldsymbol{l}$. Further, $\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l})$ are n^{th} elements in $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$, respectively. $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$ are expressed as follows:

$$\begin{aligned} \hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i) =& \underset{\boldsymbol{q} \in \mathcal{S}(\boldsymbol{l}^i)}{\text{argmax}} P(\boldsymbol{q} | \boldsymbol{X}^i, \boldsymbol{l}^i, \Lambda), \\ \hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}) =& \underset{\boldsymbol{q} \in \mathcal{S}(\boldsymbol{l})}{\text{argmax}} P(\boldsymbol{q} | \boldsymbol{X}^i, \boldsymbol{l}, \Lambda). \end{aligned} \tag{5.6}$$

It should be noted that the Viterbi paths depend on Λ.

As an emission pdf, a log-linear model in an RKHS is considered as a model for a vector $\boldsymbol{x}_n^i$ in a sequence, as follows:

$$\begin{aligned} P(\boldsymbol{x}_n^i | \boldsymbol{\lambda}_s) =& \frac{1}{Z_{\boldsymbol{\phi}}(\boldsymbol{\lambda}_s)} \exp \left\{ \boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x}_n^i) \right\}, \\ Z_{\boldsymbol{\phi}}(\boldsymbol{\lambda}_s) =& \int_x \exp \left\{ \boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x}) \right\} dx. \end{aligned} \tag{5.7}$$

Here, $\boldsymbol{\lambda}_s$ is a weight vector in a log-linear model; $\boldsymbol{\phi}$, a feature warping function (as described in Section 5.2); and $Z_{\boldsymbol{\phi}}$, a partition function obtained by marginalizing out a vector $x \in \mathbb{R}^D$. The likelihood evaluation form of the proposed HMMs is very similar to that of HCRFs [Gunawardana et al., 2005, Reiter et al., 2007]. It should be noted that although kernel machines based on HCRFs are not realized in [Gunawardana et al., 2005, Reiter et al., 2007], the proposed extensions can also be applied to HCRFs. This thesis focused on HMM-based kernel machines.

In general cases, the integral in $Z_{\boldsymbol{\phi}}$ is intractable. Here, the calculation of $Z_{\boldsymbol{\phi}}$ is omitted and assumed to be constant, as in [Sha and Saul, 2007]. By substituting Eq. (5.7) into Eq. (5.5) and by omitting the $Z_{\boldsymbol{\phi}}$, the discriminant function $\tilde{\mathcal{D}}$ is obtained as follows:

$$\tilde{\mathcal{D}}(X^i, \boldsymbol{l}; \Lambda) = \sum_t \left(\boldsymbol{\lambda}_{\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l}^i)} - \boldsymbol{\lambda}_{\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l})} \right)^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x}_n^i) + \log \frac{P(\hat{\boldsymbol{q}}_n(\boldsymbol{X}^i, \boldsymbol{l}^i)) P(\boldsymbol{l}^i)}{P(\hat{\boldsymbol{q}}_n(\boldsymbol{X}^i, \boldsymbol{l})) P(\boldsymbol{l})}. \tag{5.8}$$

Because of the omission of the normalization term $Z_{\boldsymbol{\phi}}$ in Eq. (5.8), the non-normalized log-likelihood $\boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x})$ is used to compute the emission probability at the state s in our methods. Therefore, hereinafter, this quantity (non-normalized log-likelihood) is termed as "score." While conventional methods use GMMs to model $P(\boldsymbol{X}_n^i | s, \Lambda)$, the proposed method uses the simple pseudo-probabilistic distribution $\exp(\boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x}))$ to apply kernel methods.

Although the Viterbi approximated discriminant function is used in order to simplicity and computational efficiency, the approximation strategy used in Chapter 4 is also suitable for the method described in this chapter.

5.4 MRED Training

In this section, an estimation method for parameters of emission pdfs $\boldsymbol{\lambda}_s$ used in the discriminant function (Eq. (5.8)) is described.

Although several frameworks have been identified for estimating the parameters, a training method used in this thesis is derived from the minimum relative entropy discrimination (MRED) framework. As a result of using the MRED framework, several extensions for MRED can be used in future works. For example, multistream speech recognition [Janin et al., 1999] can also be integrated into this framework by employing dynamic kernel combination methods for MRED [Lewis, 2008].

The remainder of this section is organized as follows. First, in Section 5.4.1, the formulation of MRED, a general solution of posterior pdf, and a general form of the objective function are presented and described in terms of this chapter. Then, in Section 5.4.2, an analytical posterior pdf and an analytical objective function are derived by introducing conjugate prior pdfs. Finally, in Section 5.4.3, a method for avoiding the explicit representation of the feature warping function $\boldsymbol{\phi}(\boldsymbol{x})$ by plugging in a kernel function $K(\boldsymbol{x}, \boldsymbol{y})$ into the objective function and the emission pdfs is described.

5.4.1 MRED framework

In MRED, the training of classifiers is formulated as a convex optimization problem, where MRED treats all variables in convex optimization (both the parameters Λ and the slack variables ξ) as random variables. By representing these random variables as distributions, regularization can be performed by minimizing the Kullback-Leibler divergence (KL divergence) of the prior distribution $P^0(\Lambda, \xi)$ from the posterior distribution $P(\Lambda, \xi)$ under the discriminative constraints. The author emphasizes that MRED can estimate a model parameter even if the model is not a probabilistic model. Therefore, the omission of the normalization term $Z_{\boldsymbol{\phi}}$ in the discriminant function $\tilde{\mathcal{D}}$ (Eq. (5.8)) is not crucial in the MRED training process.

The primary problem of this optimization is expressed as follows:

$$\begin{aligned} &\underset{P(\Lambda,\xi)}{\text{minimize}}\ \mathrm{KL}[P(\Lambda, \xi)||P^0(\Lambda, \xi)], \\ &\text{subject to } \left\langle \tilde{\mathcal{D}}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) - \xi_{\boldsymbol{l}}^i \right\rangle_{P(\Lambda,\xi)} \geq 0, \quad \forall i, \forall \boldsymbol{l} \neq \boldsymbol{l}^i. \end{aligned} \tag{5.9}$$

Here, $\langle f(x) \rangle_{g(x)}$ is the expectation of $f(x)$ over the distribution $g(x)$, that is, $\langle f(x) \rangle_{g(x)} \stackrel{\text{def}}{=} \int_x g(x) f(x) dx$. $\mathrm{KL}[f(x)||g(x)]$ is the KL divergence of $g(x)$ from $f(x)$. $\xi = \{\xi_{\boldsymbol{l}}^i | \forall i, \forall \boldsymbol{l} \neq \boldsymbol{l}^i\}$ is a set of slack variables. Each slack variable corresponds to each constraint (i.e., each i and $\boldsymbol{l} \neq \boldsymbol{l}^i$) in the optimization. By decreasing the slack variable $\xi_{\boldsymbol{l}}^i$, the area of the feasible region of the constraint can be increased. However, in general, decrease in slack variables is penalized by introducing a prior pdf $P^0(\xi_{\boldsymbol{l}}^i)$ that favors larger slack variables.

By considering the Lagrange functional of the above optimization problem and from the Karush-Kuhn-Tucker conditions (KKT conditions), the following posterior distribution is obtained by using the variational method:

$$P(\Lambda, \xi) \propto P^0(\Lambda, \xi) \exp\left[\sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} \alpha_{\boldsymbol{l}}^i \left(\tilde{\mathcal{D}}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) - \xi_{\boldsymbol{l}}^i\right)\right], \quad \alpha_{\boldsymbol{l}}^i \geq 0. \tag{5.10}$$

Here, $\alpha \stackrel{\text{def}}{=} \{\alpha_{\boldsymbol{l}}^i \geq 0 | \forall i, \forall \boldsymbol{l} \neq \boldsymbol{l}^i\}$ is a set of Lagrange multipliers of this optimization problem (Eq. (5.9)). Similar to slack variables, the Lagrange multipliers are also introduced for each constraint in the optimization.

Then, the primary problem (Eq. (5.9)) with the $P(\Lambda, \xi)$ optimization is replaced with the following dual problem with α optimization as follows:

$$\begin{aligned} &\underset{\alpha}{\text{maximize}}\ J(\alpha), \\ &\text{subject to } \alpha_{\boldsymbol{l}}^i \geq 0, \qquad \forall i, \forall \boldsymbol{l} \neq \boldsymbol{l}^i \\ &\quad \text{where} \\ &J(\alpha) = -\log Z(\alpha), \\ &Z(\alpha) = \left\langle \exp \sum_{i,\boldsymbol{l}} \alpha_{\boldsymbol{l}}^i \left(\tilde{\mathcal{D}}(\boldsymbol{X}^i, \boldsymbol{l}; \Lambda) - \xi_{\boldsymbol{l}}^i\right)\right\rangle_{P^0(\Lambda,\xi)}. \end{aligned} \tag{5.11}$$

The detailed derivations of the dual problem are described in Appendix A.

5.4.2 Definitions of prior pdfs and derivations of the closed-form objective function

In this section, the closed-form expression of the posterior pdf $P(\Lambda)$ and the objective function $J(\alpha)$ is obtained by introducing conjugate prior pdfs into Eqs. (5.10) and (5.11), respectively. Here, it is assumed that the prior pdf $P^0(\Lambda, \xi)$ can be decomposed into the product of the prior pdf of the parameter of each HMM state $P^0(\boldsymbol{\lambda}_s)$ and that of the slack variable of each constraint $P^0(\xi_{\boldsymbol{l}}^i)$ as follows:

$$P^0(\Lambda, \xi) \stackrel{\text{def}}{=} \prod_s P^0(\boldsymbol{\lambda}_s) \prod_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} P^0(\xi_{\boldsymbol{l}}^i). \tag{5.12}$$

As in the case of large-margin methods including soft-margin SVMs, regularization is performed by minimizing the L2-norm of weight vectors $||\boldsymbol{\lambda}_s||^2$ and the L1-norm of slack variables $||\xi_{\boldsymbol{l}}^i||^1$. Since KL-divergence is defined as the expectation of the difference between log-likelihoods of two distributions. the functional forms of regularization terms used in this method are identical to the logarithm of prior pdfs. Therefore, these regularization criteria are realized by employing Gaussian distributions as the priors of the parameter $\boldsymbol{\lambda}_s$ and exponential distributions as the priors of slack variables $\xi^i_{\boldsymbol{l}}$. The prior pdfs are expressed as follows:

$$
\begin{aligned}
P^0(\boldsymbol{\lambda}_s) &\overset{\text{def}}{=} \mathcal{N}(\boldsymbol{\lambda}_s|0,\boldsymbol{I}),\\
P^0(\xi^i_{\boldsymbol{l}}) &\overset{\text{def}}{=} \begin{cases} \frac{1}{c^0}\exp\left\{-c^0|\delta(\boldsymbol{l}^i,\boldsymbol{l})-\xi^i_{\boldsymbol{l}}|\right\} & \xi^i_{\boldsymbol{l}} < \delta(\boldsymbol{l}^i,\boldsymbol{l}),\\ 0 & \text{otherwise,}\end{cases}
\end{aligned}
\tag{5.13}
$$

where $\delta(\boldsymbol{l}^i,\boldsymbol{l})$ is the label similarity between $\boldsymbol{l}$ and $\boldsymbol{l}^i$. The determinant of the covariance matrix of $P^0(\boldsymbol{\lambda}_s)$ and the hyper-parameter c^0 correspond to the weight variable in soft-margin SVM, which control the trade-off between empirical error minimization and margin maximization. The prior distribution is simplified by setting the covariance matrix in $P^0(\boldsymbol{\lambda}_s)$ as $\boldsymbol{I}$, without any loss of generality. c^0 is scaled appropriately. These prior settings lead to analytical and explicit solutions of the posterior pdfs ($P(\boldsymbol{\lambda}_s)$ and $P(\xi^i_{\boldsymbol{l}})$), because the prior pdfs given in Eq. (5.13) can be assumed to be conjugate prior pdfs.

In discriminative training methods, it is important to determine which measurement of error should be minimized. For example, MCE [McDermott and Katagiri, 1997] attempts to minimize the sequence-level error that is “0” when all elements in a hypothesis sequence $\boldsymbol{l}$ are correct, and “1” otherwise. Because this measurement is coarse and is difficult to minimize, recent approaches measure the impact of an error hypothesis by introducing a fine error measurement. For example, MPE [Povey and Woodland, 2002] uses the approximated phoneme-level edit distance of label sequences, and LM-HMM [Sha and Saul, 2007] uses Hamming distance (frame-level error measurement) between the Viterbi sequence of the correct label sequence and that of a hypothesis sequence $\boldsymbol{l}$. In the proposed method, several error measurements can be incorporated by designing label similarity function $\delta(\boldsymbol{l}^i,\boldsymbol{l})$ in the prior pdf of slack variables $\xi^i_{\boldsymbol{l}}$ (Eq. (5.13)). Because the optimization attempts to ensure that the value of the discriminant function $\tilde{\mathcal{D}}(.)$ is higher than that of $\xi^i_{\boldsymbol{l}}$ (Eq. (5.9)), setting of $\delta(\boldsymbol{l}^i,\boldsymbol{l})$, which is equivalent to the mode value of the prior pdf $P^0(\xi^i_{\boldsymbol{l}})$, is equivalent to designing the error measurement that need to be minimized. The definition of the label similarity function is provided in the experimental sections.

The following posterior distribution of parameters $P(\Lambda|\alpha)$ is obtained by substituting the prior distributions $P^0(\Lambda,\xi)$ (Eqs. (5.12) and (5.13)) into the posterior pdf (Eq. (5.10)), as follows:

$$\begin{aligned} P(\Lambda|\alpha) =& \prod_t P(\boldsymbol{\lambda}_s|\alpha), \\ P(\boldsymbol{\lambda}_s|\alpha) \propto& \underbrace{\mathcal{N}(\boldsymbol{\lambda}_s|0,I)}_{\text{Prior pdf}} \exp\left[\sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} \alpha_{\boldsymbol{l}}^i \left(\tilde{\mathcal{D}}(\boldsymbol{X}^i;\boldsymbol{l},\Lambda) - \xi_{\boldsymbol{l}}^i\right)\right], \\ \propto& \exp\left\{-\frac{1}{2}||\boldsymbol{\lambda}_s||^2\right\} \exp\underbrace{\left[\sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i}\alpha_{\boldsymbol{l}}^i \sum_n \Psi_s(n;i,\boldsymbol{l})\boldsymbol{\phi}(\boldsymbol{x}_n^i)^{\mathsf{T}} \boldsymbol{\lambda}_s\right]}_{(\hat{\boldsymbol{\lambda}}_s(\alpha))^{\mathsf{T}}\hat{\boldsymbol{\lambda}}_s(\alpha)}, \\ \propto& \underbrace{\mathcal{N}(\boldsymbol{\lambda}_s|\hat{\boldsymbol{\lambda}}_s(\alpha),I)}_{\text{Posterior pdf}}, \end{aligned} \tag{5.14}$$

where

$$\begin{aligned} \hat{\boldsymbol{\lambda}}_s(\alpha) =& \sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} \alpha_{\boldsymbol{l}}^i \sum_n \Psi_s(n;i,\boldsymbol{l})\boldsymbol{\phi}(\boldsymbol{x}_n^i), \\ \Psi_s(n;i,\boldsymbol{l}) =& \mathbb{1}(\hat{q}_n(\boldsymbol{X}^i,\boldsymbol{l}^i),s) - \mathbb{1}(\hat{q}_n(\boldsymbol{X}^i,\boldsymbol{l}),s). \end{aligned} \tag{5.15}$$

Here, $\mathbb{1}(x,y)$ is an indicator function that returns 1 when $x = y$ and 0 otherwise, $\Psi_s(n;i,\boldsymbol{l})$ denotes the difference between the occupation probability of the n^{th} frame in the i^{th} feature sequence of the correct Viterbi path $\hat{q}_n(\boldsymbol{X}^i,\boldsymbol{l}^i)$ and that of the incorrect Viterbi path $\hat{q}_n(\boldsymbol{X}^i,\boldsymbol{l})$.

Then, the objective function $J(\alpha)$ in Eq. (5.11) is focused. By solving the integral in the objective function with given priors (Eq. (5.13)), a closed-form expression for $J(\alpha)$ is analytically obtained as a sum of parameter terms $J_{\boldsymbol{\lambda}_s}$, slack variable terms $J_{\xi_{\boldsymbol{l}}^i}$, and hidden variable terms $J_{q_{\boldsymbol{l}}^i}$, as follows:

$$J(\alpha) = \sum_s J_s^{\mathtt{EMIS}}(\alpha) + \sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i}\left(J_{i,\boldsymbol{l}}^{\mathtt{SLACK}}(\alpha) + J_{i,\boldsymbol{l}}^{\mathtt{SHIFT}}(\alpha)\right). \tag{5.16}$$

Here, the emission parameter term can be written as follows:

$$J_s^{\mathtt{EMIS}}(\alpha) = -\,||\hat{\boldsymbol{\lambda}}_s(\alpha)||^2. \tag{5.17}$$

The term $J_s^{\mathtt{EMIS}}$ involves the L2-regularization criterion of the parameter vector $\boldsymbol{\lambda}_s$.

The other terms represent the loss function used in MRED that causes an increase in the Lagrange multipliers $\alpha_{\boldsymbol{l}}^i$ such that the discriminative constraints are satisfied, as follows:

$$\begin{aligned} J_{i,\boldsymbol{l}}^{\mathtt{SLACK}}(\alpha) =& \delta(\boldsymbol{l},\boldsymbol{l}^i)\alpha_{\boldsymbol{l}}^i + \log(c^0 - \alpha_{\boldsymbol{l}}^i), \\ J_{i,\boldsymbol{l}}^{\mathtt{SHIFT}}(\alpha) =& -\,\alpha_{\boldsymbol{l}}^i\left(\log\frac{P(\hat{q}^i)P(\boldsymbol{l}^i)}{P(\hat{q}_{\boldsymbol{l}}^i)P(\boldsymbol{l})}\right). \end{aligned} \tag{5.18}$$

Thus, the α optimization can be solved by maximizing Eqs. (5.17) and (5.18). Further, the optimal posterior pdf can be obtained as $P(\Lambda|\hat{\alpha})$, where $\hat{\alpha} = \mathrm{argmax}_{\alpha} J(\alpha)$.

5.4.3 Kernel-based representations of the objective function and the posterior pdfs

As a result of deriving the dual problem, the parameter term of the objective function ($J_{\boldsymbol{\lambda}_s}(\alpha)$ in Eq. (5.17)) can be rewritten as the weighted sum of the inner product between $\boldsymbol{\phi}(\boldsymbol{x})$'s. As discussed in Section 5.2, since the objective function can be expressed by using the inner product of warped features $\boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{y})$, it is not necessary to explicitly represent the warped features $\boldsymbol{\phi}(\boldsymbol{x})$. The parameter term of the objective function can be rewritten by using the kernel function $K(\boldsymbol{x},\boldsymbol{y}) \overset{\mathrm{def}}{=} \boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{y})$ as follows:

$$J_s^{\mathtt{EMIS}}(\alpha) = -\sum_{i',\boldsymbol{l}'\neq\boldsymbol{l}^{i'}} \sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^{i}} \sum_{n,n'} \alpha_{\boldsymbol{l}}^{i}\alpha_{\boldsymbol{l}'}^{i'} \Psi_s(n;i,\boldsymbol{l})\Psi_s(n';i',\boldsymbol{l}')K(\boldsymbol{x}_n^i,\boldsymbol{x}_{n'}^{i'}). \tag{5.19}$$

When this representation of the parameter term $J_s^{\mathtt{EMIS}}(\alpha)$ is used, it is found that the explicit representation of $\boldsymbol{\phi}$ is removed from all the terms in the objective function (Eqs. (5.17) and (5.18)), and the kernel-based representation can be used to compute the objective function. The practical solver for α optimization is described in Appendix C.

In the evaluation phase (including Viterbi path computation), the score of an unknown input vector $\boldsymbol{x}$ can be evaluated by marginalizing out parameter $\boldsymbol{\lambda}_s$ from the posterior pdf $P(\boldsymbol{\lambda}_s|\hat{\alpha})$ as follows:

$$\langle \log P(\boldsymbol{x}|\boldsymbol{\lambda}_s)\rangle_{P(\boldsymbol{\lambda}_s|\alpha)} = \left\langle \boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{\lambda}_s \right\rangle_{\mathcal{N}(\boldsymbol{\lambda}_s|\hat{\boldsymbol{\lambda}}_s(\hat{\alpha}))} = \boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}}\hat{\boldsymbol{\lambda}}_s(\hat{\alpha}). \tag{5.20}$$

As in the case of the objective function, the kernel-based representation of the score can be obtained by substituting Eq. (5.20) into Eq. (5.15) as follows:

$$\begin{aligned} \langle \log P(\boldsymbol{x}|\boldsymbol{\lambda}_s)\rangle_{P(\boldsymbol{\lambda}_s|\alpha)} =& \boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}} \sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} \alpha_{\boldsymbol{l}}^i \sum_t \Psi_s(n;i,\boldsymbol{l})\boldsymbol{\phi}(\boldsymbol{x}_n^i), \\ =& \sum_{i,\boldsymbol{l}\neq\boldsymbol{l}^i} \alpha_{\boldsymbol{l}}^i \sum_t \Psi_s(n;i,\boldsymbol{l})K(\boldsymbol{x}_n^i,\boldsymbol{x}). \end{aligned} \tag{5.21}$$

Similar to the objective function, the explicit representation of $\boldsymbol{\phi}$ is not necessary in the score evaluation procedure.

Thus, the kernel machines are obtained by handling RKHS via kernel function K. The author termed the models specified by $(\hat{\alpha},\Psi,K)$ as "hidden Markov kernel machines (HMKMs)." As described in the previous section, although HMKM is a kernel machine, the model formulation of HMKM can be treated as that of standard HMMs. Therefore, the scheme of proposed method is similar to that of conventional HMMs trained by discriminative training methods.

5.5 Phoneme classification experiments

In order to evaluate the performance of the proposed method as a sequential classifier, isolated phoneme classification experiments are performed to compare the proposed method with the conventional HMMs that use GMMs as emission pdfs (continuous density HMMs; CD-HMMs).

The objective of the experiments in this section is to evaluate the exact performance of the proposed method, and therefore, approximation techniques of kernel machines are not applied. Since the training session of kernel machines requires enormous computational resources, it is unrealistic to evaluate the exact performance of the proposed method using a large-scale dataset, as discussed in Section 5.1. Therefore, the amount of training datasets used in the experiments is restricted.

5.5.1 Experimental setup

Our method is compared with conventional GMM-based CD-HMMs using two training methods, i.e., maximum likelihood estimation (MLE) and maximum mutual information estimation (MMIE) [Woodland, 2002]. The extended Baum-Welch (EBW) algorithm is used to implement the optimization of MMIE. Although the MLE is the most widely used estimation method for CD-HMMs, MLE procedures are not designed for minimizing classification error. Therefore, our method is also compared with the most widely accepted discriminative training method MMIE [*1].

In these experiments, training datasets of 3 sizes (*small*, *medium* and *large*) and 1 test dataset are prepared for isolated phoneme classification experiments by segmenting the TIMIT dataset according to the label information. There is no overlap between the speakers of the test dataset and those of the training dataset. Table 5.1 summarizes the details of the datasets. All acoustical models in these experiments were constructed as gender-independent models. All feature vectors in the training and test data were whitened by using statistics (covariance matrix and average vector) obtained from the training dataset; the whitening operation is commonly used for the training of discriminative models. In the experiments, both the conventional CD-HMMs and the proposed method have left-to-right 3 states for each 39 phoneme categories defined in [Lee and Hon, 1989]. Configurations for acoustical analysis are summarized in Table 5.2.

[*1] It is reported that the performance of MMIE is similar to that of other discriminative training methods, such as MCE [Schlüter et al., 2001].

Table. 5.1 Dataset description

# categories	39 (defined in [Lee and Hon, 1989])		
Training set	*Small*	*Medium*	*Large*
# segments	3,089	9,275	26,208
# frames	25,390	77,463	219,675
Test set			
# segments	4,243		
# frames	36,790		

Table. 5.2 Acoustical analysis configuration; ∇ denotes time-domain derivative of feature sequence

Sampling rate	16 kHz
Quantization	16 bits
Feature vector	MFCC (12 dims.), Energy, ∇ MFCC, ∇ Energy, $\nabla\nabla$ MFCC, $\nabla\nabla$ Energy. (Total: 39 dims.)
Window len./ shift	25 ms / 10 ms

The following Gaussian kernel was used in the experiments:

$$K(\boldsymbol{x}, \boldsymbol{y}) = \exp\{-\gamma ||\boldsymbol{x} - \boldsymbol{y}||^2\}, \tag{5.22}$$

where γ denotes a hyper-parameter. The Gaussian kernel is widely used in kernel machines because the number of dimensions of $\boldsymbol{\phi}(\boldsymbol{x})$, which satisfies $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{y})$, is infinite when Gaussian kernels are used as K [Schölkopf and Smola, 2002].

Hamming distance between the Viterbi sequences computed from the given word sequences is defined and used as a label similarity $\delta(\boldsymbol{l}^i, \boldsymbol{l})$ in Eq. (5.13), as follows:

$$\delta(\boldsymbol{l}^i, \boldsymbol{l}) = \sum_{n=1}^{N(\boldsymbol{X}^i)} \left(1 - \mathbb{1}\left(\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l}^i), \hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l})\right)\right). \tag{5.23}$$

The Hamming-distance-based measurement of label similarity is widely used in the discriminative training methods (e.g., LM-HMMs [Sha and Saul, 2007] use this measurement); therefore, this label similarity function is used in these experiments. The Hamming distance between 2 phonemes is identical to the number of frames in the sequence ($\delta(\boldsymbol{l}^i, \boldsymbol{l}) = N(\boldsymbol{X}^i)$) because the experiments in this section are isolated phoneme classification experiments (i.e., $\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{q}_n(\boldsymbol{X}^i, \boldsymbol{l})$ are always different for all possible $\boldsymbol{l} \neq \boldsymbol{l}^i$ and n). The hyper-parameter c was set to 5 empirically, and the hyper-parameter γ was varied to examine the behavior of the proposed method.

5.5.2 Discussions on classification performance

Figures 5.2, 5.3, and 5.4 show the comparisons between the phoneme classification error rate of the proposed model and the conventional CD-HMMs trained by *small*, *medium*, and *large* datasets, respectively. The numbers of mixture components in the conventional CD-HMMs are varied, as shown in these figures.

From the experimental results, it is confirmed that our kernel-based models steadily reduce the classification errors. In comparison with CD-HMMs trained by the standard MLE procedure, the proposed kernel machines (HMKMs) reduced the errors by 5.6%, 6.1%, and 10.3% relatively over *small*, *medium*, and *large* datasets, respectively, under the best condi-

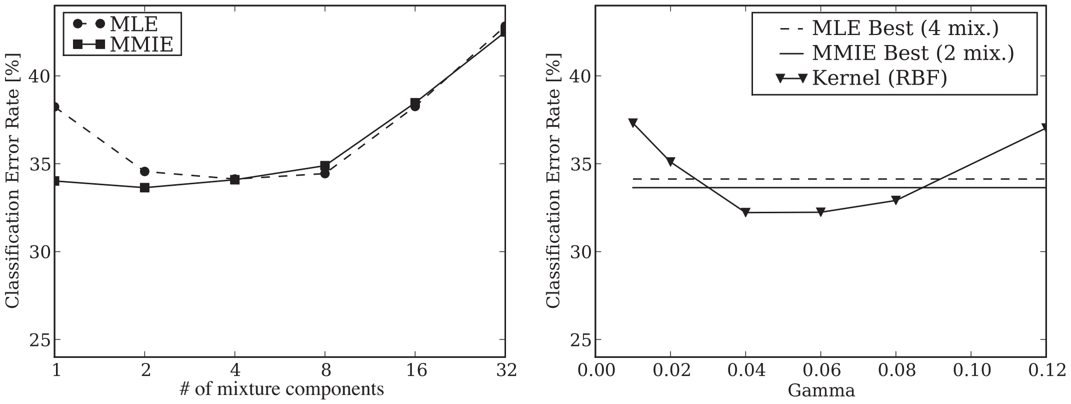


Figure 5.2 Left: Classification error rates of CD-HMMs trained by maximum likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: Classification error rates of hidden Markov kernel machines and CD-HMMs. (*small* dataset)

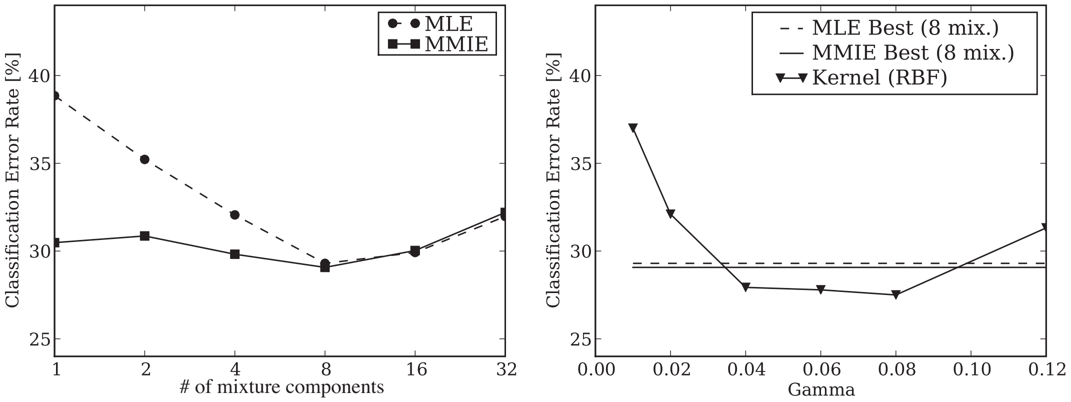


Figure 5.3 Left: Classification error rates of CD-HMMs trained by maximum likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: Classification error rates of hidden Markov kernel machines and CD-HMMs. (*medium* dataset)

tion of each method. In comparison with CD-HMMs with a discriminative training procedure (MMIE), HMKMs reduced the errors by 4.2%, 5.4%, and 5.2% relatively over *small*, *medium*, and *large* datasets, respectively, under the best condition of each method. Therefore, it was concluded that the proposed method achieved improvements in terms of reducing the errors in comparison with conventional CD-HMMs, with the best setting of the number of mixture components for all training datasets.

From Fig. 5.2 (*small* dataset), it is confirmed that the performances of CD-HMMs are saturated by increasing the number of mixture components. In particular, it is found that the performance of the discriminative training method (MMIE) degraded for the models with a large number of mixture components. It is considered that these results are attributed to overfitting problems. However, the proposed method achieved lower error rates even under such conditions. It is considered that this advantage results from the L2-regularization introduced to $\boldsymbol{\lambda}_s$. As in the case of SVMs, the L2-regularization introduced by Gaussian prior (Eq. (5.13)) yields large-margin classifiers that have advantages in generalization ability.

As shown in Fig. 5.4 (*large* dataset), although the overfitting problems might be avoided due to sufficient amounts of data, the relative advantages of the proposed method are confirmed. It is considered that this relative advantage probably results from the prevention of problems arising from local optima. Because our method can prevent the risk of local optima by avoiding mixture models, the problems arising from local optima might be avoided as compared to those occurring in conventional CD-HMMs with a large number of mixture components.

Further, it is observed that setting of the hyper-parameter γ was not so sensitive to classification performance in the proposed method as compared to the setting of the number of mixture components in the GMM methods. For example, in the case of conventional CD-

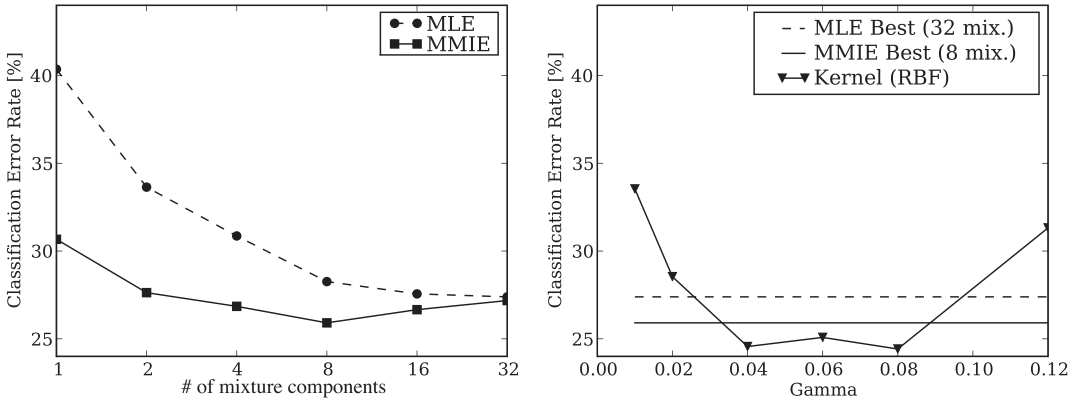


Figure 5.4 Left: Classification error rates of CD-HMMs trained by maximum likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: Classification error rates of hidden Markov kernel machines and CD-HMMs. (*large* dataset)

HMMs, it is observed that the number of mixture components that achieved the best performance in the evaluation of the *small* dataset yielded poor performance in the evaluation of the *large* dataset. On the other hand, the hyper-parameter γ that achieved the best performance in the evaluation of the *small* dataset ($\gamma = 0.04$) also caused performance improvements as compared with CD-HMMs in the evaluation of the *large* dataset. In the experiments, improvements were confirmed in the range $0.4 \leq \gamma \leq 0.8$, even when the amount of training datasets was varied. This property is important for a practical situation since the tuning of γ is not necessary for new datasets, unlike the number of mixture components used in conventional CD-HMMs.

Therefore, it is confirmed that the problems associated with CD-HMMs with GMM-type emission pdfs, i.e., overfitting and local optima, are avoided in HMKMs. Further, it is confirmed that HMKMs with Gaussian kernel offer an advantage in terms of tuning parameters.

5.5.3 Discussions on sparseness

Here, the number of non-zero Lagrange multipliers obtained in the above experimental results are examined. Conventionally, this number is used to evaluate the generalization ability of SVMs. As shown in Eq. (5.14), the sequence $\boldsymbol{X}^i$, corresponding to $\alpha^i_{\boldsymbol{l}} = 0$, does not disturb the estimated posterior pdf $P(\Lambda|\alpha)$ even if it is removed from the training set. In addition, because the zero Lagrange multiplier ($\alpha^i_{\boldsymbol{l}} = 0$) indicates that the inequality constraint attributed to i^{th} training sequence and an incorrect label $\boldsymbol{l}$ is satisfied, the sequence is certainly not misclassified into the incorrect label $\boldsymbol{l}$ by using the estimated posterior pdf $P(\Lambda)$ when $\alpha^i_{\boldsymbol{l}} = 0$. These two properties of Lagrange multipliers indicate that $\boldsymbol{X}^i$ with $\alpha^i_{\boldsymbol{l}} = 0$ is not misclassified into the incorrect label $\boldsymbol{l}$ by using a posterior pdf estimated from the remaining training data. Therefore, a decrease in the number of non-zero $\alpha^i_{\boldsymbol{l}}$ leads to a better performance in leave-one-out cross-validation (LOO-CV), which is commonly used to estimate the generalization performance of SVMs.

In the isolated phoneme classification experiments described in this section, the number of the Lagrange multipliers M corresponds to the product of the number of sequences in the training dataset N and the number of error hypothesis, i.e., $M \overset{\mathrm{def}}{=} |\{(i, \boldsymbol{l})|i \in [1, N], \boldsymbol{l} \neq \boldsymbol{l}^i\}|$. Table 5.3 lists the number of non-zero multipliers $M_+ \overset{\mathrm{def}}{=} |\{(i, \boldsymbol{l})|\alpha^i_{\boldsymbol{l}} \neq 0, i \in [1, N], \boldsymbol{l} \neq \boldsymbol{l}^i\}|$ and the ratio of M_+ to M. From the table, it is confirmed that the proposed method also leads to sparse solutions. The ratios of non-zero multipliers in the experiments were less than or around 10%, as shown in Table 5.3 (8.4%, 10.1%, and 3.8%, respectively), and therefore, the proposed method should achieve good generalization ability.

Further, a sparse solution is also important for reducing the computational complexity. As mentioned in Section 5.2, loop computation over the training data is essential in kernel-based methods. However, if $\alpha^i_{\boldsymbol{l}}$ is 0, the computation due to vectors $\boldsymbol{x}^i_n$, which are related to $\alpha^i_{\boldsymbol{l}}$, can

Table. 5.3 The number of non-zero Lagrange multipliers M^+ in estimated models that achieved the best performance on each dataset, and the ratio of M^+ to the number of Lagrange multipliers M

Dataset	*Small*	*Medium*	*Large*
The best setting of γ	0.04	0.08	0.08
M^+	9843	35771	37518
M^+/M	8.4%	10.1%	3.8%

be omitted. Thus, the computational cost required for evaluating HMKMs can be reduced to M^+/M. Although training and evaluation still require a considerably high computational cost, the proposed method is effective in comparison with kernel-based methods, which yield dense solutions.

5.6 Conclusion

In this chapter, a method for sequential pattern classification derived from kernel methods was proposed; this method is called the hidden Markov kernel machine (HMKM). In the proposed method, vectors in the input sequences are warped to a high-dimensional feature space (reproducing kernel Hilbert space; RKHS) defined by a kernel function and then modeled by hidden Markov models (HMMs) with log-linear emission probability distribution functions (pdfs). Nonlinear classification is achieved without using mixture models by using emission pdfs in RKHS.

The efficiency of the proposed method is confirmed by isolated phoneme classification experiments. The experimental results show that the proposed method outperforms conventional hidden Markov models that use Gaussian mixture models as emission pdfs.

In future, the author intends to reduce the computational costs of training and evaluation by using approximation techniques, aiming for acceleration of kernel-based methods developed in the machine learning community [Kashima et al., 2009, Freitas et al., 2006]. Then, the author also intends to apply our method to large-scale problems, e.g., large vocabulary continuous speech recognition.

Chapter 6

Conclusions

This thesis proposed three methods based on the regularized discrimination of high-dimensional signal representations in order to improve the performance of automatic speech recognizers. This thesis has attempted to indicate the essential significance of the approach by evaluating the proposed three methods.

6.1 Summary of the thesis

In Chapter 1 and Chapter 2, the current situation of state-of-the-art automatic speech recognition research is discussed. First, the current scheme of automatic speech recognition (ASR) and the approach used in this thesis to improve the performance of ASR are presented and described. Then, the conventional methods used for feature extraction, acoustic model estimation, and feature augmentation are described.

In Chapter 3, a method for feature extraction from frequency modulation (FM) of speech signals is presented. The aim of this method is to construct a high-dimensional speech representation that will have complementarity with conventional feature extraction methods. The proposed method is evaluated by carrying out noisy speech recognition experiments and reverberant speech recognition experiments. Further, the properties of FM-based features are discussed.

In Chapter 4, a method for utilizing regularized discrimination based on continuous-density hidden Markov models (CD-HMMs) is proposed and discussed. This method enables regularized discrimination of sequential data associated with sequential labels. The proposed method is evaluated by carrying out continuous phoneme recognition experiments.

In Chapter 5, a kernel-based nonlinear feature transform method is proposed in order to augment the dimensionality of features. This method realizes regularized discrimination in higher-dimensional space. Typically, kernel methods cannot be applied to conventional HMMs because of the modifications required in training methods. However, the proposed method enables application of kernel methods by employing the training method proposed in

Chapter 4. The proposed method is evaluated by conducting isolated phoneme classification experiments.

6.2 Future works

In this section, future works related to the framework proposed in this thesis are discussed.

This thesis addressed the elemental technologies for the construction of automatic speech recognizers on the basis of regularized discrimination of high-dimensional signal representation. However, the performances of the combination technologies were not investigated sufficiently because a number of possible combinations could be identified. In order to construct high-performance speech recognizers, comparative studies are necessary.

Further, in order to apply these methods to large-scale problems, efficient implementations of the corresponding algorithms are necessary, especially in the method proposed in Chapter 5. Although this thesis mentioned some implementation problems, further improvements in computational efficiency may be possible and required.

The author is hopeful that the methods proposed in Chapter 4 and Chapter 5 can be applied into other application areas. For example, gesture recognition, which is conventionally performed by using CD-HMMs, can be enhanced by using the methods presented in this thesis.

6.3 Final remarks

There has been a drastic improvement in ASR technologies as a result of the application of the latest developments from the signal processing research community and the machine learning research community. The main objective of the author is to propose a tolerant classifier that can be used to incorporate the many successful findings arising from developments in these research communities. Specifically, the regularized classifier for high-dimensional features has been introduced to incorporate arbitrary features derived from recent developments in the signal processing research community and to apply efficient optimization techniques derived from recent developments in the machine learning research community.

Although this thesis only mentions the framework and important theories regarding regularized discrimination of high-dimensional features, more instances of this combination technology can be identified. The author considers that interdisciplinary studies of the front-end processing theories and the statistical modeling theories would be advantageous for realizing accurate speech recognition.

The studies presented in this thesis are mainly based on analyses of the engineering aspects of current speech recognition systems with little regard for certain topics in speech-science research, such as phonetics and linguistics. However, the author is hopeful that the contributions in this thesis will inspire the whole of the speech research community. It would be gratifying to the author if this thesis is able to contribute to developments of both the science and engineering aspects of speech research.

Appendix A

Derivations of MRED dual problems

To derive the dual problem, the Lagrange functional of the primary problem (Eq. (5.9)) is introduced as follows:

$$\begin{aligned} L(\alpha,\nu)[f(\Theta,\xi)] =& \left\langle \log f(\Theta,\xi) - \log P^0(\Theta,\xi) \right\rangle_{f(\Theta,\xi)} - \sum_i \alpha^i \left\langle \mathcal{D}(X^i;\Theta) - \xi^i \right\rangle_{f(\Theta,\xi)} \\ & - \nu \left(\int_\Theta \int_\xi f(\Theta,\xi) d\xi d\Theta - 1 \right). \end{aligned} \tag{A.1}$$

Here, α and ν are Lagrange multipliers; $f(\Theta,\xi)$, an argument function that represents a posterior pdf. α^i must remain non-negative.

From the KKT conditions, it is found that the solution of the primary problem $P(\Theta,\xi)$ is located on the saddle point of the Lagrange functional. By applying the variational method to the Lagrange functional, the following relational expression is obtained:

$$\begin{aligned} &\frac{\delta}{\delta f(\Theta,\xi)} L(\alpha,\nu)[f(\Theta,\xi)] \\ =&1 + \log f(\Theta,\xi) - \log P^0(\Theta,\xi) - \sum_i \alpha^i \left(\mathcal{D}(X^i;\Theta) - \xi^i \right) - \nu = 0. \end{aligned} \tag{A.2}$$

Using this equation, the optimal posterior pdf $P(\Theta,\xi)$ is obtained as follows:

$$P(\Theta,\xi) = \exp\{\nu - 1\} P^0(\Theta,\xi) \exp\left\{ \sum_i \alpha^i \left(\mathcal{D}(X^i;\Theta) - \xi^i \right) \right\}. \tag{A.3}$$

Since $P(\Theta,\xi)$ is a pdf and it must be normalized (i.e., $\int_{\Theta,\xi} P(\Theta,\xi) d(\Theta,\xi) = 1$), the multiplier ν depends on α, and it can be rewritten as follows:

$$\begin{aligned}\exp\{\nu - 1\} &= \left\langle \exp\left\{ \sum_i \alpha^i \left(\mathcal{D}(X^i;\Theta) - \xi^i\right) \right\} \right\rangle_{P^0(\Theta,\xi)}, \\ &\overset{\text{def}}{=} \frac{1}{Z(\alpha)}.\end{aligned} \tag{A.4}$$

Because of the convexity of the problem, the saddle point is located at the minimum point obtained by varying $f(\Theta,\xi)$ and the maximum point obtained by varying α. The dual problem is defined by substituting $f(\Theta,\xi)$ in the Lagrange functional L (Eq. (A.1)) by $P(\Theta,\xi)$ (Eq. (A.3)) and by considering the maximization problem with respect to α, as follows:

$$L(\alpha,\nu)[P(\Theta,\xi)] = \underbrace{-\log Z(\alpha)}_{J(\alpha)} + \text{constant}. \tag{A.5}$$

Thus, the dual objective function is obtained as used in Eqs. (4.4) and (5.11).

Appendix B

Solver for Lattice MRED optimization

In order to efficiently perform the MRED optimization, an rprop algorithm is used in the α-optimization [Igel and Hüsken, 2000]. The rprop is suitable for the α-optimization since it exhibits data-parallelism similar to the forward-backward algorithm used in the $\hat{Q}$-optimization. Algorithm 2 shows the detailed pseudo-code of the solver based on rprop.

As described in Section 4.4, the optimization is performed by incrementally adding constraints and corresponding Lagrange multipliers. In Algorithm 2, the variable O indicates the current number of constraints corresponding to each training example.

From Line 4 to Line 7, the forward-backward algorithm is performed for each training example i. Since data-parallelism is ensured in this part, the parallel computation of the forward-backward algorithm is performed by splitting the training dataset.

From Line 8 to Line 12, an initilization of the rprop parameters is performed. Since the α-optimization is a convex optimization, the constants in this part do not effect the result of the optimization. However, these constants are important for computational efficiency. In this thesis, $\alpha^{\mathtt{INIT}}$ is set at 0, and $\nabla^{\mathtt{INIT}}$ is set at $c^0/4$.

From Line 13 to Line 31, the rprop algorithm is performed. Althoguh the rprop algorithm is a gradient-based optimization algorithm, absolute values of derivatives are not used. In the rprop, sign of the derivatives s_o^i are used to determine sign of the updates, and the positive variable ∇_o^i is used to determine the intensity of the update. The ∇_o^i is controled to have a resilience. That is, the step-size is amplified when sign of derivatives s_o^i is the same with that in the previous step $\tilde{s}_o^i$, and the step-size is attenuated when sign of derivatives s_o^i is different with that in the previous step $\tilde{s}_o^i$. The amplification factor ν^+ and the attenuation factor ν^- is typically set at 1.2 and 0.5, respectively.

In the rprop algorithm, the computations of the partial derivatives $\frac{\delta}{\delta\alpha_o^i} J(\alpha, \hat{Q})$, the updates of the α_o^i and ∇_o^i, and the accumulation of delta statistics Δ can be performed by using paral-

Table. B.1 Normalized computational speed as a function of the numbers of computation threads used

# Threads	1	2	4	8	16	32
Forward-Backward	1.0	1.74	3.36	5.81	10.17	13.31
Rprop	1.0	1.70	3.30	5.78	8.51	9.04

lel computation environments. Table B.1 shows the speed-up ratio obtained by increasing the number of computation threads used. From this table, it is confirmed that the rprop can also be efficiently performed by using parallel computation environments. However, degradations are confirmed when a number of threads are used. It is considered that this deficit is due to the bus speed bottleneck.

Algorithm 2 MRED optimization algorithm

```
$\hat{\Theta}^{(0)} \overset{\text{def}}{=} \hat{\Theta}^{\texttt{MLE}}$
$O \leftarrow 1$
loop
    for all $i$ do
        Compute the sufficient statistics $\chi(\boldsymbol{X}^i, A^i; \Theta^{(O-1)}) \leftarrow \text{FwdBkwd}(A^i, \Theta^{(O-1)})$
        Compute the sufficient statistics $\chi(\boldsymbol{X}^i, \tilde{A}^i; \Theta^{(O-1)}) \leftarrow \text{FwdBkwd}(\tilde{A}^i, \Theta^{(O-1)})$
    end for
    for all $i, o \leq O$ do
        $\alpha_o^i \leftarrow \alpha^{\texttt{INIT}}$
        $\nabla_o^i \leftarrow \nabla^{\texttt{INIT}}$
        $\tilde{s}_o^i \leftarrow 0$
    end for
    loop
        Initialize delta statistics $\Delta$
        for all $i, o \leq O$ do
            $s_o^i \leftarrow \text{sign}\left\{\frac{\delta}{\delta \alpha_o^i} J(\alpha, \hat{Q})\right\}$
            if $s_o^i = 0$ or $\tilde{s}_o^i = 0$ then
                /* do nothing */
            else if $s_o^i = \tilde{s}_o^i$ then
                $\nabla_o^i \leftarrow \max\left\{\nabla^{\texttt{MIN}}, \min\left\{\nabla^{\texttt{MAX}}, \nu^+ \cdot \nabla_o^i\right\}\right\}$
            else
                $\nabla_o^i \leftarrow \max\left\{\nabla^{\texttt{MIN}}, \min\left\{\nabla^{\texttt{MAX}}, \nu^- \cdot \nabla_o^i\right\}\right\}$
            end if
            $\alpha_o^i \leftarrow \max\left\{0.0, \min\left\{\alpha_o^i + s_o^i \cdot \nabla_o^i, c^0\right\}\right\}$
            $\tilde{s}_o^i \leftarrow s_o^i$
            $\Delta \leftarrow \Delta + \alpha_o^i(\chi(\boldsymbol{X}^i, \tilde{A}^i; \Theta^{(o)}) - \chi(\boldsymbol{X}^i, \tilde{A}^i; \Theta^{(o)}))$
        end for
    end loop
    Determine $\Theta^{(O)}$ by using the prior pdf parameters and the delta statistics $\Delta$.
end loop
```

Appendix C

Solver for HMKM optimization

It is inefficient to carry out optimization by a naive implementation for convex programming solvers because the number of possible $\boldsymbol{l}$ is large. In order to handle a large number of possible $\boldsymbol{l}$, a method used in structured SVMs [Tsochantaridis et al., 2005] is used. Because the Viterbi alignment computations are required in the proposed method, some modifications to the structured SVM are required. The modified algorithm is described in Algorithm 3.

In Algorithm 3, the set $\mathcal{C}_i$ stores the working sets of label sequences (called "cutting planes" in [Tsochantaridis et al., 2005]) associated with the i^{th} training data. The label sequence $\hat{\boldsymbol{l}}$ with the smallest expected margin $M(\hat{\boldsymbol{l}};\Lambda)$ is selected and incrementally added to the set $\mathcal{C}_i$ if the expected margin $M(\hat{\boldsymbol{l}};\Lambda)$ (defined in Line 2) is smaller than the smallest expected margin among the label sequences in the current working set $\mathcal{C}_i$.

The expected margin $M(\boldsymbol{l};\Lambda)$ for a given label sequence $\boldsymbol{l}$ is defined as the difference between the current discriminant function $\tilde{\mathcal{D}}(X^i,\boldsymbol{l};\Lambda)$ and the label similarity function $\delta(\boldsymbol{l},\boldsymbol{l}^i) = \mathrm{argmax}_{\xi^i} P^0(\xi^i)$, as follows:

$$\begin{aligned} M(\boldsymbol{l};\Lambda) \overset{\mathrm{def}}{=} & \tilde{\mathcal{D}}(X^i,\boldsymbol{l};\Lambda) - \underset{\xi^i}{\mathrm{argmax}} P^0(\xi^i) \\ = & \tilde{\mathcal{D}}(X^i,\boldsymbol{l};\Lambda) - \delta(\boldsymbol{l},\boldsymbol{l}^i). \end{aligned} \tag{C.1}$$

Similar to the axis-parallel optimization described in [Jebara, 2001], the proposed algorithm only considers updating a single Lagrange multiplier α^i at each iteration (Line 18), where i and $\boldsymbol{l}$ are randomly selected in Lines 6 and 17, respectively. Because the maximization of the objective function in the direction of a single multiplier can be solved analytically, the optimization is typically very fast in comparison to gradient-based methods. Specifically,

Algorithm 3 Modified cutting plane algorithm

```
$\hat{\Lambda}(\alpha) \overset{\mathrm{def}}{=} \{\hat{\boldsymbol{\lambda}}_1(\alpha), \cdots, \hat{\boldsymbol{\lambda}}_s(\alpha), \cdots, \hat{\boldsymbol{\lambda}}_s(\alpha)\}$ (Eq. (5.15))
$M(\boldsymbol{l}; \Lambda) \overset{\mathrm{def}}{=} \tilde{\mathcal{D}}(X^i, \boldsymbol{l}; \Lambda) - \delta(\boldsymbol{l}^i, \boldsymbol{l})$
$\alpha^i \leftarrow 0$ for all $i$ and $\boldsymbol{l} \neq \boldsymbol{l}^i$
$\mathcal{C}_i \leftarrow \phi$ for all $i$
loop
  $i \leftarrow$ choose one training example
  if $\hat{\boldsymbol{\lambda}}_s(\alpha) \neq 0$ for all s then
    $\hat{\boldsymbol{l}} \leftarrow \mathrm{argmin}_{\boldsymbol{l} \neq \boldsymbol{l}^i} M(\boldsymbol{l}; \hat{\Lambda}(\alpha))$
    /* Performed by conventional decoding algorithms. */
  else
    $\hat{\boldsymbol{l}} \leftarrow$ choose one possible incorrect label sequence randomly
  end if
  if $\hat{\boldsymbol{\lambda}}_s(\alpha) = 0 \, \exists s$, or $\left(\min_{\boldsymbol{l} \in \mathcal{C}_i} M(\boldsymbol{l}; \hat{\Lambda}(\alpha))\right) > \min\{0, M(\hat{\boldsymbol{l}}; \hat{\Lambda}(\alpha))\} + \epsilon$ then
    $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup \{\hat{\boldsymbol{l}}\}$
  end if
  compute $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$ $(\forall \boldsymbol{l} \in \mathcal{C}_i)$ by using Viterbi algorithms with current
  parameters $\hat{\Lambda}(\alpha)$
  while $\alpha^i$ converges for all $\boldsymbol{l}$ do
    $\boldsymbol{l} \leftarrow$ choose random $\boldsymbol{l}$ from $\mathcal{C}_i$
    optimize $\alpha^i$ with given $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$
  end while
end loop
```

the update rule can be derived as follows:

$$
\begin{aligned}
\alpha^i \leftarrow & \left\{ \min \left[c^0 - \epsilon, \max \left\{ 0.0, \tfrac{-m(i,\boldsymbol{l}) \pm \sqrt{(m(i,\boldsymbol{l}))^2 - 4l(i,\boldsymbol{l})n(i,\boldsymbol{l})}}{2l(i,\boldsymbol{l})} \right\} \right] \right. \\
l(i,\boldsymbol{l}) =& 2A(i,\boldsymbol{l}) \\
m(i,\boldsymbol{l}) =& -\left(N(\boldsymbol{X}^i) + 2cA(i,\boldsymbol{l}) - 2B(i,\boldsymbol{l}) \right) \\
n(i,\boldsymbol{l}) =& cN(\boldsymbol{X}^i) - 1 - 2B(i,\boldsymbol{l})c \\
A(i,\boldsymbol{l}) =& \sum_{s=1}^{S} \sum_{n=1}^{N(\boldsymbol{X}^i)} \sum_{n'=1}^{N(\boldsymbol{X}^i)} \Psi_s(n; i, \boldsymbol{l}) \Psi_s(n'; i, \boldsymbol{l}) K(\boldsymbol{x}_n^i, \boldsymbol{x}_{n'}^{i'}) \\
B(i,\boldsymbol{l}) =& \sum_{s=1}^{S} \sum_{i' \neq i} \sum_{\boldsymbol{l}' \neq \boldsymbol{l}} \sum_{n=1}^{N(\boldsymbol{X}^i)} \sum_{n'=1}^{N(\boldsymbol{X}^{i'})} \alpha_{\boldsymbol{l}'}^{i'} \Psi_s(n; i, \boldsymbol{l}) \Psi_s(n'; i', \boldsymbol{l}') K(\boldsymbol{x}_n^i, \boldsymbol{x}_{n'}^{i'}).
\end{aligned}
\tag{C.2}
$$

where c^0 is the hyper-parameter. Here, the plus or minus in the equation is chosen so that maximize objective function by evaluating both.

Further, because the hidden state sequences, $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l}^i)$ and $\hat{\boldsymbol{q}}(\boldsymbol{X}^i, \boldsymbol{l})$ used in the optimization (Line 18) may be obtained as interim results of the decoding process carried out in Line 8, optimization is carried out efficiently by using conventional decoding algorithms.

The working set selection algorithm is similar to the conventional N-best approach [Chen and Soong, 1994, McDermott and Katagiri, 1997]. In the proposed method, the competitor $\boldsymbol{l}$, which is considered to be important for optimization, is selected and incrementally added to working set $\mathcal{C}_i$. Thus, it is ensured that the proposed solver converges to the explicit solution by adding all possible $\boldsymbol{l}$ to the working set. It should be noted that optimization over all possible $\boldsymbol{l}$ is not necessary in common cases because most competitors are redundant, and most α^i remain 0.

List of Works

Journal papers

(J1) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “Recognizing Reverberant Speech Based on Amplitude and Frequency Modulation,” *IEICE Transactions on Information and Systems*, vol. E91–D, no. 3, pp. 448–456, (2008).

(J2) Y. Kubo, M. Honda, K. Shirai, T. Komori, N. Seiyama, T. Takagi, “Improved High-Quality MPEG–2/4 Advanced Audio Coding Encoder,” *Acoustical Science and Technology*, vol. 29, no. 6, pp. 362–371, (2008).

International conerences/workshops (Refereed)

(IC1) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “Instantaneous phase analysis using neural networks for automatic speech recognition,” In *Proc. NCSP-2007*, pp. 321-324, (2007).

(IC2) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “A study on temporal features derived by analytic signal,” In *Proc. Interspeech-2007*, pp. 1130-1133, (2007).

(IC3) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “Noisy speech recognition using temporal AM-FM combination,” In *Proc. ICASSP-2008*, pp. 4709-4712, (2008).

(IC4) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “Independent feature selection algorithms for the creation of multistream speech recognizers,” In *Proc. ITRW on Speech Analysis and Processing for Knowledge Discovery*, June 2008.

(IC5) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, “A comparative study on AM and FM features,” In *Proc. Interspeech-2008*, pp. 642–645, (2008).

Domestic conferences/workshops (in Japanese)

(DC1) Y. Kubo, M. Honda, K. Shirai, “Improvement of MPEG–2/4 AAC coder aiming for broadcasting,” In *Proc. Fall Meeting of ASJ 2006*, 1–Q–21, pp. 279–280, (2006).

(DC2) Y. Kubo, M. Honda, K. Shirai, T. Komori, N. Seiyama, T. Takagi, “Improvement of

MPEG–2/4 AAC coder using optimal bit allocation aiming for broadcasting," In *Proc. Spring Meeting of ASJ 2007*, 3–P–13, pp. 623–624, (2007).

(DC3) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, "Long-term instantaneous phase analysis for automatic speech recognition of Japanese spontaneous speech," In *Proc. Spring Meeting of ASJ 2007*, 3–10–9, pp. 121–122, (2007).

(DC4) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, "Speech recognition using narrow-band analytic signal and non-linear discriminant analysis," In *Technical Report of IEICE*, SP–2007–116, pp. 85–90, (2007).

(DC5) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, "A Study on speech recognizer based on temporal AM-FM analysis," In *Technical Report of IEICE*, SP–2007–356, pp. 31–36, (2007).

(DC6) S. Kusano, Y. Kubo, A. Kurematsu, K. Shirai, "Rescoring of Speech Recognition by Use of Language Model with Prosodic Condition," In *Proc. of the IEICE General Conference,* pp. 184, (2008).

(DC7) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, "Multi-stream speech recognizers using independent feature decomposition," In *Proc. Spring Meeting of ASJ 2008*, 2–10–4, pp. 73–74, (2008).

(DC8) S. Kusano, Y. Kubo, A. Kurematsu, K. Shirai, "Rescoring of N-best Recognition Score Using Prosodic Conditional Language Models," In *Proc. Fall Meeting of ASJ 2008*, 2–P–14, pp. 141–142, (2008).

(DC9) Y. Kubo, S. Okawa, A. Kurematsu, K. Shirai, "A Regularized Discriminative Training Method for Continuous Density Hidden Markov Models Based on Minimum Relative Entropy Discriminative Formulation," In *Proc. Spring Meeting of ASJ 2009*, 2–5–16, pp. 85–88, (2009).

(DC10) Y. Kubo, S. Watanabe, A. Nakamura, T. Kobayashi, "A Regularized Discriminative Training Method for Continuous Density Hidden Markov Models Based on Minimum Relative Entropy Discriminative Formulation," In *Proc. Spring Meeting of ASJ 2009*, 2–5–16, pp. 85–88, (2009).

(DC11) Y. Kubo, S. Watanabe, A. Nakamura, E. McDermott, T. Kobayashi, "A Kernel Machine Derived by Minimum Relative Entropy Discrimination For Automatic Speech Recognition," In *IPSJ SIG Technical Report*, vol. 2009–SLP–77, no. 6, (2009).

(DC12) Y. Kubo, S. Watanabe, A. Nakamura, E. McDermott, T. Kobayashi, "Hidden Markov Kernel Machines Derived by Minimum Relative Entropy Discrimination Training of Hidden Markov Models for Automatic Speech Recognition," In *Proc. Fall Meeting of ASJ 2009*, 1–1–4, pp. 11–14, (2009).

(DC13) Y. Kubo, S. Watanabe, A. Nakamura, E. McDermott, T. Kobayashi, "Sequence Classification Using Hidden Markov Kernel Machines and its Application to Phoneme Recognition Task," In *Collection of Preview Slides of the 12*$^{\mathrm{th}}$ *Workshop on*

Information-Based Induction Sciences (IBIS 2009), pp. 92, (2009).

(DC14) Y. Kubo, S. Watanabe, A. Nakamura, T. Kobayashi, “Parallelizable Optimization Methods and Lattice-based Representations for Minimum Relative Entropy Discrimination Training,” In *IPSJ SIG Technical Report*, vol. 2009–SLP–80, (2009).

Acknowledgements

First of all, I would like to express my gratitude to Prof. Katsuhiko Shirai, who was the main supervisor of this thesis, for leading the laboratory in which I joined during my master course and my undergraduate course. Further, I am grateful to all members of my thesis committee: Prof. Yasuo Matsuyama, Prof. Yoshinori Sagisaka, and Prof. Tetsunori Kobayashi. Especially, I again extend warm thanks to Prof. Tetsunori Kobayashi for leading the laboratory in which I joined during my doctoral course.

The research projects described in this thesis were supported by many advises from many collaborators. The research projects were started at Human-Machine Interface Laboratory (Prof. K. Shirai Laboratory) in Waseda University during my master course, sophisticated by the discussions with the collaborators in NTT Communication Science Laboratories during my internship program, and completed at the Perceptual Computing Laboratory (Prof. T. Kobayashi Laboratory) in Waseda University.

I would like to extend thanks to Prof. Akira Kurematsu, who is the leader of the Spoken Dialogue System Seminar in Human-Machine Interface Laboratory, for valuable discussions which are reflected throughout this thesis. Further, I would like to express gratitude to Prof. Mikio Tohyama, who is the leader of the Acoustic Signal Processing Seminar in Human-Machine Interface Laboratory, for valuable advises which make mention to both technical and philosophical matters. My first research projects, which were collaborative projects with NHK Science & Technology Research Laboratories, were in the area of high-quality audio signal encoders/ decoders. I would like to express my gratitude to Prof. Masaaki Honda, who is the leader of the Speech Processing Seminar in Human-Machine Interface Laboratory, for valuable direction in these projects although I was a beginner of engineering research. Furthermore, I would like to extend warm thanks to members of NHK Science & Technology Research Laboratories: Mr. Tomoyasu Komori, Mr. Nobumasa Seiyama (currently at NHK Engineering Service), and Dr. Tohru Takagi for their valuable comments for audio coding research and subjective quality assessments. I was fortunate since such an advantageous training is provided during my initial period as a researcher.

I would like to express my gratitude to Prof. Shigeki Okawa at Chiba Institute of Technology for valuable discussions about MLP/HMM-tandem approach of acoustic modeling. I often feel encouraged by discussions with Prof. Shigeki Okawa.

I would also like to thank Dr. Shinji Watanabe at NTT Communication Science Laboratories for many valuable comments about statistical inference methods. A portion of this thesis is supported by the internship program of the NTT Communication Science Laboratories, and the collaborative projects between Waseda University and NTT Communication Science Laboratories. I am very grateful to members of the NTT Communication Science Laboratories: Dr. Erik McDermott, Dr. Atsushi Nakamura, and Dr. Jun Suzuki, for in-depth discussions and insightful suggestions.

Part of this thesis is proofread by Ms. Shizuka Nakamura and Mr. Yuichi Koshiba. I would like to thank for their assistance. They made it possible that I could concentrate on the contents of this thesis, without concerning grammatical issues, until just before the due date.

While these are not mentioned by this thesis, the collaborative projects with my colleagues always inspire me. I would like to extend gratitude to collaborators: Mr. Kentaro Hikosaka (currently at Hewlett-Packard Japan, Ltd.), Mr. Hirotsugu Tabata (currently at The Kansai Electric Power Co., Inc.), Mr. Shunichi Kusano, Mr. Naoya Sugahara, and Ms. Fuyuka Suzuki. I am grateful to all of my colleagues. Especially, I would like to thank Dr. Toru Taniguchi, Dr. Tetsuji Ogawa, Mr. Takahiro Miyajima, Ms. Yukiko Ipposhi, and Ms. Manami Ogawa, for their assistance in everyday matters.

Finally, I would like to thank to my father, my mother and my sisters. They made every possible assistance so that I could concentrate on the research projects.

Acknowledgements in Japanese

本論文は，著者が早稲田大学基幹理工学研究科情報理工学専攻の白井研究室，および小林研究室で行なった研究をまとめたものになります．本論文をまとめるにあたり，私の学士／修士在学中の指導教官であり，本論文の主査でもある早稲田大学 白井克彦教授に感謝の意を表します．私の博士課程在学中の指導教官であり，本論文の副査でもある小林哲則教授からも技術面のみならず様々な点において懇切な指導，御助言を多数頂きました．御二方の指導がなければ本論文の完成はなかったと考えております．ならびに，副査として本論文についての有益な助言を多数頂いた，松山泰男教授，匂坂芳典教授に深く感謝致します．

本研究は，早稲田大学基幹理工学研究科の白井克彦研究室で始まり，日本電信電話 コミュニケーション科学基礎研究所での実習経験を通し，小林哲則研究室での研究を経て完成されたものであり，非常に多くの人の助言，指導があって成り立っています．

白井克彦研究室 音声対話グループの指導教官である榑松明教授とは研究のスタートアップから博士論文の完成に至るまで，常に議論を行ない，様々なアドバイスをいただきました．深く感謝致します．また，音響分析技術に関しましては，音響信号処理グループの指導教官である東山三樹夫教授から助言を頂きました．教授からはの技術的側面のみではなく様々な面でアドバイスをいただきました．

私の最初の研究テーマはオーディオ信号の高品質符号化に関するものでした．まだ工学の研究について右も左もわからない私に対して，様々なアドバイスをくださった早稲田大学 誉田雅彰教授に深く感謝しております．また，符号化アルゴリズムの技術的課題，聴覚実験の実行の際のノウハウ等，様々なことを教えてくださった日本放送協会 放送技術研究所の小森智康氏，清山信正氏 (現・財団法人 NHK エンジニアリングサービス)，都木徹博士に深く感謝したいと思います．研究生活のスタートアップ時に，綿密な指導を受けられたことが，現在の研究のモチベーションに繋っていると思っています．

高次元特徴の取り扱い，とりわけ相補的な特徴量の取り扱いに関しては，千葉工業大学 大川茂樹教授から多くの助言を頂きました．深く感謝致します．初めての国際会議への投稿も，論文誌への投稿も，教授の激励があればこそ，自信を持って投稿できたと思っています．

音響モデルの統計的推定問題に関しては，日本電信電話株式会社 渡部晋治博士から非常に有益なアドバイスを多数頂きました．また，日本電信電話株式会社の実習生制度，および同社との共同研究において，渡部晋治博士のみならず，エリックマクダーモット博

士，中村篤博士，鈴木潤博士らと綿密な議論ができたことによって本研究の成果は実現できたと考えております．渡部晋治博士には，技術的な側面だけに留まらず，論文の書き方，研究の進めかたなど，様々な点で影響を受けました．あらためて感謝いたします．

中村静氏および小柴侑一氏には，本論文の一部を精読してもらい，目が届かなかった英文法の誤りを指摘してもらいました．彼女らの支えがあればこそ，提出期限直前まで論旨に集中ができたと思っています．

また，研究室の先輩・後輩でもあり，共同研究者でもある彦坂健太郎氏 (現・日本ヒューレットパッカード株式会社)，田畠大嗣氏 (現・関西電力株式会社)，草野俊一氏，菅原尚也氏，鈴木冬華氏との研究は，本論文の内容とは直接関連していませんが，様々な刺激を受けました．

また，共同研究者ではありませんが，私の研究を様々な面でサポートしてくれた，白井研究室の秘書である一法師由紀子氏，小林研究室の秘書である小川真奈美氏，白井研究室の先輩である谷口徹博士，宮島崇浩氏，および小林研究室の先輩である小川哲司博士にも改めて感謝をしたいと考えております．

最後に，今日までの研究生活を精神的に支え，私が研究に専念できる環境を提供してくれた家族に感謝の意を表わします．

Bibliography

[Allauzen et al., 2007] Allauzen, C., Riley, M., Schalkwyk, J., Skit, W., and Mohri, M. (2007). OpenFst: A general and efficient weighted finite-state transducer library – (extended abstract of an invited talk). *Lecture Notes in Computer Science*, 4783:11–23.

[Allen and Li, 2009] Allen, J. B. and Li, F. (2009). Speech perception and cochlear signal processing. *Signal Processing Magazine, IEEE*, 26(4):73–77.

[Atal, 1974] Atal, B. S. (1974). Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America*, 55(6):1304–1312.

[Attias, 2000] Attias, H. (2000). A variational bayesian framework for graphical models. *Advances in Neural Information Processing Systems*, pages 49–52.

[Bahl et al., 1986] Bahl, L. R., Brown, P. F., de Souza, P. V., and Mercer, R. L. (1986). Maximum mutual information estimation of hidden Markov model parameters for speech recognition. In *Proc. ICASSP–86*, pages 49–52.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[Boashash, 1992] Boashash, B. (1992). Estimating and interpreting the instantaneous frequency of a signal. *Proceedings of the IEEE*, 80(4):520–538.

[Boser et al., 1992] Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proc. CoLT–1992*, pages 144–152. ACM New York, NY, USA.

[Chen et al., 2001] Chen, B. Y., Chang, S., and Sivadas, S. (2001). Learning discriminative temporal patterns in speech: Development of novel TRAPS-like classifiers. In *Proc. EUROSPEECH–2001*, pages 429–432.

[Chen et al., 2004a] Chen, B. Y., Zhu, Q., and Morgan, N. (2004a). Learning long-term temporal features in LVCSR using neural networks. In *Proc. ICSLP–2004*.

[Chen et al., 2005] Chen, B. Y., Zhu, Q., and Morgan, N. (2005). Tonotopic multi-layered perceptron: A neural network for learning long-term temporal features for speech recognition. In *Proc. ICASSP–2005*, volume 1, pages 945–948.

[Chen et al., 2004b] Chen, J., Huang, Y., Li, Q., and Paliwal, K. K. (2004b). Recognition of noisy speech using dynamic spectral subband centroids. *IEEE Signal Processing Letters*, 11(2):258–261.

[Chen and Soong, 1994] Chen, J.-K. and Soong, F. (1994). An N-best candidates-based discriminative training for speech recognition applications. *Speech and Audio Processing, IEEE Transactions on*, 2(1):206–216.

[Collobert and Bengio, 2004] Collobert, R. and Bengio, S. (2004). Links between perceptrons, MLPs and SVMs. In *Proc. ICML–2004*. ACM New York, NY, USA.

[Cuturi et al., 2007] Cuturi, M., Vert, J. P., Birkenes, Ø., and Matsui, T. (2007). A kernel for time series based on global alignments. In *Proc. ICASSP–2007*, volume 2, pages II–413–II–416.

[Dimitriadis et al., 2005] Dimitriadis, D., Maragos, P., and Potamianos, A. (2005). Robust AM-FM features for speech recognition. *IEEE Signal Processing Letters*, 12(9):621–624.

[Freitas et al., 2006] Freitas, N. D., Wang, Y., Mahdaviani, M., and Lang, D. (2006). Fast krylov methods for n-body learning. *Advances in Neural Information Processing Systems*, 18:251–258.

[Furui, 1981] Furui, S. (1981). Cepstral analysis technique for automatic speaker verification. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(2):254–272.

[Gajic and Paliwal, 2003] Gajic, B. and Paliwal, K. K. (2003). Robust speech recognition using features based on zero crossings with peak amplitudes. In *Proc. ICASSP–2003*, volume 1, pages I–64–I–67.

[Ganapathiraju et al., 2000] Ganapathiraju, A., Hamaker, J. E., and Picone, J. (2000). Hybrid SVM/HMM architectures for speech recognition. In *Proc. ICSLP–2000*, volume 4, pages 504–507, Beijing, China.

[Ganapathiraju et al., 2004] Ganapathiraju, A., Hamaker, J. E., and Picone, J. (2004). Applications of support vector machines to speech recognition. *Signal Processing, IEEE Transactions on*, 52(8):2348–2355.

[Godfrey et al., 1992] Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: telephone speech corpus for research and development. In *Proc. ICASSP–92*, volume 1, pages 517–520.

[Green, 1976] Green, G. G. R. (1976). *Temporal aspects of audition*. PhD thesis, University of Oxford.

[Gunawardana et al., 2005] Gunawardana, A., Mahajan, M., and Acero, A. (2005). Hidden conditional random fields for phone classification. In *Proc. INTERSPEECH–2005*, pages 1117–1120, Lisbon, Portugal.

[Hermansky, 1990] Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87:1738–1752.

[Hermansky, 1998] Hermansky, H. (1998). Should recognizers have ears? In *Proc. ESCA ETRW on Robust Speech Recognition for Unknown Communication Channels*, pages 1–10.

[Hermansky et al., 2000] Hermansky, H., Ellis, D. P. W., and Sharma, S. (2000). Tandem connectionist feature extraction for conventional HMM systems. In *Proc. ICASSP-2000*, volume 3, pages 1635–1638.

[Hermansky and Morgan, 1994] Hermansky, H. and Morgan, N. (1994). RASTA processing of speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578–589.

[Hermansky and Sharma, 1998] Hermansky, H. and Sharma, S. (1998). TRAPS - classifiers of temporal paterns. In *Proc. ICSLP–98*, Sydney, Australia.

[Houtgast and Steeneken, 1985] Houtgast, T. and Steeneken, H. J. M. (1985). A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria. *Journal of the Acoustical Society of America*, 77(3):1069–1077.

[Huang et al., 2006] Huang, B. Q., Du, C. J., Zhang, Y. B., and Kechadi, M.-T. (2006). A hybrid HMM-SVM method for online handwriting symbol recognition. In *Proc. ISDA–2006*, pages 887–891, 1.

[Igel and Hüsken, 2000] Igel, C. and Hüsken, S. (2000). Improving the rprop learning algorithm. In *Proc. the 2nd International ICSC Symposium on Neural Computation (NC–2000)*, pages 115–121.

[Ikbal et al., 2004] Ikbal, S., Misra, H., Sivadas, S., and Hermansky, H. (2004). Entropy based combination of tandem representations for noise robust asr. In *Proc. ICSLP–2004*, pages 2553–2556.

[Jaakkola et al., 2000] Jaakkola, T. S., Meila, M., and Jebara, T. (2000). Maximum entropy discrimination. *Advances in Neural Information Processing Systems*, 12:470–476.

[Janin et al., 1999] Janin, A., Ellis, D. P. W., and Morgan, N. (1999). Multi-stream speech recognition: Ready for prime time? In *Proc. EUROSPEECH–1999*, pages 591–594.

[Jebara, 2001] Jebara, T. (2001). *Discriminative, Generative and Imitative Learning*. PhD thesis, Columbia University.

[Joder et al., 2008] Joder, C., Essid, S., and Richard, G. (2008). Alignment kernels for audio classification with application to music instrument recognition. In *Proc. EUSIPCO-2008*, Lausanne, Switzerland.

[Juang and Katagiri, 1992] Juang, B. H. and Katagiri, S. (1992). Discriminative learning for minimum error classification. *Signal Processing , IEEE Transactions on*, 40(12):3043–3054.

[Kaiser, 1993] Kaiser, J. F. (1993). Some useful properties of Teager's energy operators. In *Proc. ICASSP-93*, volume 3, pages 149–152.

[Kashima et al., 2009] Kashima, H., Ide, T., Kato, T., and Sugiyama, M. (2009). Recent advances and trends in large-scale kernel methods. *Information and Systems, IEICE Transactions on*, E92-D(7):1338–1353.

[Kim et al., 1999] Kim, D.-S., Lee, S.-Y., and Kil, R. (1999). Auditory processing of speech signals for robust speech recognition in real-world noisy environments. *Speech and Audio Processing, IEEE Transactions on*, 7(1):55–69.

[Lamel et al., 1986] Lamel, L. F., Kassel, R. H., and Seneff, S. (1986). Speech database development: Design and analysis of the acoustic-phonetic corpus. In *Proc. DARPA Speech Recognition Workshop*, pages 100–109.

[Lee and Hon, 1989] Lee, K.-F. and Hon, H.-W. (1989). Speaker-independent phone recognition using hidden Markov models. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(11):1641–1648.

[Lewis, 2008] Lewis, D. P. (2008). *Combining Kernels for Classification.* PhD thesis, Columnbia University.

[Lippmann, 1997] Lippmann, R. P. (1997). Speech recognition by machines and humans. *Speech Communication*, 22(1):1–16.

[Lööf et al., 2007] Lööf, J., Gollan, C., Hahn, S., Heigold, G., Hoffmeister, B., Plahl, C., Rybach, D., Schlüter, R., and Ney, H. (2007). The RWTH 2007 TC-STAR evaluation system for european English and Spanish. In *Proc. INTERSPEECH–2007*, pages 2145–2148.

[Maekawa, 2003] Maekawa, K. (2003). Corpus of spontaneous Japanese: Its design and evaluation. In *Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pages 7–12.

[Malkin et al., 2009] Malkin, J., Subramanya, A., and Billmes, J. (2009). On the semi-supervised learning of multi-layered perceptrons. In *Proc. INTERSPEECH-2009.*

[McDermott et al.,] McDermott, E., Hazen, T. J., Roux, J. L., Nakamura, A., and Katagiri, S. Discriminative training for large-vocabulary speech recognition using minimum classification error. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(1):203–223.

[McDermott and Katagiri, 1997] McDermott, E. and Katagiri, S. (1997). String-level MCE for continuous phoneme recognition. In *Proc. EUROSPEECH–1997*, pages 123–126.

[McDermott and Katagiri, 2005] McDermott, E. and Katagiri, S. (2005). Minimum classification error for large scale speech recognition tasks using weighted finite state transducers. In *Proc. ICASSP–2005*, volume 1, pages 113–116.

[Morgan et al., 2005] Morgan, N., Zhu, Q., Stolcke, A., Sönmez, K., Sivadas, S., Shinozaki, T., Ostendorf, M., Jain, P., Hermansky, H., Ellis, D., Doddington, G., Chen, B., Çetin, O., Bourlard, H., and Athineos, M. (2005). Pushing the envelope – aside. *IEEE Signal Processing Magazine*, 22(5):81–88.

[Nakamura et al., 2005] Nakamura, S., Takeda, K., Yamamoto, K., Yamada, T., Kuroiwa, S., Kitaoka, N., Nishiura, T., Sasou, A., Mizumachi, M., Miyajima, C., Fujimoto, M., and Endo, T. (2005). AURORA-2J: an evaluation framework for Japanese noisy speech recognition. *Information and Systems, IEICE Transactions on*, E–88–D:535–545.

[Okawa et al., 1998] Okawa, S., Bocchieri, E., and Potamianos, A. (1998). Multi-band speech recognition in noisy environments. In *Proc. ICASSP–98*, volume 2, pages 641–

644.

[Oppenheim et al., 1989] Oppenheim, A. V., Schafer, R. W., and Buck, J. R. (1989). *Discrete-time signal processing*. Prentice hall Englewood Cliffs, NJ.

[Paliwal and Alsteris, 2003] Paliwal, K. and Alsteris, L. (2003). Usefulness of phase spectrum in human speech perception. In *Proc. EUROSPEECH-2003*.

[Pearce and Hirsh, 2000] Pearce, D. and Hirsh, H. G. (2000). The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy conditions. In *Proc. ICSLP–2000*, volume 4, pages 29–32.

[Povey and Woodland, 2002] Povey, D. and Woodland, P. C. (2002). Minimum phone error and I-smoothing for improved discriminative training. In *Proc. ICASSP–02*, pages I–105–I–108.

[Reiter et al., 2007] Reiter, S., Schuller, B., and Rigoll, G. (2007). Hidden conditional random fields for meeting segmentation. In *Proc. ICME–2007*, pages 639–642.

[Rumelhart and McClelland, 1986] Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press.

[Schlüter et al., 2001] Schlüter, R., Macherey, W., Müller, B., and Ney, H. (2001). Comparison of discriminative training criteria and optimization methods for speech recognition. *Speech Communication*, 34(3):287–310.

[Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels*. MIT Press.

[Sha and Saul, 2007] Sha, F. and Saul, L. K. (2007). Large margin hidden Markov models for automatic speech recognition. *Advances in Neural Information Processing Systems*, pages 1249–1256.

[Sharma, 1999] Sharma, S. R. (1999). *Multi-stream approach to robust speech recognition*. PhD thesis, Oregon Graduate Institute of Science and Technolgy.

[Shimodaira et al., 2002] Shimodaira, H., Noma, K., Nakai, M., and Sagayama, S. (2002). Dynamic time-alignment kernel in support vector machine. *Advances in Neural Information Processing Systems*, 2:921–928.

[Suzuki et al., 2006] Suzuki, H., Ma, F., Izumi, H., Yamazaki, O., Okawa, S., and Kido, K. (2006). Instantaneous frequencies of signals obtained by the analytic signal method. *Acoustical Science and Technology*, 27(3):163–170.

[Tsochantaridis et al., 2005] Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484.

[Valente and Wellekens, 2003] Valente, F. and Wellekens, C. (2003). Maximum entropy discrimination (MED) feature subset selection for speech recognition. In *Proc. IEEE ASRU–2003*, pages 327–332.

[Vapnik, 1999] Vapnik, V. N. (1999). *The Nature of Statistical Learning Theory – Second*

Edition. Springer.

[Vuuren and Hermansky, 1997] Vuuren, S. and Hermansky, H. (1997). Data-driven design of RASTA-like filters. In *Proc. EUROSPEECH–1997*, pages 409–412.

[Wang et al., 2003] Wang, Y., Hansen, J., Allu, G. K., and Kumaresan, R. (2003). Average instantaneous frequency (AIF) and average log-envelopes (ALE) for ASR with the AURORA 2 database. In *Proc. EUROSPEECH–2003*, pages 25–28.

[Watanabe, 2006] Watanabe, S. (2006). *Speech Recognition Based on a Bayesian Approach*. PhD thesis, Waseda University.

[Woodland, 2002] Woodland, P. C. (2002). Large scale discriminative training of hidden Markov models for speech recognition. *Computer Speech and Language*, 16(1):25–47.

[Yoshida et al., 2002] Yoshida, K., Kazama, M., and Tohyama, M. (2002). Pitch and speech-rate conversion using envelope modulation modeling. In *Proc. ICASSP-2002*, pages 425–428.

[Zeng et al., 2005] Zeng, F., Nie, K., Stickney, G., Kong, Y., Vongphoe, M., Bhargave, A., Wei, C., and Cao, K. (2005). Speech recognition with amplitude and frequency modulations. *Proceedings of the National Academy of Sciences of the United States of America*, 102(7):2293.