Master Thesis

Research on Dynamic Features Derived From Speech Structure

音声の構造的表象から導出される動的特徴に関する研究

48-106415 Shinya Shimizu

Department of Information and Communication Engineering,
Graduate School of Information Science and Technology,
the University of Tokyo

Feb. 8th, 2012

Supervisor : Prof. Nobuaki Minematsu

Abstract

Due to the spread of smartphones, automatic speech recognition (ASR) systems are getting more and more popular as an interface to computers. While their successes have shown that the ASR systems have reached a practical level, the basic algorithm of state-of-the-art ASR systems is still Hidden Markov Model (HMM) based algorithm, which has been the de facto standard algorithm for ASR since 1980s. The HMM-based algorithms assume the frame-by-frame Markov property to decrease the calculation amount to the realistic level. Because of the assumption, long-term features, which cannot be defined for each time frame, such as duration of words, can never be considered. Researchers have developed various methods to improve the performance of ASR systems with the constraint of Markov property. However, the ASR algorithms are undergoing a paradigm shift. The new paradigm algorithms don't assume the Markov property have been proposed, and they showed better performance than HMM-based old paradigm algorithms in the practical calculation time. Those new paradigm algorithms can consider long-term features, which can never be considered in the old paradigm algorithms. Therefore, effective long-term features are now being investigated by researchers.

Speech structure is one of the long-term features, which can potentially be a effective feature for the new paradigm algorithm. Speech structure was proposed as a feature that is invariant for non-linguistic variations, such as the difference of speakers, recording environment, etc. While the speech structure has been applied to several applications, such as pronunciation proficiency assessment, and has shown the good performance, it has not been applied to continuous speech recognition, because it is not a frame-by-frame feature but a long-term feature and cannot be used as a feature for the old paradigm algorithms. On the contrary, the new paradigm algorithms can leverage the speech structure. An preliminary experiment on combining the speech structure with a new paradigm algorithm was already carried out and showed the good performance.

However, the current implementation of speech structure is still immature and can be improved in some aspects. Dynamic feature is one of them. Dynamic features are defined as temporal derivatives of static features. They were firstly proposed in 1986, and are now effectively used in almost all the speech systems including ASR, speech synthesis, speaker identification, etc. However, no algorithms to leverage dynamic features in speech structure was proposed, and dynamic features are omitted in previous studies on speech structure.

To solve the problem, I propose two algorithms to leverage dynamic features derived from speech structure, differential speech structure and trajectory speech structure. By using these algorithms, the dynamic features, can be effectively used for speech systems based on speech structure. Several experiments were carried out to show the effectiveness of proposed methods. By using the differential speech structure 11.0% relative decrease in word error rate was obtained in an experiment of isolated word recognition. Furthermore, by using the trajectory speech structure, 28.5% relative decrease in word error rate was obtained in an experiment of N-best rescoring of isolated word recognition. These results show that the proposed method works effectively and contributes to the speech structure as the feature for the new paradigm algorithms.

Contents

List of Figures

List of Tables

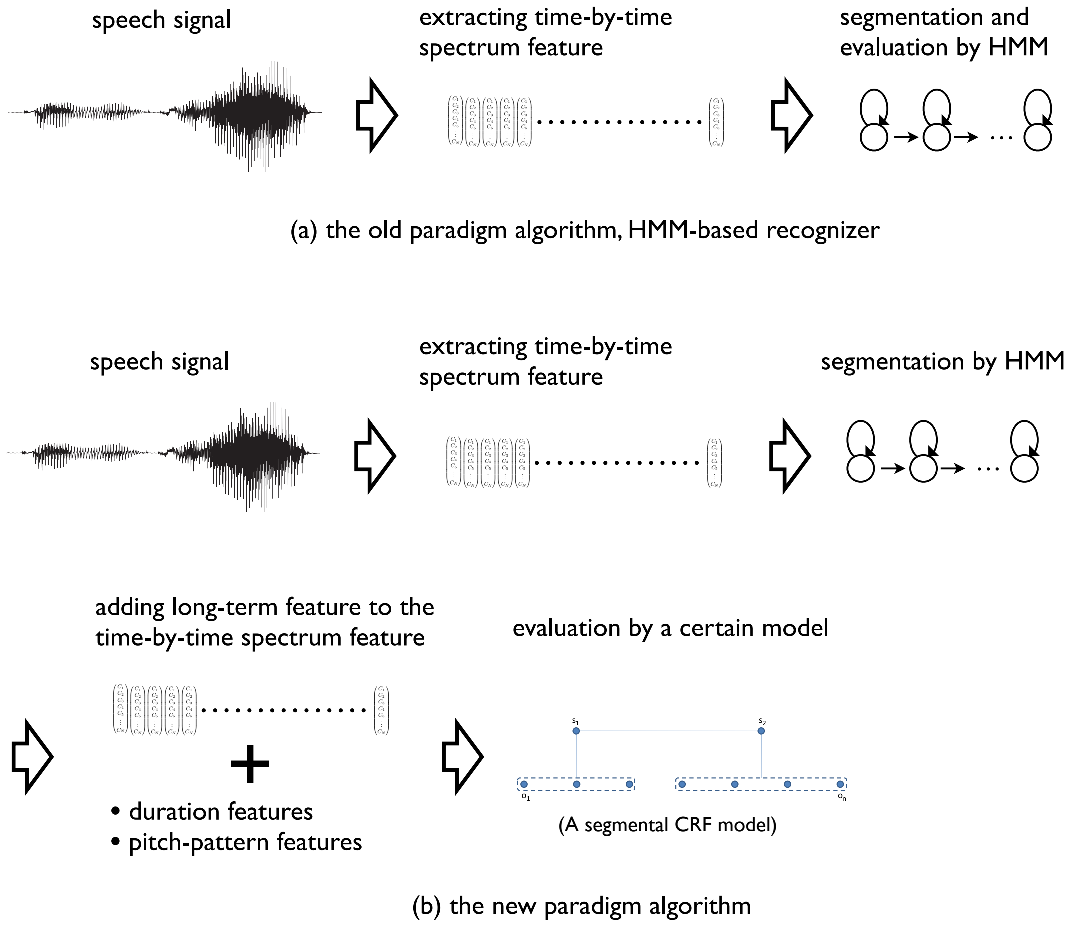
Chapter 1

Introduction

1.1 Background

The smartphone revolution has completely wiped out the skepticism whether automatic speech recognition (ASR) systems can be a next-generation interface to computers. Siri, which is an intelligent software assistant for iOS including ASR and natural language understanding system, is widely used among iPhone users to operate iPhone by their voice. Google, which provides Android OS and is the almost only competitor to Apple in the smartphone OS market, is providing Google Voice Search and it is also widely used among Android users. Although users often feel reluctant to speak to their laptop or desktop computers, they can naturally speak to their smartphones. Thanks to smartphones, the public interest and demand for ASR systems are rapidly increasing.

While their successes have shown that the ASR systems have reached a practical level, the basic algorithm of state-of-the-art ASR systems such as [1, 2, 3] are still based on Hidden Markov Model (HMM), which has been the de facto standard algorithm since 1980s. One of the intrinsic difficulties in ASR is the segmentation of utterances. In a simple term, the basic strategy of HMM-based recognizers is a template matching, which collects the features of each phonemes in advance and selects the most likely phoneme by comparing the collected features and the input features. However, because the length of a phoneme is not constant, to evaluate how likely an utterance is to consist of a certain word sequence, recognizers have to determine a segmentation of the utterance, which part of the utterance corresponds to which phoneme. Although recognizers can determine the best segmentation by evaluating all the possible segmentations, it is virtually impossible because of its large calculation amount. The HMM-based recognizers avoid this problem by assuming the feature sequence, which usually consists of FFT spectrum-based features of short time frames, are generated by a Markov process. Because it means the generation of the feature at each time depends only on its present state and is independent of its past and future, the segmentation problem can be formulated as a shortest path problem for a graph with edge path costs and the best segmentation can be calculated not by evaluating all the


speech signal
extracting time-by-time spectrum feature
segmentation and evaluation by HMM
speech signal
extracting time-by-time spectrum feature
segmentation by HMM
adding long-term feature to the time-by-time spectrum feature
+
• duration features
• pitch-pattern features
evaluation by a certain model
s_1
s_2
o_1
o_n
(A segmental CRF model)

(a) the old paradigm algorithm, HMM-based recognizer

(b) the new paradigm algorithm

Fig.1.1: The overview of ASR algorithms in the old paradigm and the new paradigm

possible segmentations but by a Dijkstra-like fast algorithm. However, the assumption of the Markov property has several disadvantages. By definition, the Markov processes cannot model the long-term relationships. For example, pitch pattern, which often distinguishes interrogative sentences and declarative sentences in many languages and sometimes distinguishes the meanings of words in pitch accent languages such as Japanese, is considered as a long-term feature and cannot be modeled by HMM. Actually, HMM-based recognizers cannot distinguish homophones that can be distinguished by pitch pattern, and can distinguish them only by their context. Although the HMM-based recognizers have these disadvantages, various extensions to HMM are developed and they have made the HMM-based recognizers successfully applied to many systems.

However, the ASR algorithms are undergoing a paradigm shift. While the old paradigm algorithms, HMM-based recognizers, execute a segmentation and its evaluation simultaneously by assuming the Markov property, the new paradigm algorithms do them separately. In the new paradigm, firstly they segment an utterance by assuming the Markov property and obtain a segmentation, and secondly they concatenate additional features including long-term features and evaluate them without assuming the Markov property[4, 5, 6, 7] (Fig.1.1) . In other words, they use the old paradigm recognizers as a segmentation machine. In the new paradigm, long-term features that cannot be modeled in the old paradigm, such as duration of words, can be modeled[6]. Speech structure[8] is one of the long-term features, which can potentially be a effective feature for the new paradigm algorithm.

Speech signals contain various kinds of information, such as linguistic messages, speaking styles, speaker identities, recording conditions, etc. For example, in many cases, spectrums of women's speeches have higher energy in high frequency region than that of men's even if the contents of the utterance were completely same. Because ASR is a system that extracts linguistic messages from speech signals, such non-linguistic variations have to be canceled. To solve the problem, feature adaptations, which transform the features of utterances by a certain mapping function, and model adaptations, which transform the recognizer's model to get close to the speaker of input utterance, are often used[9, 10, 11, 12, 13, 14]. However, such adaptations are reported to be still ineffective in some applications, such as children's speech recognition [15]. To solve this problem, speech structure, which is completely different from the conventional adaptation methods, was proposed[8]. In speech structure, an input utterance is converted into several distributions of spectrum feature, and the distances between the distributions are adopted as features that represent the utterance. By using f-divergence , which is mathematically proved to be invariant with any invertible and differentiable transformation, as the distance measure, the speech structure is a invariant representation for non-linguistic information, which is often approximated as some transformation. It was shown that speech structure works effectively in pronunciation assessment[16, 17, 18, 19]. In addition, several ASR algorithms based on speech structure have been proposed[20, 21, 22, 23, 24, 25]. Although it is difficult to combine speech structure with the ordinary HMM-based ASR systems because speech structure is not a frame-by-frame feature, it is possible to combine it with the new paradigm algorithms, which can consider long-term features. Speech structure can potentially be a effective feature used in the new paradigm algorithms. However, the implementations of the speech structure is still immature and can be improved in some aspects. Dynamic feature is one of them. Dynamic features are defined as temporal derivatives of static features. They were firstly proposed in 1986[26], and are now effectively used in almost all the speech systems including ASR, speech synthesis, speaker identification, etc. However, no algorithms to leverage dynamic features in speech structure was proposed, and dynamic features are omitted in previous studies on speech structure.

1.2 Objectives of the study

The objectives of the study are to investigate the methods to leverage dynamic features in speech structure, and to improve the performance of ASR using speech structure. I propose two algorithms to leverage dynamic features derived from speech structure, differential speech structure and trajectory speech structure. By using these algorithms, the dynamic features, which were omitted in previous studies, can be effectively used for speech systems based on speech structure.

1.3 Organization of the thesis

This thesis consists of 6 chapters. In chapter 1 (this chapter), the background and the objectives are introduced. In chapter 2, the ASR systems in both the old paradigm and the new paradigm are introduced. In chapter 3, the basic algorithm of speech structure and its applications are introduced. In chapter 4, two proposed algorithms to derive dynamic features from speech structure are introduced. In chapter 5, three experiments are introduced and the effectiveness of proposed methods is shown. Finally in chapter 6, the whole thesis is summarized and several future works are introduced.

Chapter 2

ASR systems

In this chapter, the basic procedures of ASR in both the old and the new paradigm are introduced. An ASR system consists of an acoustic model, which models the acoustic feature, a language model, which models the acceptable word sequence, and a decoder, which searches for the solution using an acoustic model and a language model. Because acoustic models are focused on in this thesis, acoustic models are mainly introduced and the details of other parts are omitted.

2.1 Basic procedure of HMM-based ASR

The basic procedure of ASR in the old paradigm is shown in Fig.2.1. Firstly, X, a sequence of acoustic features which are usually FFT spectrum-based features, are extracted from speech signals. Secondly, using an acoustic model and a language model, a decoder decides which word sequence is most likely for the feature sequence. The process of obtaining the most likely word sequence is formulated as

$$\hat{W} = \operatorname*{argmax}_{W} P(W|X), \tag{2.1}$$

where W is a word sequence, X is the input feature sequence, and $\hat{W}$ is the word sequence selected as the recognition result. Because it is difficult to directly model $P(W|X)$, the probability of a word sequence W given a feature sequence X, Eq.(2.1) is changed using Bayes' rule as

$$\begin{aligned} \hat{W} &= \operatorname*{argmax}_{W} P(W|X) \\ &= \operatorname*{argmax}_{W} \frac{P(X|W)P(W)}{P(X)} \\ &= \operatorname*{argmax}_{W} P(X|W)P(W). \end{aligned} \tag{2.2}$$

A model to describe $P(X|W)$, the probability of an acoustic feature sequence X generated from a word sequence W, is called acoustic model. On the other hand, a model to describe

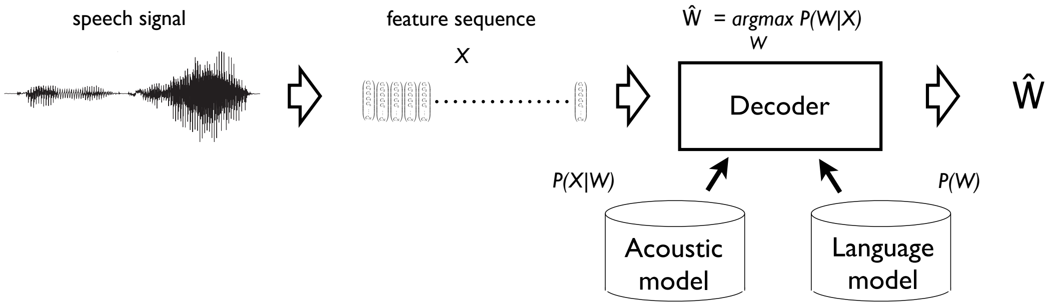


Fig.2.1: The overview of ASR in the old paradigm

$P(W)$, the probability of a word sequence W, is called language model. In this paradigm, a word sequence W that maximize the joint probability $P(X, W) = P(X|W)P(W)$ is selected as the recognized result.

2.2 Acoustic features

2.2.1 Cepstrum

Cepstrum is the most widely used acoustic feature in ASR. The procedure to calculate a cepstrum from a waveform is shown in Fig.2.2. Firstly, a short time period, which is usually several tens of milliseconds and called as "frame", is clipped from the input waveform. Secondly, the spectrum is calculated by Short Time Fourier Transform (STFT). Finally, the cepstrum is obtained by applying Inverse STFT to the log spectrum. Because the shape of vocal tract, which makes the feature of phonemes, affects the low-order parts of cepstrum, usually the low-order parts are adopted as the cepstral feature.

2.2.2 Mel scale cepstrum

It is known that humans' perception of sounds is not linear in frequency domain. The perception is log-like, and its resolution is high in low frequency region and low in high frequency region. Mel frequency f_{mel} is defined as

$$f_{mel}(f) = 2595 \log_{10}(1 + \frac{f}{700}). \tag{2.3}$$

Several acoustic features that consider these characteristics of humans' perception are proposed and Mel-Frequency Cepstrum Coefficient (MFCC) is the mostly used one. To calculate MFCC, mel frequency spectrum is first obtained by a filter bank analysis using mel-scaled triangle windows shown in Fig.2.3. Then, MFCC is obtained by applying Discrete Cosine Transform (DCT) to the mel frequency spectrum.

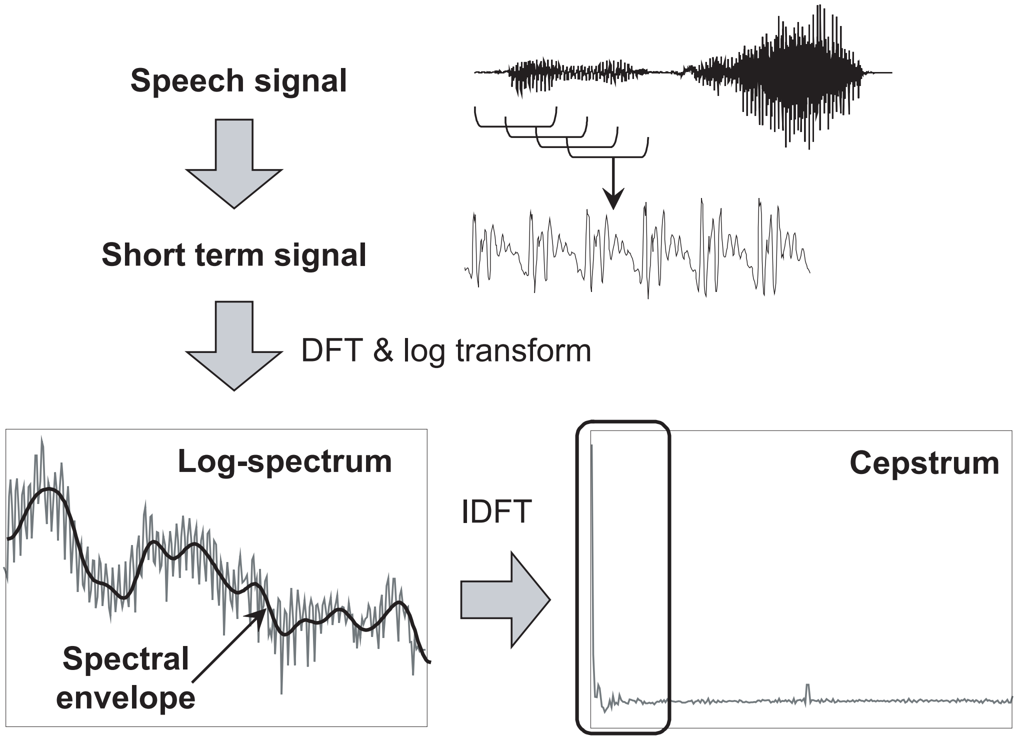


Fig.2.2: Cepstrum analysis

2.2.3 Delta cepstrum

Although a cepstrum represent a spectral feature at a time frame, it is known that the time variation of the feature is also important for humans' perception. As a feature that represents the time variation, delta cepstrum ($\Delta \boldsymbol{c}$) and delta delta cepstrum ($\Delta^2 \boldsymbol{c}$) were proposed. They are defined as first-order and second-order coefficients of the quadratic approximation for the time frame and its adjacent L frames, and calculated as

$$\Delta \boldsymbol{c}_t = \frac{\sum_{\tau=-L}^{L} \tau \boldsymbol{c}_{t+\tau}}{\sum_{\tau=-L}^{L} \tau^2} \tag{2.4}$$

$$\Delta^2 \boldsymbol{c}_t = \frac{\sum_{\tau=-L}^{L} (a_0 \tau^2 - a_1) \boldsymbol{c}_{t+\tau}}{\sum_{\tau=-L}^{L} (a_2 a_0 - a_1^2)}, \tag{2.5}$$

where $a_2 = \sum_{\tau=-L}^{L} \tau^4$, $a_1 = \sum_{\tau=-L}^{L} \tau^2$, $a_0 = \sum_{\tau=-L}^{L} 1$, and $\boldsymbol{c}_t$ is the cepstrum of t-th frame. It is shown that using the concatenation, $\boldsymbol{x}_t = \left[\boldsymbol{c}_t^{\mathrm{T}}, \Delta \boldsymbol{c}_t^{\mathrm{T}}, \Delta^2 \boldsymbol{c}_t^{\mathrm{T}}\right]^{\mathrm{T}}$ instead of $\boldsymbol{c}_t$ improves the performance of recognizers significantly[26, 27].

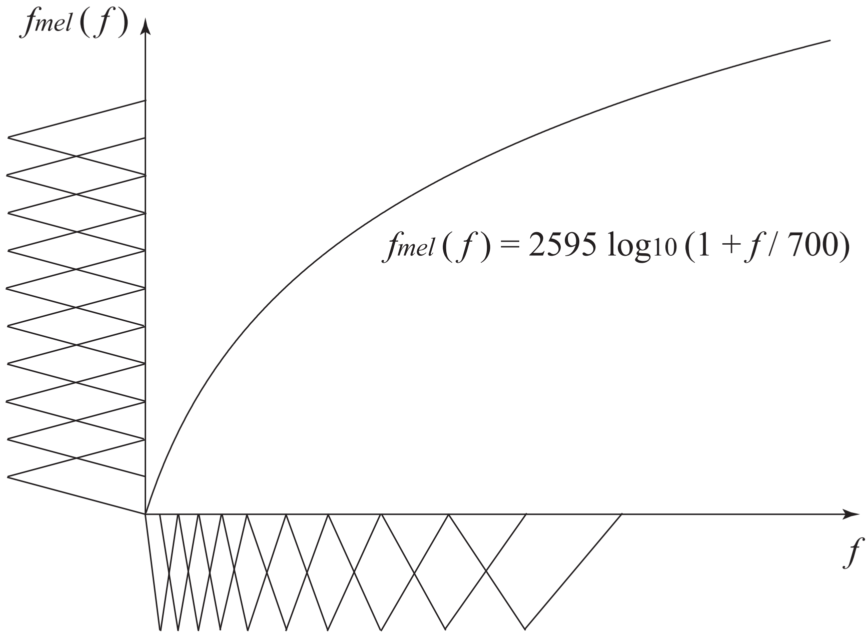


Fig.2.3: Mel frequency and triangle windows in mel scale

2.3 Acoustic model

2.3.1 Hidden Markov Model

Acoustic models describe $P(X|W)$, the probability of an acoustic feature sequence X given a word sequence W, and Hidden Markov Model (HMM) is the mostly used model (Fig. 2.4). HMM is a finite state machine, which has output distributions for its states. In Fig.2.4, S_i is the i-th state, a_i is the transition probability from S_i to S_{i+1}, $b_i(x)$ is the output probability of a feature x from state S_i. Although any transitions from any states are accepted in general HMMs, only transitions from S_i to S_{i+1} are accepted in the HMMs used as acoustic models. That is because each state represents a phoneme (or a sub-phoneme) and the order of the phonemes have to be maintained. Gaussian distribution or Gaussian mixture distribution are usually used as the output distributions $b_i(x)$. For ASR, HMMs for each unit, such as word or phoneme, are trained using a certain speech database in advance.

2.3.2 Training of HMM

In the training of a HMM, parameters $\theta = \{a_i, b_i(x)\}$ are trained with some criteria. Although several new criterion to train HMM has been proposed recently, Maximum Likelihood (ML) estimation, which has been a de facto standard since 1980s, is introduced

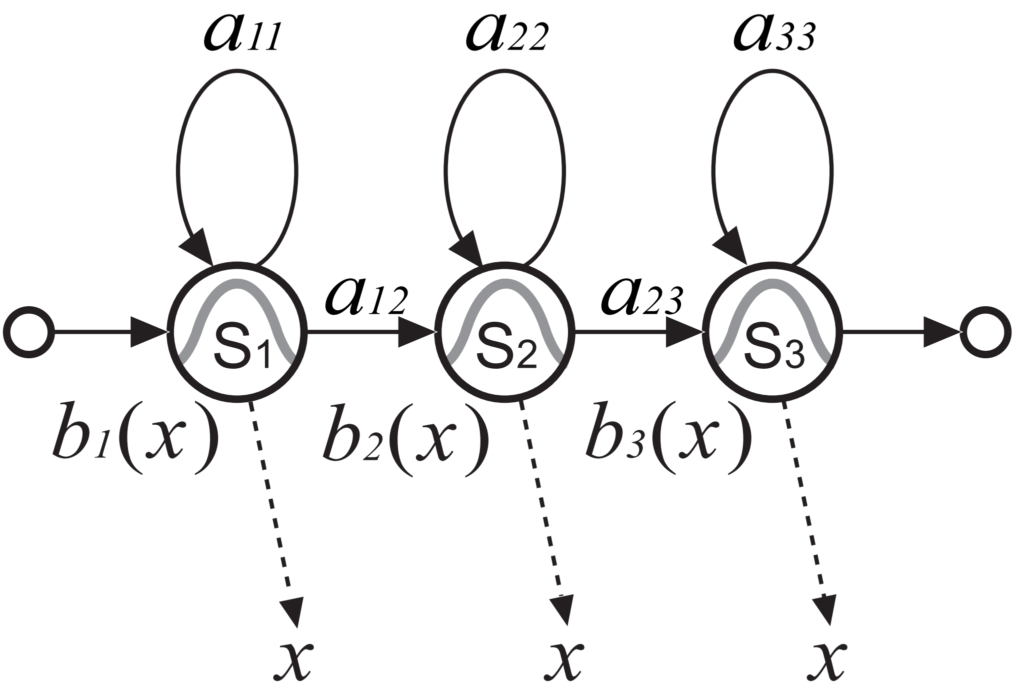


Fig.2.4: A left-to-right Hidden Markov Model

here. In ML estimation, θ is estimated to maximize the likelihood of the model for given data X as

$$\hat{\theta} = \operatorname*{argmax}_{\theta} P(X|\theta), \tag{2.6}$$

where θ is the trained parameter. However, Eq.2.6 cannot be solved analytically. To estimate the distributions $\{b_i(x)\}$, the alignment, which features belong to which state, is needed, and to estimate the alignment, the distributions are needed. Therefore, Baum-Welch algorithm, which estimates the alignment and the parameters iteratively, are used to obtain a local optimum. In isolated word recognition systems, HMMs are constructed for each word. On the other hand, in large vocabulary continuous speech recognition (LVCSR) systems, it is usually challenging to prepare speech corpus large enough to construct HMMs for each words. Therefore, HMMs are constructed for each phoneme instead of word and each word are represented by a concatenation of phoneme HMMs. Furthermore, because the feature of a phoneme changes depending on its adjacent phonemes, HMMs are usually constructed for a triple, a phoneme, its previous phoneme, and its next phoneme.

2.3.3 Decoding by HMM

As shown in Eq.(2.2), $P(X|W)$ have to be calculated by an acoustic model. In this section, the procedure of calculating $P(X|W)$ by a HMM is introduced with an example,

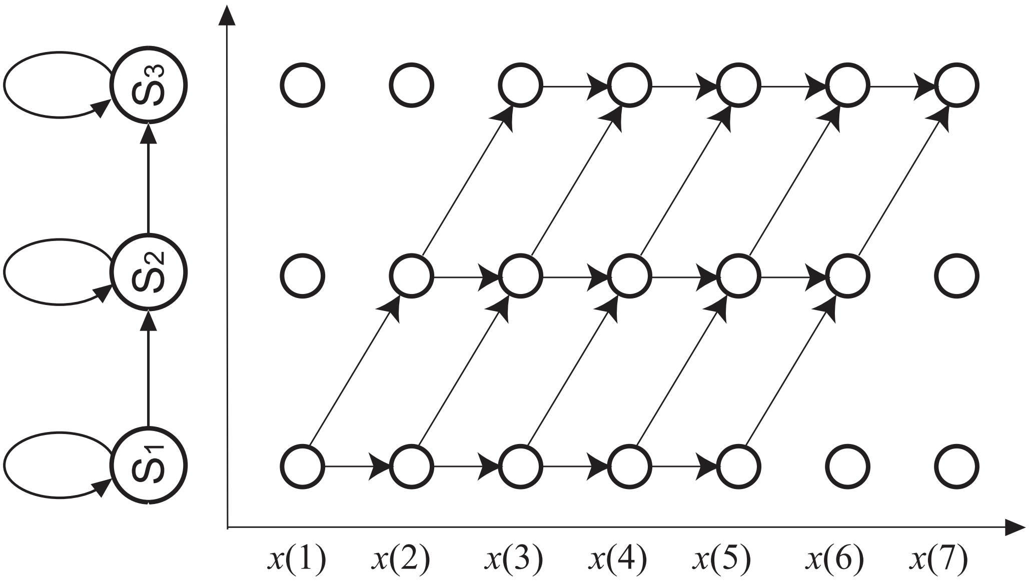


Fig.2.5: The trellis paths of a HMM's state transition

assuming an input feature sequence of 7 frames, $X = \{x(1), x(2), \cdots, x(7)\}$, and a HMM with 3 states (Fig.2.5). The horizontal paths and the oblique paths correspond to the intra-state transitions and the inter-state transitions in HMM. Let q_t denote a state index (1 to 3 in this sample) of t-th frame, and $\boldsymbol{q}$ denote their concatenation $\{q_1, q_2, \cdots, q_7\}$. The output probability $P(X|W)$ is given by

$$P(X|W) = \sum_{\text{all } \boldsymbol{q}} P(X|\boldsymbol{q}, W)P(\boldsymbol{q}|W). \tag{2.7}$$

Eq.(2.7) means that $P(X|W)$ is obtained by adding the probabilities of all the possible paths. However, Because it is computationally hard to calculate all the path in actual cases, it is usually approximated as

$$P(X|W) \approx P(X|\boldsymbol{q}^\star, W)P(\boldsymbol{q}^\star|W), \tag{2.8}$$

where $\boldsymbol{q}^\star$ is given by

$$\boldsymbol{q}^\star = \operatorname*{argmax}_{\boldsymbol{q}} P(X|\boldsymbol{q}, W)P(\boldsymbol{q}|W). \tag{2.9}$$

The most likely segmentation $\boldsymbol{q}^\star$ can be solved by applying Viterbi algorithm, which is a Dijkstra-like shortest path search algorithm, to the trellis in Fig.2.5.

2.4 Non-linguistic variations and adaptation methods for them

So far, the basic algorithm of ASR, extracting cepstrum and modeling it by HMM, has been introduced. However, the algorithm is not enough in actual use, because cepstrums change depending not only on the content of the utterance, but also on other non-linguistic information. For example, if a recognizer tries to recognize an utterance of a person whose utterance doesn't appear in the training corpus, the performance will drop significantly. To solve this problem, various adaptation methods have been proposed. Most of them assume that the cepstrums are distorted by convolutional noise and linear transformation noise as

$$\boldsymbol{c}' = \boldsymbol{A}\boldsymbol{c} + \boldsymbol{b}, \tag{2.10}$$

where $\boldsymbol{c}$ is the clean cepstrum, $\boldsymbol{c}'$ is the observed cepstrum, $\boldsymbol{A}$ is the linear transformation noise, and $\boldsymbol{b}$ is the convolutional noise. Because cepstrum is defined as the IDFT of a log power spectrum, a convolutional noise, which is represented as a transfer function in spectrum domain, is represented as an addition $\boldsymbol{c}' = \boldsymbol{c} + \boldsymbol{b}$ in cepstrum domain. The characteristic of a microphone is a typical one. On the other hand, the linear transformation $\boldsymbol{c}' = \boldsymbol{A}\boldsymbol{c}$ is typically caused by the difference of vocal tract length. Longer vocal tract warps the log spectrum toward the low frequency region. It is shown that any monotonically increasing continuous warping in the log spectrum domain is represented as a linear transformation in cepstrum domain[28].

To suppress the noises, various methods have been proposed. In the rest of this section, cepstrum mean normalization (CMN)[29], vocal tract length normalization (VTLN)[11], and maximum likelihood linear regression (MLLR)[14] are introduced as examples.

2.4.1 Cepstrum mean normalization

CMN is a simple and effective normalization method for convolutional noises. For each utterance, CMN just subtracts the mean vector of the cepstrum sequence $\bar{\boldsymbol{c}} = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{c}_t$ from each cepstrum $\boldsymbol{c}_t$. Ideally, if the convolutional noise is stationary and the utterance is long enough, CMN can remove the convolutional noise completely.

2.4.2 Vocal tract length normalization

VTLN is a method to normalize a linear transformation noise caused by the difference of vocal tract length. VTLN estimates the vocal tract length of a speaker and transform the vocal tract length to get close to the reference speaker's one. Practically, the process of VTLN is not divided into two parts, estimating the vocal tract length and transforming it, but done directly by transforming the input cepstrum to get close to the reference speaker's cepstrum[30].

2.4.3 Maximum likelihood linear regression

MLLR is a method of model adaptation for both convolutional and linear transformation noise. While CMN and VTLN transform the input cepstrum to decrease the noises and fit to the model, MLLR transforms the model to fit to the input cepstrum. Assuming that a HMM is used as the acoustic model, MLLR modifies the output distributions by applying an affine transformation to the mean vectors of the output distributions, $\boldsymbol{\mu}' = \boldsymbol{A}\boldsymbol{\mu}+\boldsymbol{b}$. The transformation parameters $\boldsymbol{A}$ and $\boldsymbol{b}$ are estimated to maximize the likelihood $P(X|W)$.

2.5 ASR in the new paradigm

So far, the basic ASR algorithms in the old paradigm have been introduced. These algorithms have some limitations due to their assumption of frame-by-frame Markov property. They can never consider long-term features, such as durations and pitch patterns. However, the ASR algorithm is undergoing a paradigm shift. While old paradigm systems, HMM-based recognizers, execute a segmentation and its evaluation simultaneously by assuming the Markov property, new paradigm systems do them separately. In the new paradigm, firstly they segment an utterance by assuming the Markov property and obtain a segmentation, and secondly they concatenate additional features including long-term features and re-evaluate them without assuming the Markov property[4, 5, 6, 7]. Segmental Conditional Random Field (SCRF) is a typical new paradigm algorithm and is going to be used more because its software is provided by Microsoft Research[5]. In the rest of this section, the basic ideas and algorithms of ASR using SCRF are introduced.

HMMs model $P(o_t|s_i)$, the probability of t-th observation o_t given the i-th state s_i and finally calculate the whole generation probability $P(\boldsymbol{o}|\boldsymbol{s})$. On the contrary, SCRFs model $P(s_i|\boldsymbol{o}(s_i))$, the conditional probability of the i-th state s_i given the set of observations which belong to the i-th state, $\boldsymbol{o}(s_i)$, and finally calculate the whole conditional probability $P(\boldsymbol{s}|\boldsymbol{o})$. Fig.2.6 is a illustration of SCRF. As shown in Fig.2.6, SCRF has states $s_1, s_2, \cdots, s_N$ like HMM, and four concepts, e, s_l^e, s_r^e, and $o(e)$, to model the conditional probabilities. e denotes a edge between states, s_l^e and s_r^e denote the left and right state of a edge e, and $o(e)$ denotes the set of observations which belong to the state s_r^e. With this notation, let $f_k(s_l^e, s_r^e, o(e))$ denote any feature function which is defined for each set of $\{s_l^e, s_r^e, o(e)\}$. The conditional probability of a state sequence $\boldsymbol{s}$ given an observation sequence $\boldsymbol{o}$ for a SCRF is given by

$$P(\boldsymbol{s}|\boldsymbol{o}) = \frac{\sum_{\boldsymbol{q}} \exp(\sum_e \lambda_k f_k(s_l^e, s_r^e, o(e)))}{\sum_{\boldsymbol{s}'} \sum_{\boldsymbol{q}} \exp(\sum_e \lambda_k f_k({s'}_l^e, {s'}_r^e, o(e)))}. \tag{2.11}$$

In the training, the parameters $\{\lambda_i\}$ are estimated to maximize the likelihood $P(\boldsymbol{s}|\boldsymbol{o})$ with $L1$ and $L2$ norm regularization. In [5], gradient descent is used to estimate the parameters. In ASR using SCRF, each state s_i, observation o_t, and feature function $f_k(s_l^e, s_r^e, o(e))$,

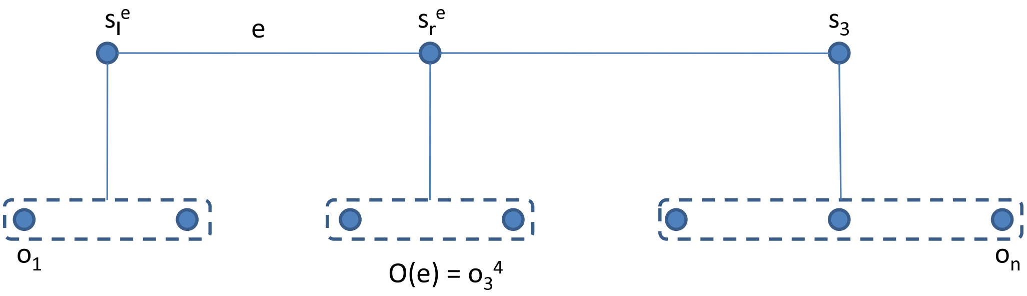


Fig.2.6: A Segmental CRF

correspond to a word, a frame, and any feature respectively. Because the feature functions $f_k(s_l^e, s_r^e, o(e))$ is defined not for each frame but for each segment, it can be a long-term feature, such as the duration of a segment (Fig.2.7). In [5], to keep a baseline level of performance, a baseline feature is defined. the base line feature for a segment is always either $+1$ or -1. If a segment and the corresponding word are completely same as the 1st candidate (the best result) of the baseline recognizer, the feature is $+1$, and is -1 otherwise. In [6], $P(duration|s_i)$, the probability of a duration given a word, is used as a duration feature by assuming a Gaussian distribution for each word duration. The levenshtein distances between the phoneme sequences obtained by other recognizers are the another features used in [5]. While these various features can be defined in SCRF, it is computationally impossible to calculate these features for all candidates and segmentations. To solve it, the word trellis obtained by a baseline recognizer is used as a set of candidates and (Fig.2.8).

2.6 Summary

In this section, the basic procedures of ASR systems in both the old paradigm and the new paradigm have been introduced. The old paradigm recognizers model frame-by-frame feature sequences by HMM, applying adaptation methods to compensate the non-linguistic variations. They have some limitations due to their assumption of frame-by-frame Markov property. They can never consider long-term features, such as durations and pitch patterns. On the contrary, the new paradigm recognizers firstly obtain candidates by applying an old paradigm algorithm, secondly add several features including long-term features to the candidates and their segmentations, and finally model them by some algorithm. By using the new paradigm algorithm, long-term features can be modeled and the demand for effective long-term features are increasing. In the next chapter, speech structure, which is a long-term feature and is robust for the non-linguistic variations, is introduced.

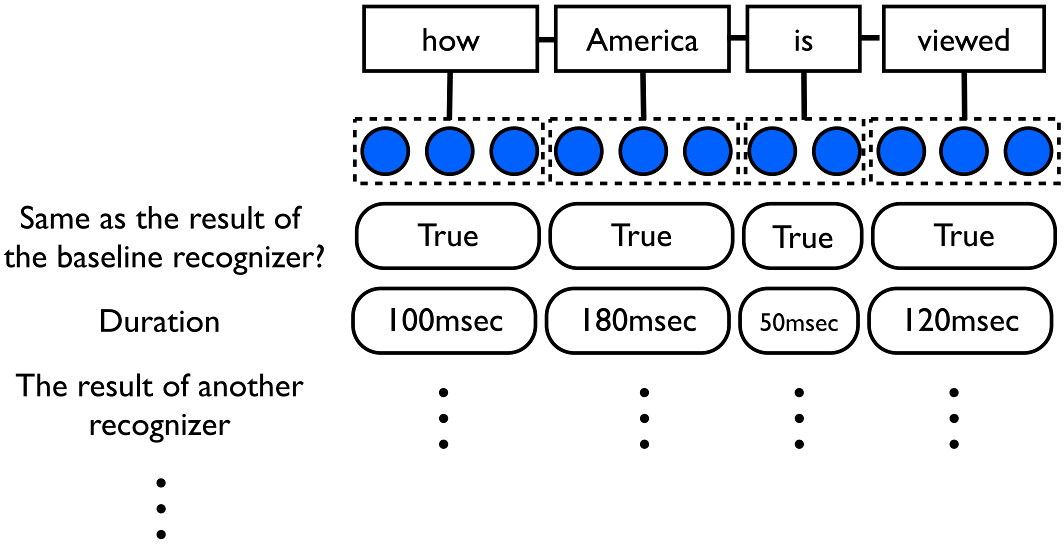

how
America
is
viewed
Same as the result of the baseline recognizer?
True
True
True
True
Duration
100msec
180msec
50msec
120msec
The result of another recognizer

Fig.2.7: Features for SCRF

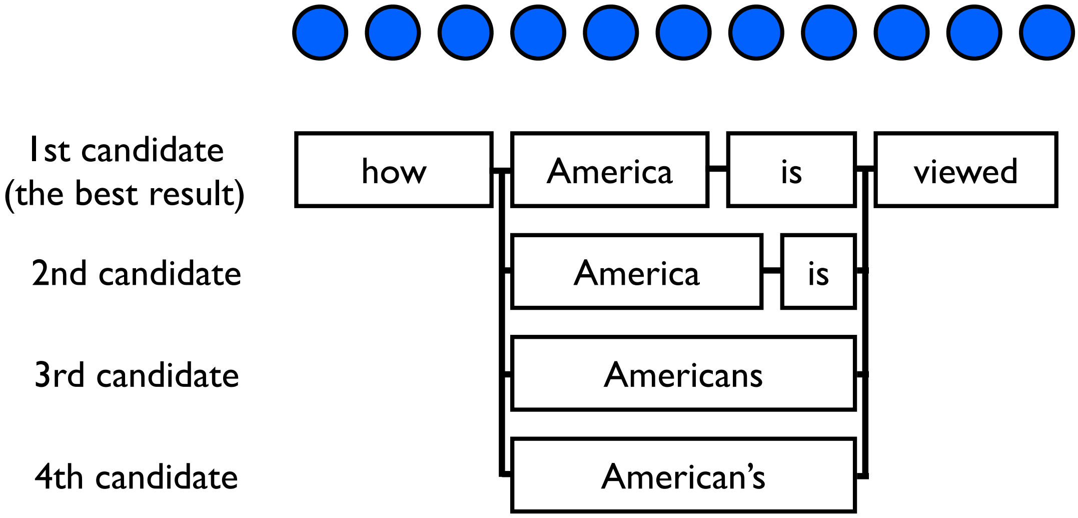


Fig.2.8: Candidates obtained by a baseline recognizer

Chapter 3

Speech structure

3.1 Introduction

Speech signals contain various kinds of information, such as linguistic messages, speaking styles, speaker identity, recording conditions, etc. When one tries to get some specific kinds of information from speech signals, one wants to extract the acoustic features that represent only the target information and are independent of the other kinds of information. ASR systems, which convert speech signals to texts, need the acoustic features that convey linguistic information only. However, mel-cepstrum-based features, which are most commonly used, are not independent at all of non-linguistic information. Therefore, researchers have developed various methods to compensate for non-linguistic variation in speech features. These methods are, for example, feature normalization, noise suppression, speech enhancement, and model adaptation. However these methods are reported to be ineffective in some applications, such as children's speech recognition [15].

To solve the problem, a method was proposed [8] to extract the acoustic features that are mathematically independent of the non-linguistic variations. The proposed representation is called speech structure. In the proposed representation, first, the speech feature sequence is converted to a sequence of distributions, from each pair of which a distance is calculated using f-divergence . The obtained distance matrix is adopted as a speech representation of the input utterance. f-divergence is mathematically proved to be invariant with any continuous and differentiable transformation, as which any non-linguistic speech variation is usually approximated. These facts indicate that the f-divergence distance matrix can be regarded as invariant representation with non-linguistic variations. It was shown that speech structure can be effectively used for pronunciation assessment[16, 17, 18, 19]. In addition, several ASR algorithms based on speech structure have been proposed[20, 21, 22, 23, 24, 25].

In this section, speech structure, pronunciation assessment based on speech structure, isolated word recognition, and continuous speech recognition are introduced.

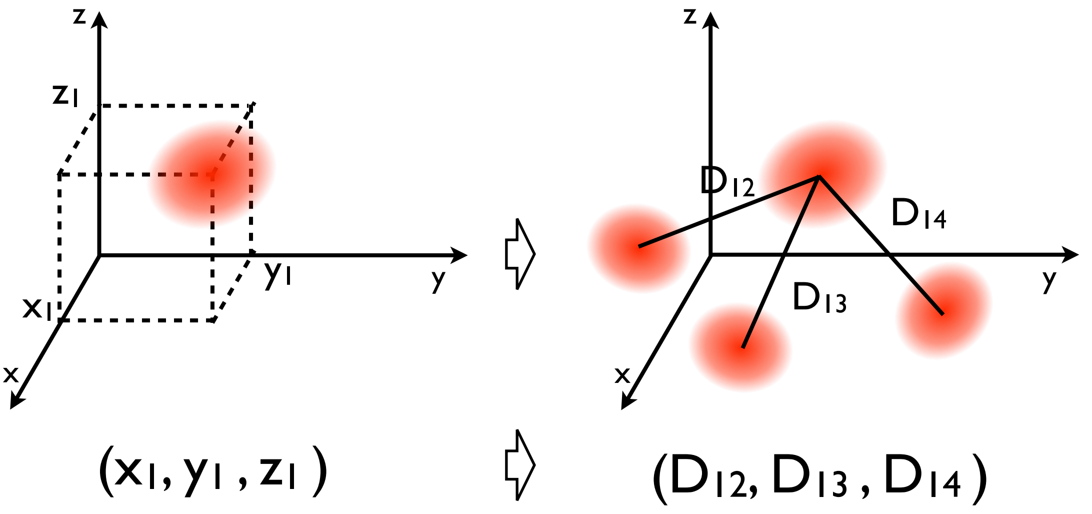


Fig.3.1: While an absolute coordinate is usually used as a representation of a phoneme, the distances between the phoneme and other phonemes are used as a representation in speech structure

Table3.1: Examples of f-divergences

kind of divergence	$g(t)$
Bhattacharyya coefficient	$\sqrt{t}$
KL-divergence	$t\log(t)$
Symmetric KL-divergence	$t\log(t) - \log(t)$
Hellinger distance	$(\sqrt{t}-1)^2$
Total variation	$\lvert t-1\rvert$
Pearson divergence	$(t-1)^2$
Jensen-Shannon divergence	$\frac{1}{2}\left(t\log\frac{2t}{t+1}+\log\frac{2}{t+1}\right)$

3.2 Speech structure

Non-linguistic variations of cepstrum are compensated by applying a mapping in most of adaptation methods. For example, MLLR adaptation assumes the linear transformation as the mapping and transform the model by estimating the parameters of linear transformation. On the contrary, in a speech structure, only distances between two distributions, which often refer to phonemes, are calculated and absolute features are discarded instead (Fig.3.1). f-divergence is used as the distance measure between distributions. The f-divergence between the two distributions p_i, p_j is given by

$$f\text{-div}(p_i, p_j) = \int p_j(\boldsymbol{x})\, g\left(\frac{p_i(\boldsymbol{x})}{p_j(\boldsymbol{x})}\right) d\boldsymbol{x}, \tag{3.1}$$

where g is a convex function, which determine the kind of f-divergence . As shown in Table.3.1, there exist various f-divergences , including some well-known measures, such as Kullback-Leibler (KL) divergence and Jensen-Shannon divergence. In speech structure, Bhattacharyya distance, which is the logarithm of Bhattacharyya coefficient and was found to work well by previous studies, is used. Let $p_i(\boldsymbol{x})$ and $p_j(\boldsymbol{x})$ denote Gaussian distributions over $\boldsymbol{R}^D$ with their means of $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$, and their variance matrices of Σ_i and Σ_j. The Bhattacharyya distance between $p_i(\boldsymbol{x})$ and $p_j(\boldsymbol{x})$ is given as

$$BD(p_i(\boldsymbol{x}), p_j(\boldsymbol{x})) = \frac{1}{8}(\boldsymbol{\mu_i} - \boldsymbol{\mu_j})^T \Sigma^{-1} (\boldsymbol{\mu_i} - \boldsymbol{\mu_j}) + \frac{1}{2}\log(\frac{|\Sigma|}{|\Sigma_i|^{1/2}|\Sigma_j|^{1/2}}), \tag{3.2}$$

where $\Sigma = (\Sigma_i + \Sigma_j)/2$. Using Eq.(3.2), $\binom{N}{2}$ distances are obtained from N distributions in a utterance, and they are used as the representation of the utterance.

In the rest of this section, a proof that f-divergence is invariant for any invertible transformation is introduced. Assume a space X, two distributions $p_i(\boldsymbol{x})$ and $p_j(\boldsymbol{x})$ in X, and a mapping function $h : X \to Y$, which convert $\boldsymbol{x}$ into $\boldsymbol{y}$ as

$$\boldsymbol{y} = h(\boldsymbol{x}). \tag{3.3}$$

Under Eq.(3.3), distribution $q_i(\boldsymbol{y})$ in Y is given by

$$q_i(\boldsymbol{y}) = p_i(h^{-1}(\boldsymbol{y}))J(\boldsymbol{y}), \tag{3.4}$$

where $J(\boldsymbol{y})$ is the determinant of the Jacobian matrix of function $h^{-1}(\boldsymbol{y})$. Then we obtain,

$$\begin{aligned} f\text{-div}(p_i, p_j) &= \int p_j(\boldsymbol{x}) g\left(\frac{p_i(\boldsymbol{x})}{p_j(\boldsymbol{x})}\right) d\boldsymbol{x} \\ &= \int p_j(h^{-1}(\boldsymbol{y})) g\left(\frac{p_i(h^{-1}(\boldsymbol{y}))J(\boldsymbol{y})}{p_j(h^{-1}(\boldsymbol{y}))J(\boldsymbol{y})}\right) dJ(\boldsymbol{y})\boldsymbol{y} \\ &= \int q_j(\boldsymbol{y}) g\left(\frac{q_i(\boldsymbol{y})}{q_j(\boldsymbol{y})}\right) d\boldsymbol{y} \\ &= f\text{-div}(q_i, q_j). \end{aligned} \tag{3.5}$$

Therefore, it was shown that f-divergence is invariant for any invertible and differentiable mapping. Although it can be also shown that a functional with two arguments $f(p_i, p_j)$ is invariant for any invertible and differentiable mapping only if f is f-divergence , the proof is omitted here.

3.3 Pronunciation assessment using speech structure

One of the application of speech structure is a pronunciation assessment. Here, the basic procedure of structure-based pronunciation assessment is introduced along with [31]. The

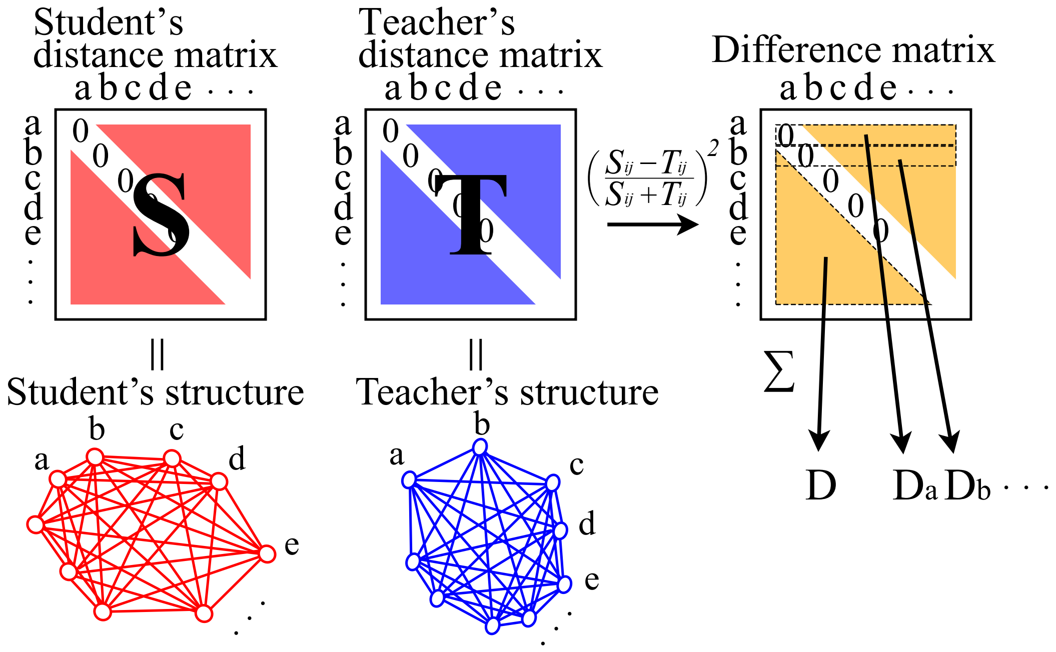


Fig.3.2: Pronunciation assessment using speech structure

pronunciation assessment task is defined as an estimation of a student's pronunciation score which was given by a teacher. A diagram of structure-based pronunciation assessment is shown in Fig.3.2. A student's distance matrix S and a teacher's one T are constructed with their utterances. In [31], 44 phonemes are pronounced by each student and teacher, and the shape of a distance matrix is 44 × 44. From a student's distance matrix and a teacher's one, a difference matrix is calculated. The difference matrix between S and T, $\{D_{ij}\}$ is defined as

$$D_{ij} = \left(\frac{S_{ij} - T_{ij}}{S_{ij} + T_{ij}}\right)^2. \tag{3.6}$$

In naive implementation, the root mean square of the difference matrix,

$$D_{rms} = \sqrt{\frac{1}{M}\sum_{i<j} D_{ij}}, \tag{3.7}$$

is used as a explanatory variable for the pronunciation score. Although regression analyses can be used to improve the performance, it is practically impossible to apply a regression analysis with its explanatory variables of all the distance matrix's elements because the number of explanatory variables are too many to train the regression machine correctly. To solve the problem, multi-layered regression analysis are proposed (Fig.3.3). As shown in Fig.3.3, two steps of regression analysis are applied from the difference matrix to the score, which means that the regression parameters are shared in rows and columns respectively instead of assuming the different parameters for all the elements in the matrix. The first

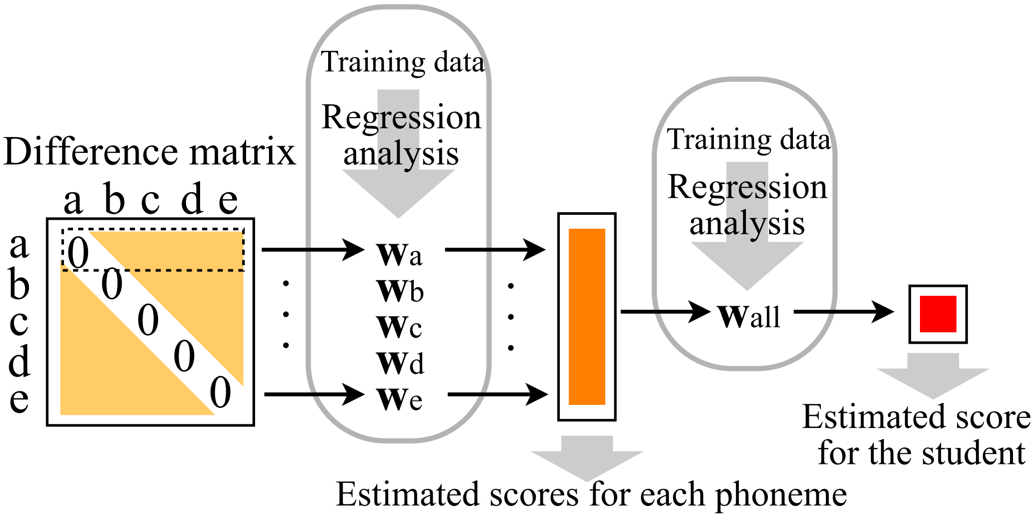


Fig.3.3: Two-layered regression analysis

regression is an estimation of each phoneme's score from each phoneme's distance vector, and the second one is an estimation of the overall score from the scores of phonemes. By using this two-layered regression, the number of parameters to estimate decreased from $\binom{N}{2}$ to $2N$, where N is the number of phonemes.

3.4 Isolated word recognition using speech structure

An algorithm of isolated word recognition using only speech structure was proposed[20]. The procedure is shown in Fig.3.4. Firstly, an HMM with N states is trained from an input utterance and a structure vector with its dimension of $\binom{N}{2}$ is calculated by the HMM. Then, because each utterance has a fixed-dimension feature vector, this is formulated as a simple pattern recognition problem. However, two major problems are found in this approach. One is called "a problem of too strong invariance". Because f-divergence is invariant for any invertible transformations, different words can have a similar f-divergences . The other is so-called "curse of dimensionality". Each utterance has one feature vector with its dimension of $\binom{N}{2}$, and it is difficult to estimate the distributions or patterns of such high dimensional feature with small number of training data.

First, the problem of too strong invariance and the solutions are introduced. As written in Section 3.2, because f-divergence is invariant for any invertible transformations and non-linguistic variations are often represented as some invertible transformations, it is concluded that f-divergence is invariant for non-linguistic variation. However, this logic only says that f-divergence is invariant for non-linguistic variation and doesn't say that f-divergence is invariant only for non-linguistic variation. f-divergence can be also invariant for linguistic information. In ASR, a feature which is invariant for non-linguistic variation

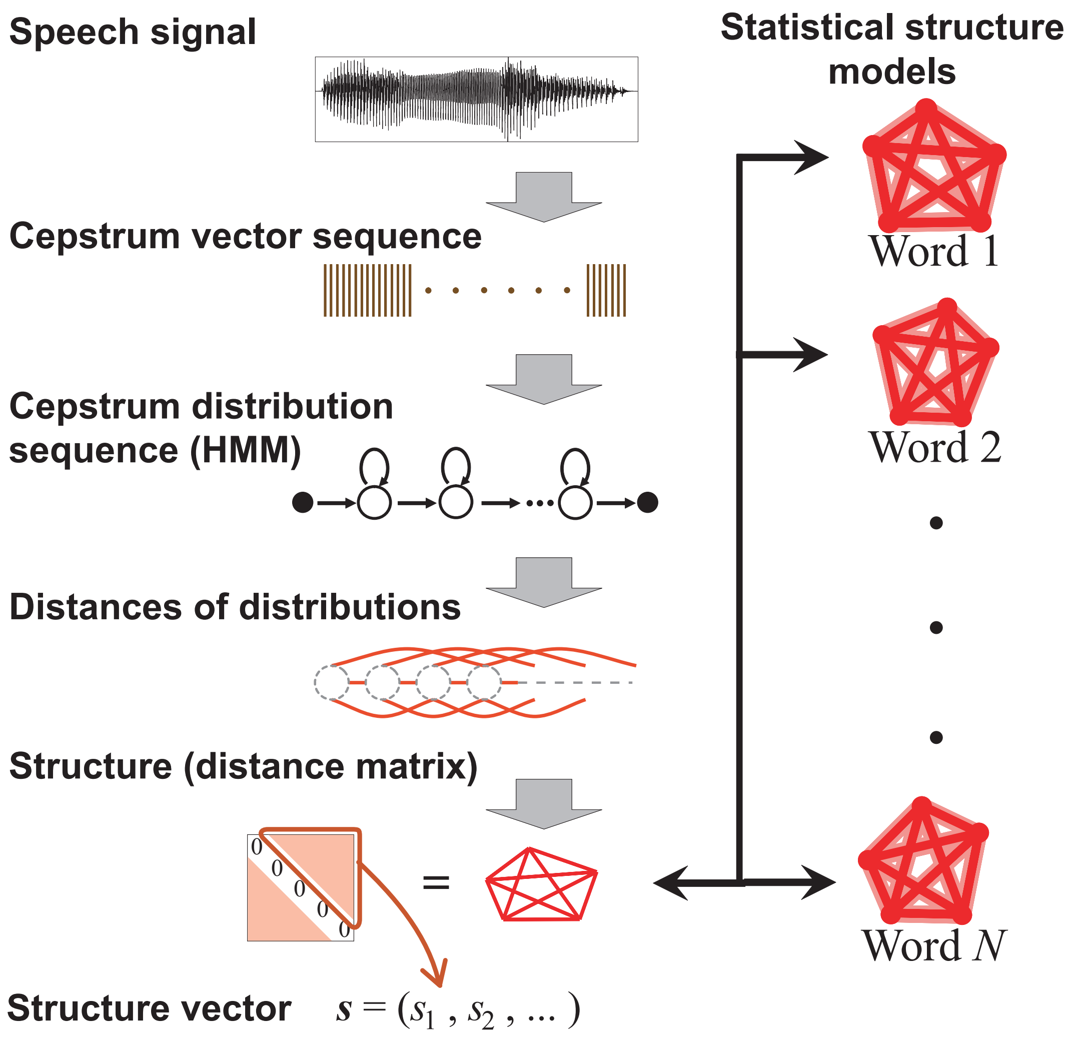

Speech signal
Cepstrum vector sequence
Cepstrum distribution sequence (HMM)
Distances of distributions
Structure (distance matrix)
0
0
0
0
0
=
Structure vector
$s = (s_1, s_2, \ldots)$
Statistical structure models
Word 1
Word 2
Word N

Fig.3.4: Isolated word recognition based on speech structure

and not invariant for linguistic information is desirable. To solve this problem, multi-stream structure was proposed[20]. The multi-stream structure assumes that the transformations caused by the difference of vocal tract length, microphone and stationary environment noise are approximated by a linear transformation by a band matrix. For example, in [32], the transformation used in Vocal Tract Length Normalization (VTLN) is shown to be written in a linear transformation A,

$$A = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \cdots \\ 0 & 1-\alpha^2 & 2\alpha-2\alpha^3 & \cdots & \cdots \\ 0 & -\alpha+\alpha^3 & 1-4\alpha^2+3\alpha^4 & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}, \tag{3.8}$$

where α is the parameter of frequency warping, which is estimated in VTLN. Furthermore, assuming $|\alpha| \ll 1$ and approximating it in first-order, A is approximated as

$$A = \begin{pmatrix} 1 & \alpha & 0 & 0 & \cdots \\ 0 & 1 & 2\alpha & 0 & \cdots \\ 0 & -\alpha & 1 & 3\alpha & \cdots \\ 0 & 0 & -2\alpha & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}. \tag{3.9}$$

In this approximation, A is a tridiagonal matrix. Considering them, it is appropriate to derive a feature which is invariant only for transformations by band matrix. The multi-stream structure was proposed as features which are invariant for band-matrix transformations and not invariant for full-matrix transformations. The concept is illustrated in Fig.3.5. In multi-stream structure, the feature space is divided into subspaces, each of which is composed of several successive dimensions. In Fig.3.5, each subspace has 3 successive dimensions, which means they are invariant for tridiagonal matrices like Eq.(3.9) and are not invariant for full matrices. The block size s is a parameter that determines the strength of invariance. When the block size s is equal to the dimension of cepstrum M, each subspace has only 1 dimension, which means they are invariant only for diagonal matrices, and the invariance is weakest. On the contrary, when the block size s is 1, it corresponds to the ordinary structure without multi-stream, and the invariance is strongest.

While multi-stream structure can control the strength of invariance and it provides better features for ASR, it increases the number of dimension by several times. Because the dimension of the structure vector is $\binom{N}{2}$ and much higher than that of ordinary cepstrum features, multi-stream structure makes the curse of dimensionality more serious. To solve the problem, several methods of dimensionality reduction have been proposed, such as random projection and Linear Discriminant Analysis (LDA) [21], Principal Component Analysis (PCA) and LDA[22], two-layered LDA[33], and parameter sharing[34]. As an example, the two-layered LDA is introduced. The basic procedure of structure-based isolated word recognition using two-layered LDA is shown in Fig.3.6. Firstly, a HMM with N states is trained from an utterance. Secondly, structure vectors are calculated for all the subspaces. Thirdly, a dimensionality reduction is applied to each stream using LDA. Finally, the dimensionality-reduced vectors in all the streams are concatenated and LDA is applied to the concatenated vector. Then, the dimensionality-reduced feature vector is obtained for each utterance.

In this section, the isolated word recognition using speech structure has been introduced. Recently, an algorithm of continuous speech recognition using speech structure was proposed[25]. In the next section, the algorithm is introduced.

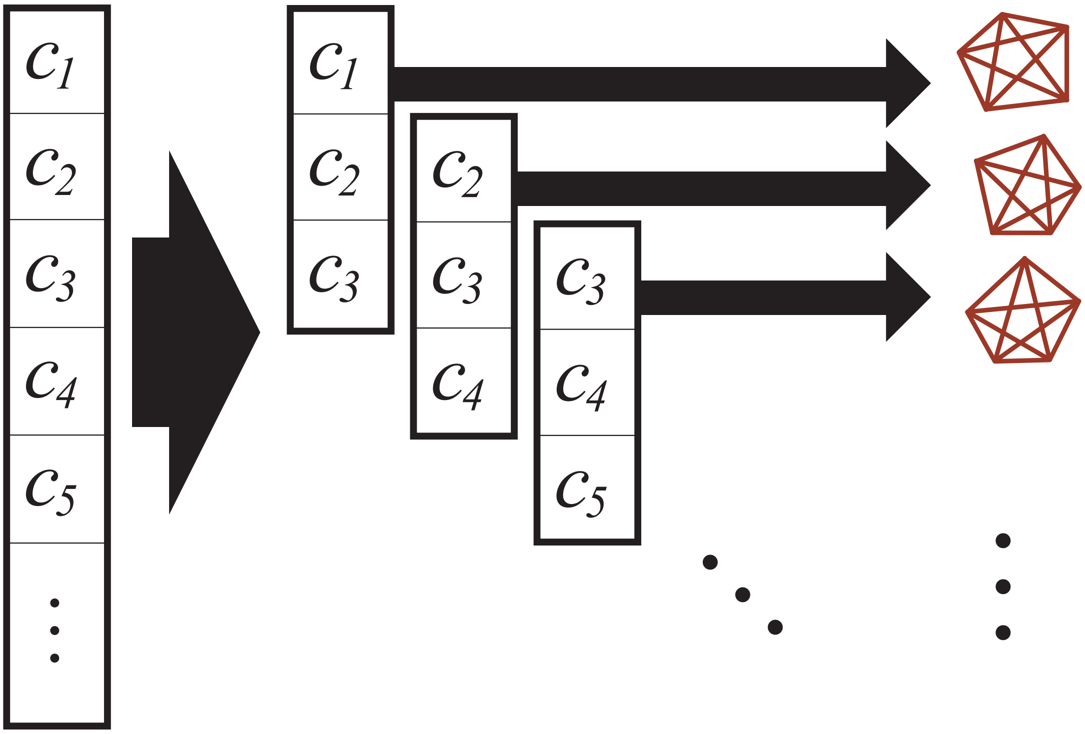


Fig.3.5: Multi-stream structure

3.5 Continuous speech recognition

An algorithm of continuous speech recognition using speech structure was proposed[25]. It rescores the result of HMM-based recognizer by using speech structure, which can be seen as a naive implementation of the new paradigm algorithms introduced in Section.2.5. In this section, the limitations of the previous methods to leverage speech structure for ASR and the solution for that are introduced along with [25].

There are two reasons why the previous studies on isolated word recognition using speech structure cannot be applied to continuous speech recognition. One is the number of distributions in an utterance. In the previous studies, a HMM with fixed number of states is trained from an utterance to construct speech structure. However, as long as a HMM with the fixed number of states is assumed, it can never be applied to continuous speech recognition, because the number and the kinds of words cannot be known in advance. The other one is the number of models. Generally speaking, it is practically impossible to train models for all the words. In most of continuous speech recognition tasks, because the number of words are too many to train the models without overfitting and the models are trained for each phonemes, the number of which is much smaller than that of words. However, in the previous studies on isolated word recognition using speech structure, the model is inevitably trained for each word because the speech structure is defined for each word.

To solve the problem, N-best re-ranking of the result generated by an HMM-based ASR system was proposed. The basic procedure is illustrated in Fig.3.7. While a HMM with the fixed number of states is trained from an utterance and the fixed number of distributions are obtained in the previous studies, variable number of distributions are obtained according to the candidates and its segmentations generated by HMM-based ASR in this approach. The procedure of extracting distributions and calculating the distances between them is shown in Fig.3.8. Firstly, N-best candidates of word sequence for an utterance are obtained by an HMM-based ASR system. Secondly, the segmentation of the utterance, which part of the utterance corresponds to which phoneme, is estimated for each candidate. Thirdly, the distribution of each phoneme is estimated according to the segmentation. Finally, the Bhattacharyya distances between phonemes are calculated. Then, the distances are obtained for each candidate, and it solves the problem of determining the number of distributions, because the number of distributions is determined for each candidate. In addition, statistical models are build for the each phoneme pair, while they are build for each word in the previous studies. The procedure of building and applying statistical edge models (SEMs) is shown in Fig.3.9. In this figure, an edge means a distance between two phonemes. In [25], using the edges obtained by the training data, Gaussian mixture model (GMM) of the edges are trained. Assume there are P phonemes, $\binom{P}{2}$ models are build. Once the models are build, the likelihood for each edge in each candidate can be calculated. The structural likelihood for the n-th hypothesis $h^{(n)}$, $L_{sr}(h^{(n)})$ is defined as

$$L_{sr}(h^{(n)}) = \frac{\sum_{\text{all}(i,j)} L_{ij}(e_{ij}^{(n)})}{K^{(n)}}, \tag{3.10}$$

where $L_{ij}(e_{ij}^{n})$ is a log likelihood of the edge obtained from the n-th hypothesis calculated by the corresponding SEM, and $K^{(n)}$ is the number of edges in the n-th hypothesis. Using the structural likelihood $L_{sr}(h^{n})$ and the HMM-based likelihood $L_{HMM}(h^{(n)})$, the score for re-ranking $L_{all}(h^{(n)})$ is defined as

$$L_{all}(h^{(n)}) = L_{HMM}(h^{(n)}) + w_{sr} L_{sr}(h^{(n)}), \tag{3.11}$$

where w_{sr} is a weight for the structural likelihood, which is determined experimentally.

3.6 Summary

In this chapter, the basics of speech structure and its applications have been introduced. Speech structure was proposed to solve the problem of non-linguistic variation. In speech structure, an utterance is converted into several distributions and the distances between the distributions are adopted as features that represent the utterance. Because f-divergence , which is proved to be invariant for any invertible and differentiable transformations, is used as the distance measure, speech structure is invariant for non-linguistic variations, which is often approximated by some invertible and differentiable transformations. Although two major problems, too strong invariance and curse of dimensionality, were found, several solutions, such as multi-stream structure and multi-layered dimensionality reduction, have been proposed and speech structure is successfully applied to pronunciation assessments and isolated word recognitions.

Furthermore, an algorithm of continuous speech recognition leveraging speech structure was proposed. It rescores the result of HMM-based recognizer by using speech structure, which can be seen as a naive implementation of the new paradigm algorithms introduced in Section.2.5. Because speech structure is not a frame-by-frame feature but a long-term feature, it cannot be used as a feature of old paradigm algorithm, it can be a effective long-term feature used in the new paradigm algorithms .

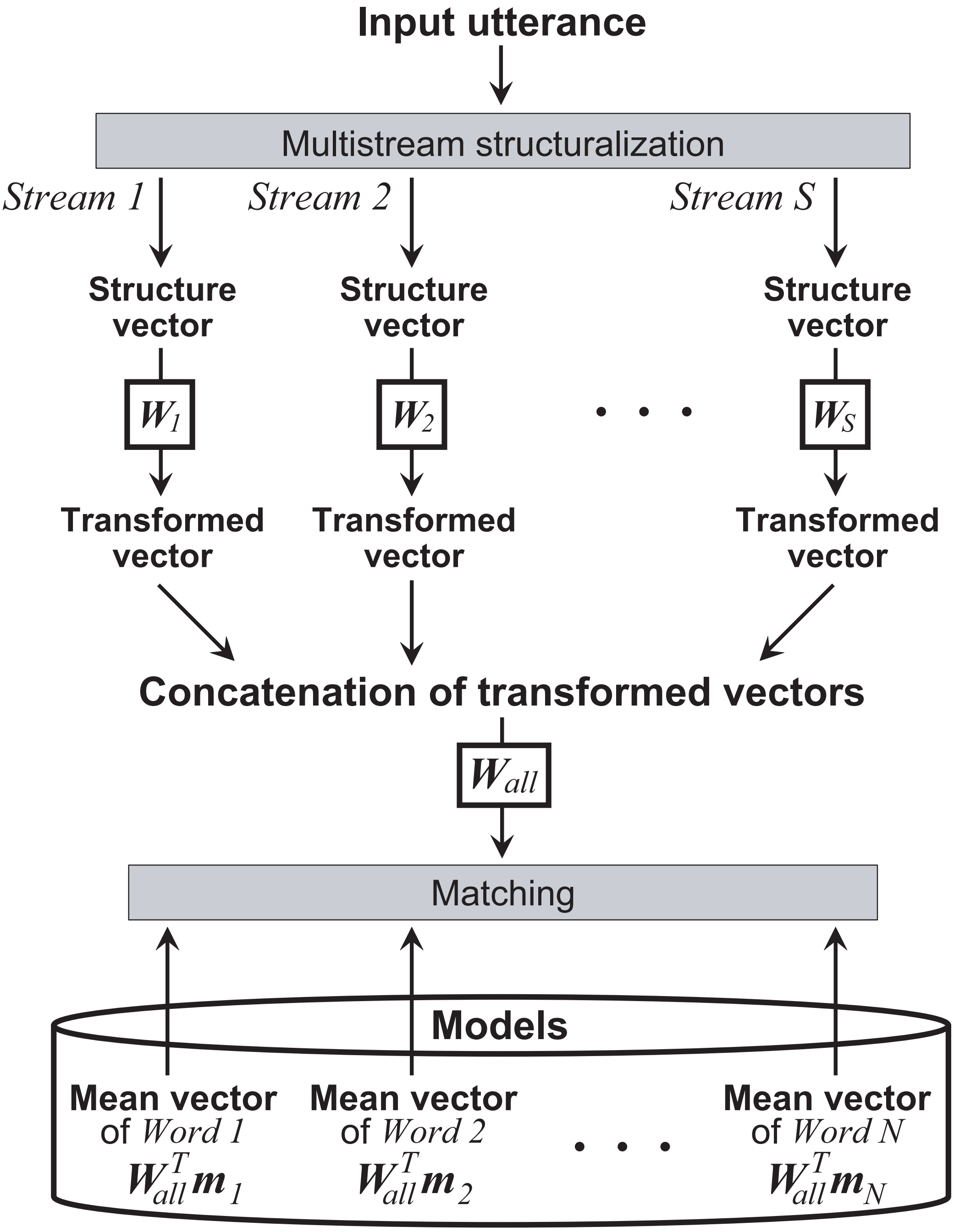


Fig.3.6: ASR using two-layered LDA

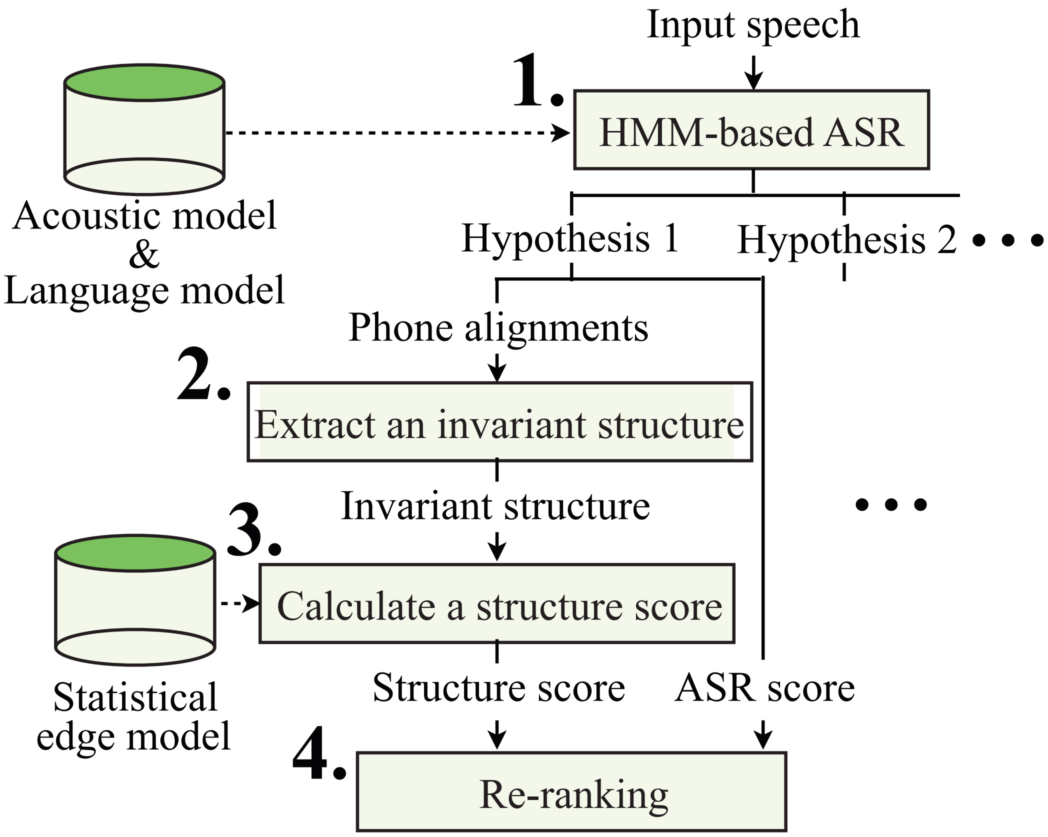


Fig.3.7: N-best re-ranking leveraging invariant structure

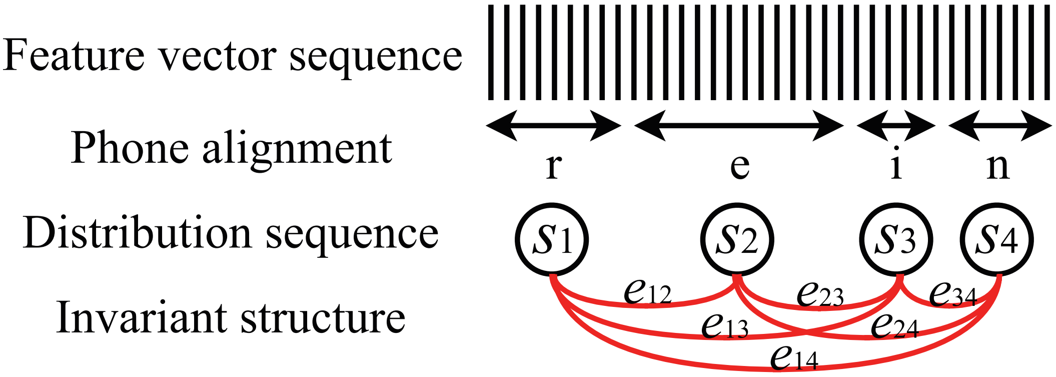


Fig.3.8: A procedure of extracting an invariant structure from a phone alignment.

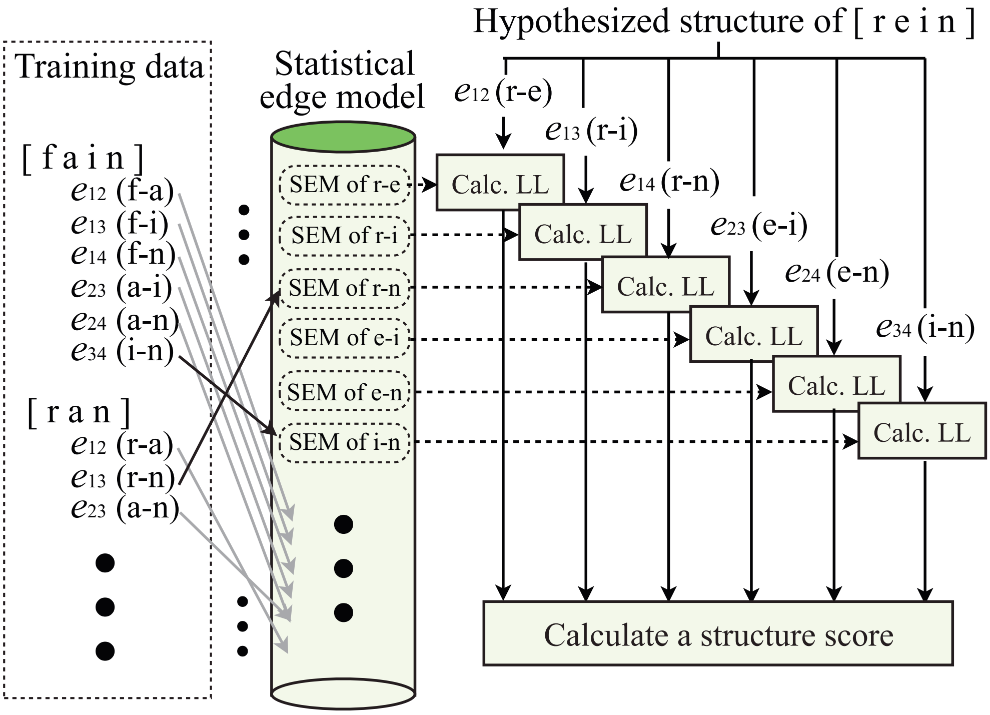


Fig.3.9: A procedure of building statistical edge models (SEMs) and calculating log likelihoods using SEMs. Log Likelihood is abbreviated as LL.

Chapter 4

Dynamic features derived from speech structure

4.1 Introduction

Dynamic features are defined as temporal derivatives of static features. They were firstly proposed in 1986[26], and are now effectively used in almost all the speech systems including ASR, speech synthesis, speaker identification, etc. However, dynamic features are reported to be ineffective in previous researches on speech structure[20]. In the previous researches, dynamic features of speech structure are defined as a speech structure constructed with dynamic features (Fig.4.1). It is true that the speech structure constructed with dynamic features are also invariant to transformations, it is slightly different from the original concept of dynamic features. Dynamic features are defined as temporal derivatives of static features. Because the distances between phonemes are adopted as features and the original cepstrum-based features are discarded in speech structure, dynamic features in speech structure should be temporal derivatives of the distances between phonemes instead of the distances between cepstrum-based dynamic features (Fig.4.2). As the implementations of the temporal derivatives of speech structure, two algorithms are proposed. One is differential speech structure. Differential speech structure is similar to the original concept in Fig.4.2. It directly defines the derivatives of distances between phonemes by assuming each phoneme distributions moves along with the mean of the derivative of its cepstrums. The other one is trajectory structure. Trajectory structure doesn't define the derivatives of distances directly. It firstly assumes each time frame has different distribution using trajectory HMM[35] while ordinary HMM assumes several states and the distribution in each state is constant in the state. Secondly, the distance vectors are obtained for each frame. Finally, the temporal derivatives are obtained by the sequence of the distance vectors like deriving dynamic features of cepstrums.

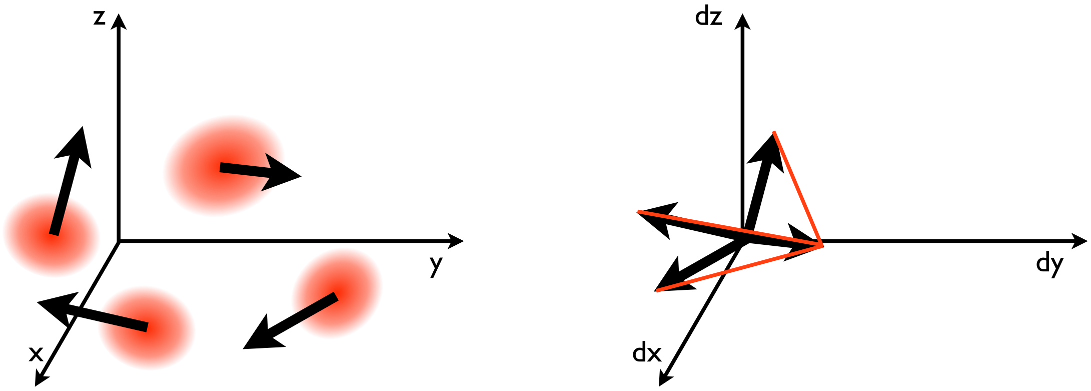


Fig.4.1: Speech structure constructed with delta features

4.2 Differential speech structure

In differential speech structure, the temporal derivatives of Bhattacharyya distances (BDs) are directly defined. Assume that an HMM with N states are trained from a utterance. By calculating BDs from every pair of the states, a sequence of distance vector $[\boldsymbol{s}_1, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_N]$ is obtained. $\boldsymbol{s}_n$ is a distance vector of the n-th state,

$$\boldsymbol{s}_n = \left[BD(P^{(n)}_{state}, P^{(1)}_{state}), BD(P^{(n)}_{state}, P^{(2)}_{state}), \cdots, BD(P^{(n)}_{state}, P^{(N)}_{state})\right]^{\mathrm{T}}. \tag{4.1}$$

Where $P^{(k)}_{state}$ is the output distribution of the k-th state. Assuming $P^{(n)}_{state}(\boldsymbol{c})$ is a Gaussian distribution $\mathcal{N}(\boldsymbol{c}|\boldsymbol{\mu}^{(n)}\Sigma^{(n)})$and the mean $\boldsymbol{\mu}^{(n)}$ temporally moves with keeping the variance $\Sigma^{(n)}$ constant, the temporal derivative of $\boldsymbol{s}_n$, $\Delta\boldsymbol{s}_n$ is defined as

$$\Delta\boldsymbol{s}_n = \frac{\partial\boldsymbol{s}_n}{\partial\boldsymbol{\mu}^{(n)}}\frac{d\boldsymbol{\mu}^{(n)}}{dt} \tag{4.2}$$

The k-th element of $\Delta\boldsymbol{s}_n$, $\Delta s^{(k)}_n$ is given by

$$\Delta s^{(k)}_n = \frac{\partial BD(P^{(n)}_{state}, P^{(k)}_{state})}{\partial\boldsymbol{\mu}^{(n)}}\frac{d\boldsymbol{\mu}^{(n)}}{dt}. \tag{4.3}$$

Because $BD(P^{(n)}_{state}, P^{(k)}_{state})$ is given by

$$BD(P^{(n)}_{state}, P^{(k)}_{state}) = \frac{1}{8}(\boldsymbol{\mu}^{(n)} - \boldsymbol{\mu}^{(k)})^T\Sigma^{-1}(\boldsymbol{\mu}^{(n)} - \boldsymbol{\mu}^{(k)}) + \frac{1}{2}\log(\frac{|\Sigma|}{|\Sigma^{(n)}|^{1/2}|\Sigma^{(k)}|^{1/2}}), \tag{4.4}$$

where $\Sigma = \frac{\Sigma^{(n)}+\Sigma^{(k)}}{2}$, $\Delta s^{(k)}_n$ is obtained as

$$\Delta s^{(k)}_n = \frac{1}{4}\Sigma^{-1}(\boldsymbol{\mu}^{(n)} - \boldsymbol{\mu}^{(k)}) \cdot \frac{d\boldsymbol{\mu}^{(n)}}{dt}. \tag{4.5}$$

In addition, $\frac{d\boldsymbol{\mu}^{(n)}}{dt}$ is approximated by the mean vector of delta cepstrums which belong to the n-th distribution as

$$\frac{d\boldsymbol{\mu}^{(n)}}{dt} \approx \frac{1}{L_n} \sum_{\text{all } t \text{ in } n\text{-th state}} \Delta \boldsymbol{c}_t, \tag{4.6}$$

where L_n is the number of frames that belong to n-th state. Then, $\Delta \boldsymbol{s}_n$ is defined for each $\boldsymbol{s}_n$, and they can be used as dynamic features derived from speech structure.

4.3 Trajectory speech structure

While in the conventional speech structure we trained an HMM with N states and obtained $\binom{N}{2}$ BDs between each pair of the N state distributions, in the trajectory structure model, we assume that each time frame has its own unique distribution. In other words, we can have a coarsely quantized distribution sequence from a classical HMM and a finely quantized sequence from a trajectory HMM. Using these two distribution sequences, we can calculate a BD between a fine distribution and a coarse distribution. Fig.4.3 shows an example of feature sequence, its speech structure, and its trajectory structure. In the standard approach, a feature sequence $[\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T]$ itself is often used as a representation of the input utterance (Fig.4.3(a)). In the conventional speech structure, we train an HMM with N states, calculate BDs from every pair of the states, and obtain a sequence of distance vector $[\boldsymbol{s}_1, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_N]$. $\boldsymbol{s}_n$ is a distance vector of the n-th state,

$$\boldsymbol{s}_n = \left[BD(P^{(n)}_{state}, P^{(1)}_{state}), BD(P^{(n)}_{state}, P^{(2)}_{state}), \cdots, BD(P^{(n)}_{state}, P^{(N)}_{state})\right]^{\mathrm{T}}. \tag{4.7}$$

Where $P^{(n)}_{state}$ is the output distribution of the n-th state. Because Bhattacharyya distance is symmetric, we just pick up $\binom{N}{2}$ distances out of the N^2 distances. These $\binom{N}{2}$ distances are used as a representation of the input utterance (Fig.4.3(b)).

In trajectory structure model, after we train a classical HMM with N states, it is used to derive frame-dependent distributions. Then, we obtain a sequence of distributions $\left[P^{(1)}_{frame}(\boldsymbol{x}), P^{(2)}_{frame}(\boldsymbol{x}), \cdots, P^{(T)}_{frame}(\boldsymbol{x})\right]$, where T is the total number of frames and $P^{(t)}_{frame}(\boldsymbol{x})$ is the distribution of the t-th frame. Each $P^{(t)}_{frame}(\boldsymbol{x})$ is calculated from trajectory HMM [35], which derives the temporally changing distributions of static features by imposing the explicit relationship between static features and dynamic features. The detailed procedure to obtain $P^{(t)}_{frame}(\boldsymbol{x})$ is described later. By using T fine distributions and N coarse distributions, at time t, we calculate a distance vector whose i-th element is BD between the t-th fine distribution and the i-th coarse distribution. Since this vector can be obtained at each time, we have T distance vectors with their dimension being N. (Fig.4.3(c)). The distance vector at time t, $\boldsymbol{d}_t$, is given by

$$\boldsymbol{d}_t = \left[BD(P^{(t)}_{frame}, P^{(1)}_{state}), BD(P^{(t)}_{frame}, P^{(2)}_{state}), \cdots, BD(P^{(t)}_{frame}, P^{(N)}_{state})\right]^{\mathrm{T}}. \tag{4.8}$$

Here, a sequence of the distance vectors $[\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_T]$ is used as a representation of the input utterance. In addition, since the distance vector is obtained at each time, we can calculate its Δ and Δ^2 features and concatenate them and $\boldsymbol{d}_t$.

So far, we have introduced the basic procedure to derive TSR model. In the rest of this section, we introduce the detailed procedure to derive $P^{(t)}_{frame}(\boldsymbol{x})$ based on trajectory HMM. Let $\boldsymbol{x}$ denote a concatenation of an input feature sequence $\left[\boldsymbol{x}_1^{\mathrm{T}}, \boldsymbol{x}_2^{\mathrm{T}}, \cdots, \boldsymbol{x}_T^{\mathrm{T}}\right]^{\mathrm{T}}$, each $\boldsymbol{x}_t$ is a concatenation of a static feature vector and its Δ and Δ^2 features, $\left[\boldsymbol{c}_t^{\mathrm{T}}, \Delta\boldsymbol{c}_t^{\mathrm{T}}, \Delta\boldsymbol{c}_t^{\mathrm{T}}\right]^{\mathrm{T}}$. Let M denote the dimension of $\boldsymbol{c}_t$. Δ and Δ^2 features are defined as weighted sums of adjacent static feature vectors as

$$\Delta\boldsymbol{c}_t = \sum_{\tau=-L}^{L} w^{(1)}(\tau)\boldsymbol{c}_{t+\tau}, \tag{4.9}$$

$$\Delta^2\boldsymbol{c}_t = \sum_{\tau=-L}^{L} w^{(2)}(\tau)\boldsymbol{c}_{t+\tau}, \tag{4.10}$$

where L is the length of window to calculate dynamic features, and $w^{(1)}(\tau)$ and $w^{(2)}(\tau)$ are coefficients for $\boldsymbol{c}_{t+\tau}$. Because each $\boldsymbol{x}_t$ is calculated by an linear transformation of $\boldsymbol{c}_t$, there exists a matrix W such that

$$\boldsymbol{x} = W\boldsymbol{c}, \tag{4.11}$$

where $\boldsymbol{c}$ is a concatenation of a static feature sequence $\left[\boldsymbol{c}_1^{\mathrm{T}}, \boldsymbol{c}_2^{\mathrm{T}}, \cdots, \boldsymbol{c}_T^{\mathrm{T}}\right]^{\mathrm{T}}$. Because the dimension of $\boldsymbol{c}$ is MT and that of $\boldsymbol{x}$ is $3MT$, W is a $3MT \times MT$ matrix. Using the coefficients $w^{(1)}(\tau)$ and $w^{(2)}(\tau)$, W is given as

$$\boldsymbol{W} = \left[\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_T\right]^{\mathrm{T}}, \tag{4.12}$$

$$\boldsymbol{w}_t = \left[\boldsymbol{w}_t^{(0)}, \boldsymbol{w}_t^{(1)}, \boldsymbol{w}_t^{(2)}\right], \tag{4.13}$$

$$\boldsymbol{w}_t^{(n)} = \left[w^{(n)}(1-t)\boldsymbol{I}_{M\times M}, w^{(n)}(2-t)\boldsymbol{I}_{M\times M}, \cdots, w^{(n)}(T-t)\boldsymbol{I}_{M\times M},\right]^{\mathrm{T}}. \tag{4.14}$$

Note that $\boldsymbol{w}_t^{(n)}$ is actually much sparser than it is seen in Eq.(4.14), because the delta coefficient $w^{(n)}(\tau)$ is non-zero only if $-L \leq \tau \leq +L$ and L is usually much smaller than T. When we assume this feature sequence is generated by an HMM that has N states each of which has a single Gaussian as output distribution, the probability of $\boldsymbol{x}$ given alignment $\boldsymbol{q}$ and HMM parameter λ, $P(\boldsymbol{x}|\boldsymbol{q}, \lambda)$ is calculated as follows.

$$P(\boldsymbol{x}|\ \boldsymbol{q}, \lambda) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t|\ \boldsymbol{\mu}_{q_t}, \Sigma_{q_t}) \tag{4.15}$$

$$= \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{\boldsymbol{q}}, \Sigma_{\boldsymbol{q}}) \tag{4.16}$$

where q_t is the index of the state to which $\boldsymbol{c}_t$ belongs. $\boldsymbol{\mu}_{\boldsymbol{q}}$ and $\Sigma_{\boldsymbol{q}}$ are concatenations of sequences $\boldsymbol{\mu}_{q_t}$ and Σ_{q_t} respectively.

$$\boldsymbol{\mu}_{\boldsymbol{q}} = \left[\boldsymbol{\mu}_{q_1}^{\mathrm{T}}, \boldsymbol{\mu}_{q_2}^{\mathrm{T}}, \cdots, \boldsymbol{\mu}_{q_T}^{\mathrm{T}}\right]^{\mathrm{T}} \tag{4.17}$$

$$\Sigma_{\boldsymbol{q}} = \mathrm{diag}\left[\Sigma_{q_1}, \Sigma_{q_2}, \cdots, \Sigma_{q_T}\right] \tag{4.18}$$

Because $\boldsymbol{x}$ satisfies Eq.4.11, there exist a mean vector $\bar{\boldsymbol{c}}_{\boldsymbol{q}}$ and a covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$ such that

$$\begin{aligned} P(\boldsymbol{x}|\, \boldsymbol{q}, \lambda) &= \mathcal{N}(\boldsymbol{W}\boldsymbol{c}|\, \boldsymbol{\mu}_{\boldsymbol{q}}, \Sigma_{\boldsymbol{q}}) && (4.19)\\ &= K_{\boldsymbol{q}}\, \mathcal{N}(\boldsymbol{c}|\, \bar{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}) && (4.20) \end{aligned}$$

where,

$$\begin{aligned} \boldsymbol{R}_{\boldsymbol{q}}\bar{\boldsymbol{c}}_{\boldsymbol{q}} &= \boldsymbol{r}_{\boldsymbol{q}}, && (4.21)\\ \boldsymbol{R}_{\boldsymbol{q}} &= \boldsymbol{W}^{\mathrm{T}}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{W} = \boldsymbol{P}_{\boldsymbol{q}}^{-1}, && (4.22)\\ \boldsymbol{r}_{\boldsymbol{q}} &= \boldsymbol{W}^{\mathrm{T}}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}}, && (4.23)\\ K_{\boldsymbol{q}} &= \frac{\sqrt{(2\pi)^{MT}|\boldsymbol{P}_{\boldsymbol{q}}|}}{\sqrt{(2\pi)^{3MT}|\boldsymbol{U}_{\boldsymbol{q}}|}} \exp\{-\frac{1}{2}\left(\boldsymbol{\mu}_{\boldsymbol{q}}^{\mathrm{T}}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}} - \boldsymbol{r}_{\boldsymbol{q}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{q}}^{-1}\boldsymbol{r}_{\boldsymbol{q}}\right)\}. && (4.24) \end{aligned}$$

An actual example of $\boldsymbol{c}$, $\boldsymbol{\mu}_{\boldsymbol{q}}$, and $\bar{\boldsymbol{c}}_{\boldsymbol{q}}$ is shown in Fig.4.4. The covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$ is not a diagonal matrix, but still a band matrix, in which only diagonal elements and their adjacent elements are non-zero. To obtain an independent distribution for each frame, we approximate $\boldsymbol{P}_{\boldsymbol{q}}$ as a block diagonal matrix as follows

$$\boldsymbol{P}_{\boldsymbol{q}} \approx \mathrm{diag}\left[\boldsymbol{p}_{\boldsymbol{q}}^{(1)}, \boldsymbol{p}_{\boldsymbol{q}}^{(2)}, \cdots, \boldsymbol{p}_{\boldsymbol{q}}^{(T)}\right]. \tag{4.25}$$

Finally, we obtain an independent distribution for each frame.

$$\mathcal{N}(\boldsymbol{c}|\, \bar{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}) \approx \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{c}_t|\, \bar{\boldsymbol{c}}_{\boldsymbol{q}}^{(t)}, \boldsymbol{p}_{\boldsymbol{q}}^{(t)}), \tag{4.26}$$

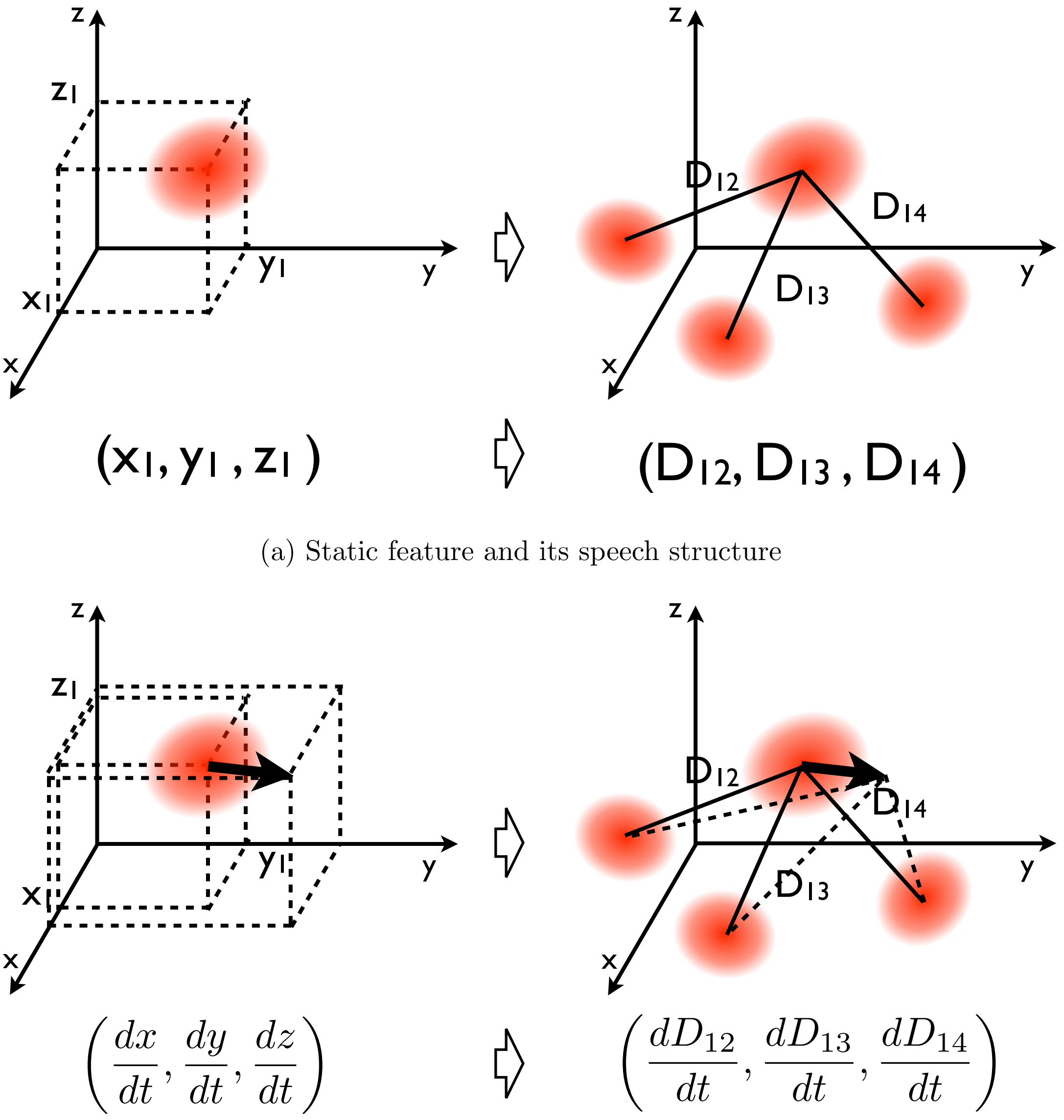
where $\bar{\boldsymbol{c}}_{\boldsymbol{q}}^{(t)}$ is a vector with its dimension of M that corresponds to the t-th time frame. From the above equation, we can define frame-dependent distribution $P_{frame}^{(t)}(\boldsymbol{c}_t)$,

$$P_{frame}^{(t)}(\boldsymbol{c}_t) = \mathcal{N}(\boldsymbol{c}_t|\, \bar{\boldsymbol{c}}_{\boldsymbol{q}}^{(t)}, \boldsymbol{p}_{\boldsymbol{q}}^{(t)}). \tag{4.27}$$

Then, the distance vector at time t, $\boldsymbol{d}_t$

$$\boldsymbol{d}_t = \left[BD(P_{frame}^{(t)}, P_{state}^{(1)}), BD(P_{frame}^{(t)}, P_{state}^{(2)}), \cdots, BD(P_{frame}^{(t)}, P_{state}^{(N)})\right]^{\mathrm{T}}, \tag{4.28}$$

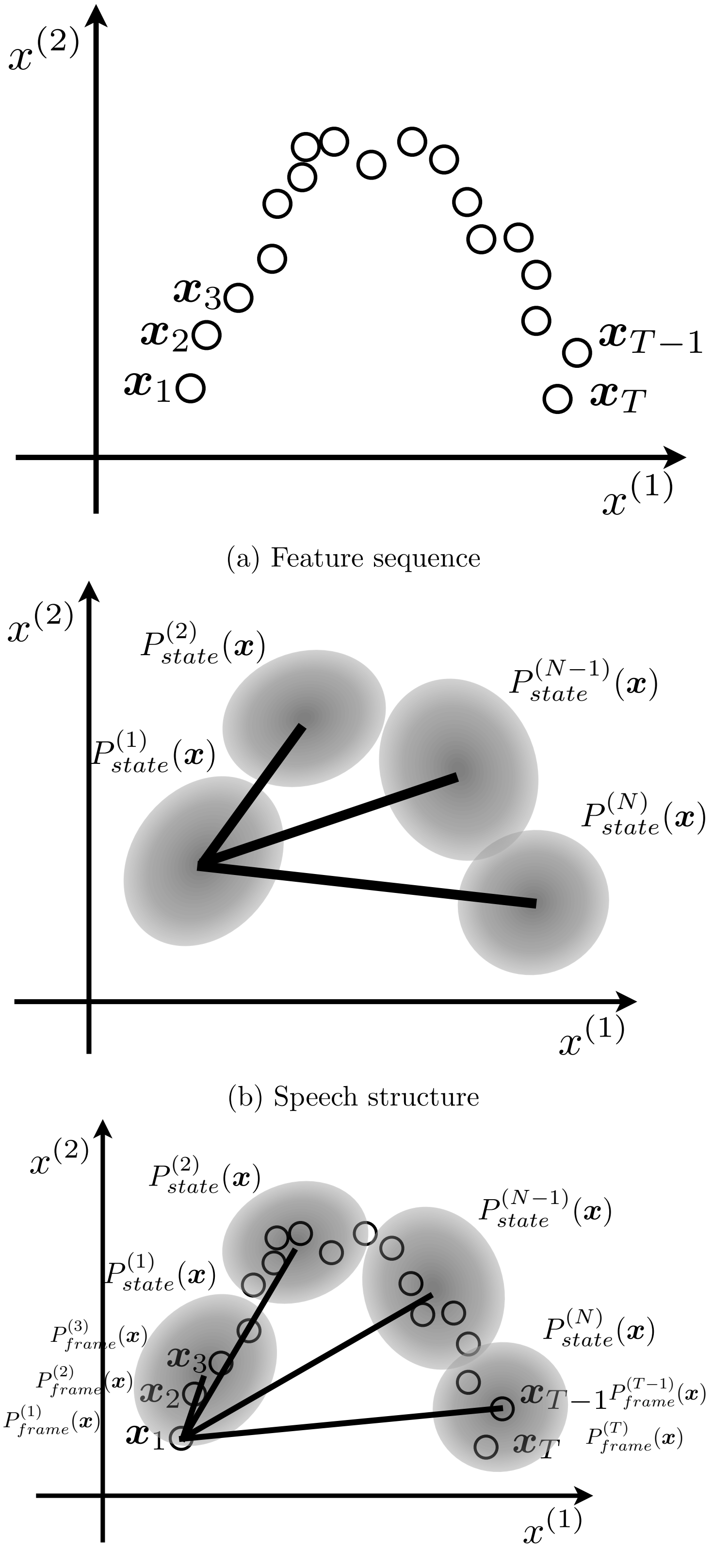
is obtained. From the sequence of distance vectors, $\Delta\boldsymbol{d}_t$ and $\Delta^2\boldsymbol{d}_t$ are calculated like ordinary delta cepstrums.


z
z1
x1
y1
y
x
D12
D13
D14
(x_1, y_1, z_1)
(D_{12}, D_{13}, D_{14})
$\left(\frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt}\right)$
$\left(\frac{dD_{12}}{dt}, \frac{dD_{13}}{dt}, \frac{dD_{14}}{dt}\right)$

(a) Static feature and its speech structure

(b) Dynamic feature and its dynamic speech structure

Fig.4.2: The basic concept of proposal, the temporal derivatives of speech structure


$x^{(2)}$
$\boldsymbol{x}_3$
$\boldsymbol{x}_2$
$\boldsymbol{x}_1$
$\boldsymbol{x}_{T-1}$
$\boldsymbol{x}_T$
$x^{(1)}$
$x^{(2)}$
$P_{state}^{(2)}(\boldsymbol{x})$
$P_{state}^{(N-1)}(\boldsymbol{x})$
$P_{state}^{(1)}(\boldsymbol{x})$
$P_{state}^{(N)}(\boldsymbol{x})$
$x^{(1)}$
$x^{(2)}$
$P_{state}^{(2)}(\boldsymbol{x})$
$P_{state}^{(N-1)}(\boldsymbol{x})$
$P_{state}^{(1)}(\boldsymbol{x})$
$P_{frame}^{(3)}(\boldsymbol{x})$
$P_{state}^{(N)}(\boldsymbol{x})$
$\boldsymbol{x}_3$
$P_{frame}^{(2)}(\boldsymbol{x})$
$\boldsymbol{x}_2$
$\boldsymbol{x}_{T-1}$
$P_{frame}^{(T-1)}(\boldsymbol{x})$
$P_{frame}^{(1)}(\boldsymbol{x})$
$\boldsymbol{x}_1$
$\boldsymbol{x}_T$
$P_{frame}^{(T)}(\boldsymbol{x})$
$x^{(1)}$

(a) Feature sequence

(b) Speech structure

(c) Trajectory structure

Fig.4.3: A feature sequence, its speech structure, and its trajectory structure

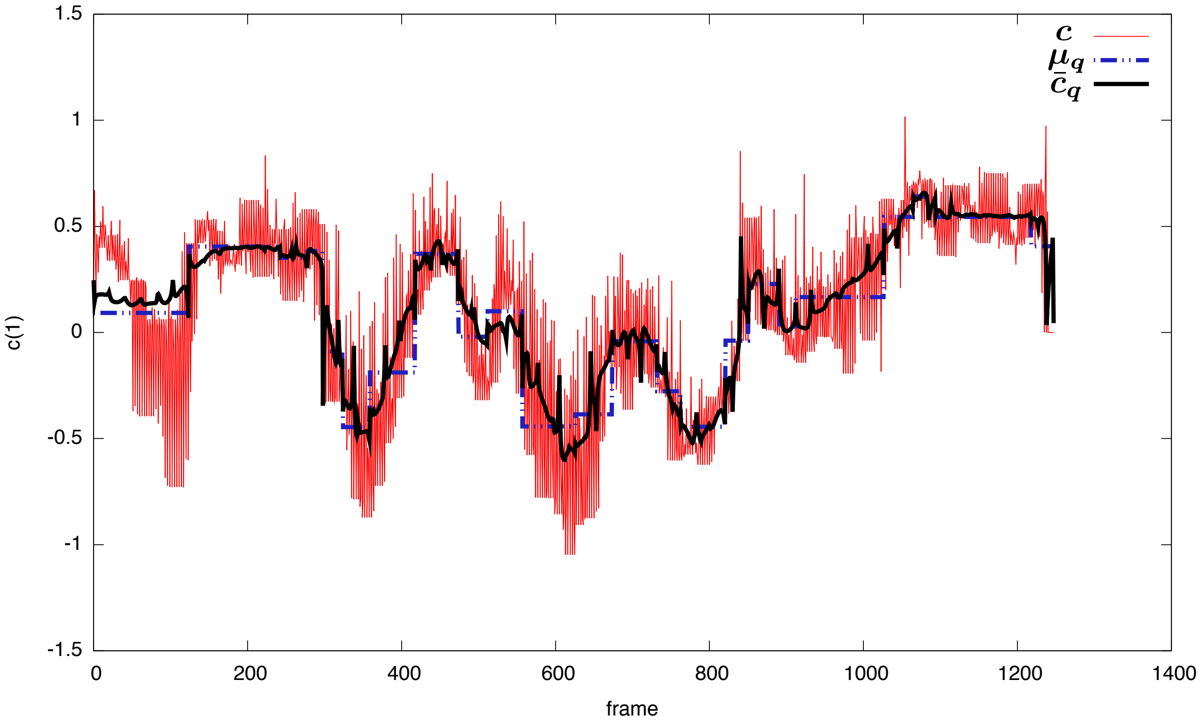


Fig.4.4: Observations of 1st-order mel-cepstrum, means of the original HMM, and the mean of the trajectory HMM

Chapter 5

Experiment

5.1 Pronunciation assessment based on differential structure

We carried out an experiment of pronunciation assessment based on differential structure. The basic procedure is same as the one introduced in Section 3.3 and Fig.3.2. To compare the pronunciation of a student and that of a teacher, the mel-cepstrum distributions of each phoneme is estimated for the student and the teacher. Because each distribution is estimated from a small number of samples, it is difficult to estimate the distribution correctly. To solve the problem, maximum a posteriori (MAP) estimation, which estimates the distribution with a prior distribution, is applied to estimate the distribution for each phoneme. It is shown that MAP estimation works effectively to estimate the distributions for speech structure in previous study[36]. The phoneme independent prior distribution is calculated by empirical bayes using all the training data. While the score for each speaker is estimated in Section 3.3, the score for each phoneme pronounced by each speaker is estimated here. Let K denote the number of phonemes, $p_k^{(S)}$ and $p_k^{(T)}$ denote the distribution of k-th phoneme pronounced by the teacher and the student respectively. The feature vector of k-th phoneme pronounced by the student, $\boldsymbol{d}_k = \{d_k^{(1)}, d_k^{(2)}, \cdots, d_k^{(K)}\}$ is defined using speech structure. The l-th dimension of $\boldsymbol{d}_k$, $d_k^{(l)}$ is given by

$$d_k^{(l)} = \left(\frac{s_k^{(l)} - t_k^{(l)}}{s_k^{(l)} + t_k^{(l)}} \right), \tag{5.1}$$

where

$$\begin{aligned} s_k^{(l)} &= BD(p_k^{(S)}, p_l^{(S)}), && (5.2) \\ t_k^{(l)} &= BD(p_k^{(T)}, p_l^{(T)}). && (5.3) \end{aligned}$$

In addition, differential speech structures, $\Delta s_k^{(l)}$ and $\Delta t_k^{(l)}$, are also calculated by the algorithm introduced in Section 4.2. Using these features, the score for each phoneme is estimated by using a regression. We adopt ridge regression as a regression algorithm. Ridge

Table5.1: Conditions

Sampling	16bit/16kHz
Window	25ms Blackman Window & 1ms Shift
Feature	mel-cepstrum 10 dim.
Distribution	Gaussian distribution with diagonal covariance matrix
The number of distributions	23
Estimation of distributions	MAP estimation

regression is a linear regression with Tikhonov regularization. Ordinary linear regression estimates the regression parameter $\hat{\boldsymbol{\theta}}$ by minimizing the error as

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_i \left(y_i - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_i\right)^{\mathrm{T}} \left(y_i - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_i\right), \tag{5.4}$$

where $\boldsymbol{x}_i$ is the feature vector and y_i is the corresponding objective variable. On the other hand, ridge regression estimates $\hat{\boldsymbol{\theta}}$ by minimizing the error and the L2 regularization factor as

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_i \left(y_i - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_i\right)^{\mathrm{T}} \left(y_i - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_i\right) + \alpha \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\theta}, \tag{5.5}$$

where α is a hyperparameter which is determined experimentally. By adding the regularization factor, the overfitting is suppressed and the generalization ability is improved. In this experiment, the feature vector $\boldsymbol{x}_i$ corresponds to $\boldsymbol{d}_k$ or $\left[\boldsymbol{d}_k^{\mathrm{T}}, \Delta\boldsymbol{d}_k^{\mathrm{T}}\right]^{\mathrm{T}}$, and the objective variable y_i corresponds to the score of the phoneme.

5.1.1 Experimental conditions

In this experiment, English Read by Japanese (ERJ) database[37] is used. We used 10 vowels, including 5 diphthongs and 5 monophthongs, shown in Table.??. The number of speakers is 45. The proficiency score is given for each phoneme of each speaker. The score is four-level rate (1 to 4), in which 1 means the worst and 4 means the best, labeled by one phonetician. The correlation between the estimated score and the manually labeled score is adopted as a performance measure. Using 45 speakers, leave-one-out cross-validation (44 speakers for training and 1 speakers for evaluation) were carried out. We compared three features, 1. static structure vector, 2. static structure vector and structure of delta (previous method), and 3. static structure vector and differential structure vector (proposal). The structure of delta means the distances between the distributions of delta cepstrums (Fig.4.1). The hyperparameter α in ridge regression (Eq.(5.5)) is set to $\alpha = 1.0$ from preliminary experiments. Other conditions are shown in Table 5.1.

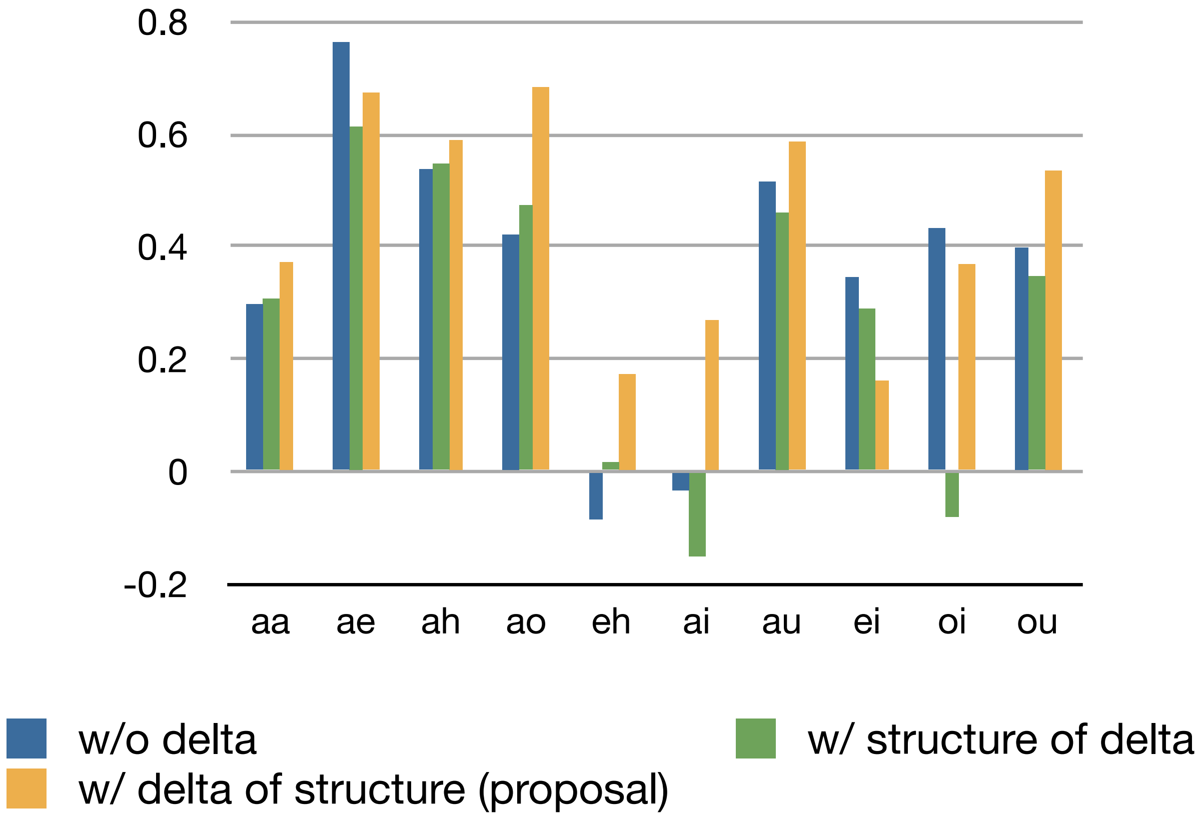


Fig.5.1: The proposed method achieves better correlations in average than other two methods

5.1.2 Results

The results are shown in Fig.5.1. The correlation between the estimated score and the manually labeled score is shown for each phoneme. As shown in the figure, the proposed method shows better correlations in average. Especially, the correlations of several phonemes that is almost zero or minus in other two methods are improved significantly. The results show that the proposed method works effectively in pronunciation assessment.

Table5.2: The list of words

Diphthongs	kite(/aɪ/)	how(/aʊ/)	sage(/eɪ/)
		oils(/ɔɪ/)	float(/oʊ/)
Monophthongs	hot(/ɑ/)	hatch(/æ/)	touch(/ʌ/)
		raw(/ɔ/)	test(/ɛ/)

Table5.3: Conditions

Sampling	16bit/16kHz
Window	25ms Blackman Window & 1ms Shift
Feature	mel-cepstrum 12 dim.
HMM	Gaussian with diagonal matrix, 25 states and 23 distributions
Estimation of distributions	MAP estimation

5.2 Isolated word recognition based on differential structure

In the previous section, it is shown that differential speech structure works effectively in pronunciation assessment. In addition, an experiment of isolated word recognition using differential speech structure was carried out. The basic procedure is same as the one introduced in Section 3.4 and Fig. 3.4. Firstly, an HMM with N states is trained from an input utterance and a structure vector with its dimension of $\binom{N}{2}$ is calculated by the HMM. Then, because each utterance has a fixed-dimension feature vector, they are solved as a simple pattern recognition problem. In this experiment, differential structure vector, which dimension is also $\binom{N}{2}$ is concatenated to the static structure vector. The output probability of a structure vector $\boldsymbol{s}$ given a word w, $P(\boldsymbol{s}|w)$, is modeled by a Gaussian with diagonal covariance matrix. In addition, multi-stream structure introduced in Section 3.4 is applied. In multi-stream structure, the cepstrum space is divided into several subspaces, each of which has several successive dimensions, and f-divergence is calculated for each subspace. By increasing the number of subspaces, the number of f-divergences increases and the invariance gets weaker.

5.2.1 Experimental conditions

We used Tohoku University and Panasonic isolated spoken word database [38], which contains 212 kinds of Japanese words spoken by 60 speakers. The word length varies from 3 morae to 7 morae. We used the utterances by 30 speakers as a training data set and ones by the other 30 speakers as an evaluation data set. Other conditions are shown in Table 5.3.

5.2.2 Results

The results are shown in Fig.5.2. The x-axis is the number of streams. Because 12 dimensional cepstrum is used as a baseline feature, the max value of the number of streams is 12. As shown in Fig.5.2, proposed method shows better results than conventional one in any number of stream. In both conventional and proposal, the best result is obtained

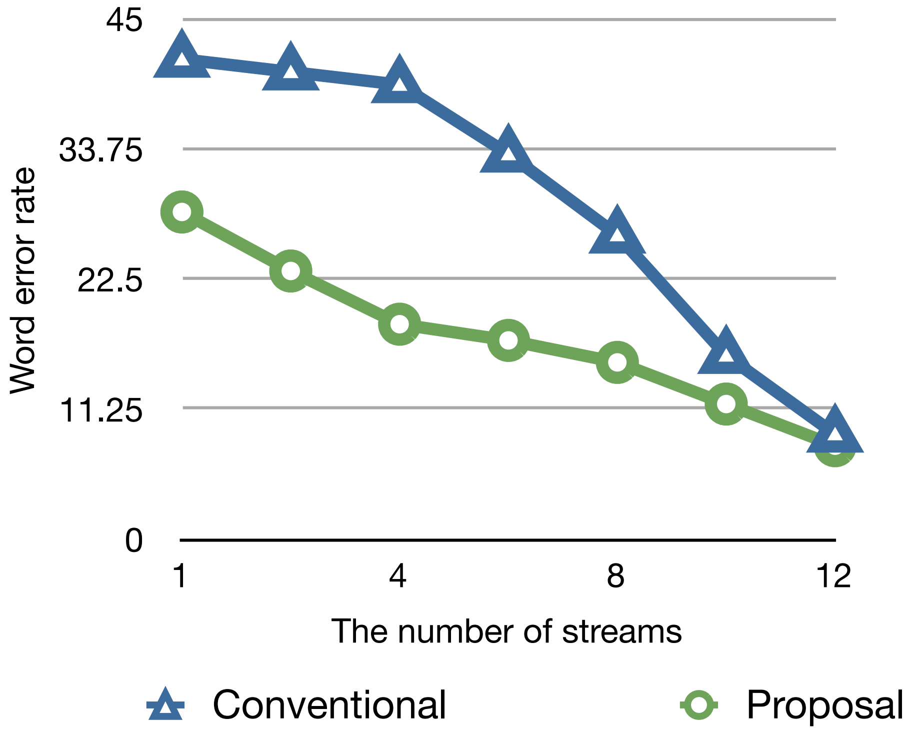


Fig.5.2: Word error rates obtained by static structure (conventional) and by static feature concatenated with differential structure (proposal)

when the number of streams is 12. When the number of stream is 12, the word error rates of proposal and conventional are 8.1% and 9.1%, which means 11.0% relative decrease is obtained by proposed method. In addition, because the difference between conventional and proposal is bigger when the number of streams is smaller, it can be thought that the differential speech structure can ease the problem of too strong invariance like multi-stream structure. These results show that the differential structure works effectively in isolated word recognition.

5.3 N-best rescoring based on trajectory structure

N-best rescoring based on speech structure is proposed in [25]. We carried out an experiment of N-best rescoring based on trajectory structure (TSR) models. The task is isolated word recognition. Let $L_{hmm}(\boldsymbol{x}|w_i)$ denote the log output probability of $\boldsymbol{x}$ given word w_i's HMM, and $L_{tsr}(\boldsymbol{c}|w_i)$ denote that of $\boldsymbol{c}$ given word w_i's TSR model. The rescored log likelihood $L_{all}(\boldsymbol{x}|\ w_i)$ is given by:

$$L_{all}(\boldsymbol{x}|w_i) = L_{all}(\boldsymbol{x}|w_i) + w_{tsr}L_{tsr}(\boldsymbol{c}|w_i) \tag{5.6}$$

where w_{tsr} is the weight of TSR model. The procedure to calculate the log TSR likelihood $L_{tsr}(\boldsymbol{c}|w_i)$ is shown in Fig. 5.3. To calculate $L_{tsr}(\boldsymbol{c}|w_i)$, a classical HMM trained for the input utterance is needed. For that, we first trained a speaker-independent HMM for each word, which is used as initial model. The parameters of this initial HMM are updated only by the input utterance and the resulting HMM is used to derive the TSR model. If we do not use speaker-independent word HMMs as initial and background models, the resulting HMM of an utterance and that of another utterance will show different alignment patterns between states and feature vectors even when the two utterances are of the same word. Because the feature vector in speech structures is composed of distances between HMM states, it is essential to satisfy a condition that state i in an HMM and state i in another HMM keep the same linguistic function when these two HMMs correspond to the same word. Using an utterance-specific but temporally aligned HMMs, we obtain a TSR vector sequence.

We can use some commonly-used sequential models like HMM, where plural alignment paths are allowed between a feature sequence and the state sequence of the HMM. In our case, however, because alignment between the TSR vector sequences and the retrained HMM is already determined, TSR vectors of a state of the HMM are modeled as Gaussian distribution. Finally the log likelihood $L_{tsr}(\boldsymbol{c}|\ w_i)$ is given as

$$L_{tsr}(\boldsymbol{c}|\ w_i) = \sum_{t=1}^{T} \log \mathcal{N}(\boldsymbol{d}_t|\ \boldsymbol{\mu}_{w_i}^{(q_t^*)}, \boldsymbol{\Sigma}_{w_i}^{(q_t^*)}), \qquad (5.7)$$

where $\boldsymbol{d}_t$ is the t-th TSR vector, $\boldsymbol{\mu}_{w_i}^{(n)}$ and $\boldsymbol{\Sigma}_{w_i}^{(n)}$ are the mean vector and covariance matrix for the n-th state in w_i, and q_t^* is the state index to which the t-th frame belongs. We adopt the Viterbi path $\boldsymbol{q}^*$ instead of considering all the paths. It is possible to consider all the paths but it is very costly.

5.3.1 Experimental Conditions

We used Tohoku University and Panasonic isolated spoken word database [38], which contains 212 kinds of Japanese words spoken by 60 speakers. The word length varies from 3 morae to 7 morae. We used the utterances by 30 speakers as a training data set and ones by the other 30 speakers as an evaluation data set. In the training, the parameters of TSR model $\boldsymbol{\mu}_{w_i}^{(n)}$ and $\boldsymbol{\Sigma}_{w_i}^{(n)}$ are estimated to maximize Eq.5.7. We set TSR weight w_{tsr} in Eq.5.6 as 1.0×10^{-11} by preliminary experiments. For rescoring, 10-best words are used as candidates. Other conditions are shown in Table 5.4.

5.3.2 Results

We compared three methods, HMM only, HMM rescored by TSR, and HMM rescored by TSR with its Δ and Δ^2. The results are shown in Table.5.5. As shown in the table,

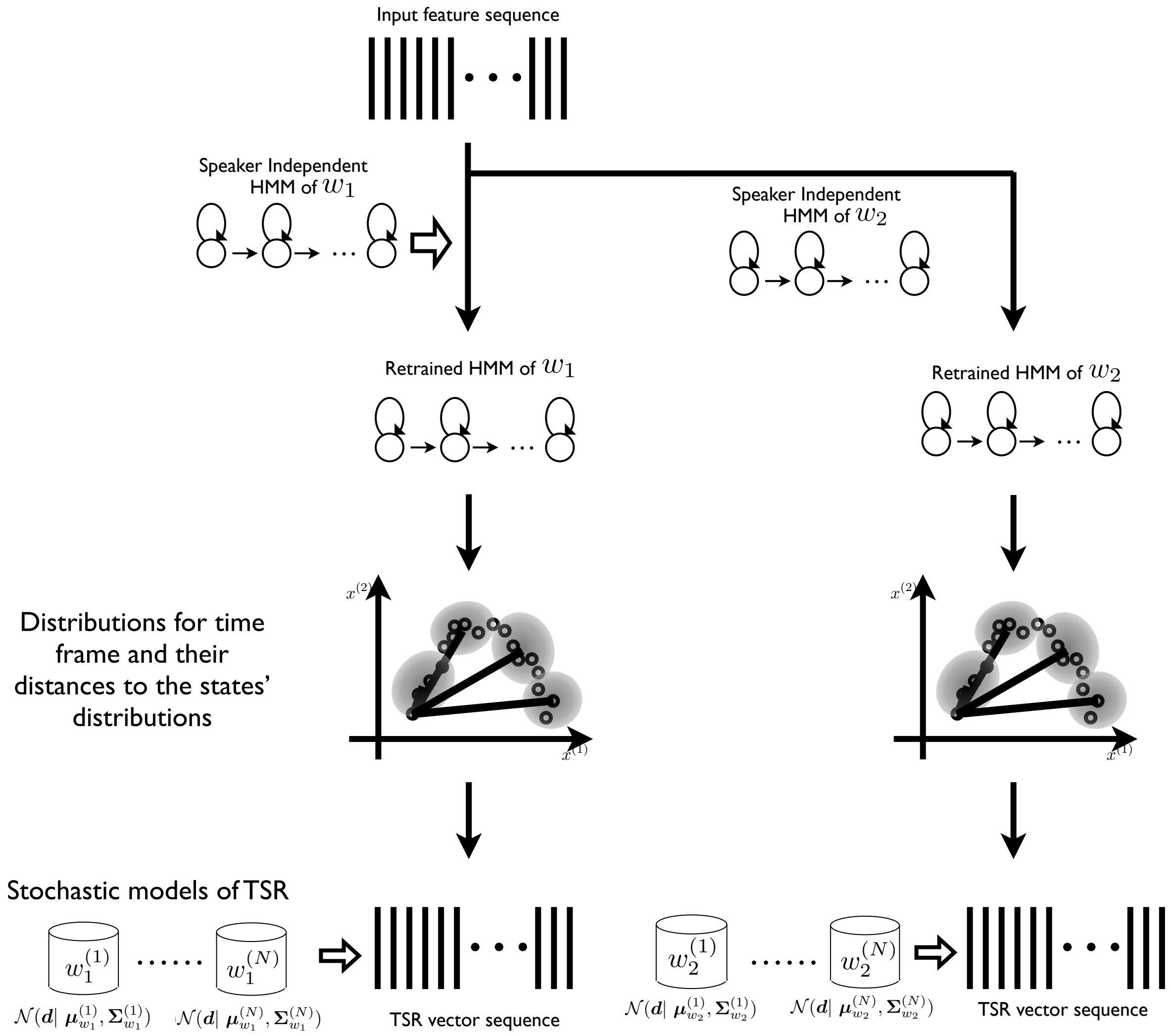

Input feature sequence
Speaker Independent HMM of w_1
Speaker Independent HMM of w_2
Retrained HMM of w_1
Retrained HMM of w_2
Distributions for time frame and their distances to the states' distributions
$x^{(2)}$
$x^{(1)}$
$x^{(2)}$
$x^{(1)}$
Stochastic models of TSR
$w_1^{(1)}$
$w_1^{(N)}$
$\mathcal{N}(\boldsymbol{d}|\,\boldsymbol{\mu}_{w_1}^{(1)},\boldsymbol{\Sigma}_{w_1}^{(1)})$
$\mathcal{N}(\boldsymbol{d}|\,\boldsymbol{\mu}_{w_1}^{(N)},\boldsymbol{\Sigma}_{w_1}^{(N)})$
TSR vector sequence
$w_2^{(1)}$
$w_2^{(N)}$
$\mathcal{N}(\boldsymbol{d}|\,\boldsymbol{\mu}_{w_2}^{(1)},\boldsymbol{\Sigma}_{w_2}^{(1)})$
$\mathcal{N}(\boldsymbol{d}|\,\boldsymbol{\mu}_{w_2}^{(N)},\boldsymbol{\Sigma}_{w_2}^{(N)})$
TSR vector sequence

Fig.5.3: Procedure to calculate TSR likelihood for each candidate

rescoring by TSR decrease the word error rate by 23.3% relative and rescoring by TSR with its Δ and Δ^2 decrease the word error rate by 28.5% relative. The results show that TSR works effectively in the isolated word recognition.

Table5.4: Conditions

Sampling	16bit/16kHz
Window	25ms Blackman Window & 1ms Shift
Feature	mel-cepstrum 18 dim., its Δ and Δ^2
Delta Window Length	20 frames
Word HMM	8-mixture with diagonal matrix 25 states and 23 distributions

Table5.5: Rescoring by TSR decreases the word error rate

Scoring method	Word error rate
HMM	1.37%
HMM + TSR	1.05%
HMM + TSR + Δ TSR + Δ^2 TSR	0.98%
N-best Oracle	0.04 %

Chapter 6

Conclusions

6.1 Summary

Speech structure was proposed as a feature that is invariant to non-linguistic information, and was successfully applied to pronunciation assessment, isolated word recognition, and continuous speech recognition. However, dynamic features have not been used effectively in speech structure, while it was shown that dynamic features work as effective features in almost all speech analysis. In this paper, I proposed two implementations of dynamic features derived from speech structure and carried out some experiments to show their effectiveness.

First, I proposed differential speech structure, which defines temporal derivatives of speech structure. It assumes each cepstrum distribution moves along its delta cepstrum and obtains the temporal derivatives of the distances between cepstrum distributions. Two experiments were carried out to show their effectiveness. One is an experiment of pronunciation assessment. By concatenating the differential speech structure to conventional static speech structure, the correlation between the estimated score and manually labeled score is improved. The other is an experiment of isolated word recognition. Similarly, by concatenating the differential speech structure, 11.0% relative decrease is obtained. Through these two experiments, the effectiveness of differential speech structure is shown.

Second, I proposed trajectory speech structure, which defines a time sequence of speech structure and derive its dynamic features like ordinary delta cepstrum. It assumes each time frame has its own distribution so the time sequence of distances between the distributions can be obtained. The frame-dependent distributions are derived from trajectory HMM. I carried out an experiment of N-best rescoring of isolated word recognition using speech structure, which itself is a new approach to isolated word recognition using speech structure. By concatenating trajectory structure to the conventional static structure, 28.5% relative decrease in word error rate is obtained. The results show that trajectory speech structure works effectively in N-best rescoring of ASR.

The differential speech structure and the trajectory speech structure can be used in any applications using speech structure. Considering the results obtained in the experiments, it is expected that these two approaches to leverage dynamic features in speech structure improve the performance of other applications.

Furthermore, ASR algorithms are undergoing paradigm shift. While the old paradigm algorithms assume frame-by-frame Markov property, the new paradigm algorithms can leverage long-term features by re-evaluating the candidates generated by the old paradigm algorithms. Therefore, the demand for effective long-term features is rapidly increasing. The speech structure and its dynamic features might potentially be a long-term feature which is effectively used in the new paradigm algorithm.

6.2 Future works

Although several new paradigm algorithms in Section 2.5 are introduced, I carried out only an experiment of N-best rescoring, which can be seen as the naivest implementation of the new paradigm algorithms. Therefore, some experiments of combining the features derived from speech structure with the high-performance algorithms, such as segmental conditional random fields and structured support vector machine, should be carried out.

In addition, the relationship between speech structure and TANDEM approaches [39, 40, 41] should be investigated. While trajectory speech structure calculate the distances between the frame-by-frame distributions and the phoneme distributions that are trained from an utterance, TANDEM approaches calculate the utterance's frame-by-frame likelihoods of phoneme models that is trained in advance and adopt the likelihoods as the feature. The distance used in speech structure and the likelihood used in TANDEM is intrinsically the same thing. The biggest difference between trajectory structure and TANDEM is the models to calculate the likelihood/distance. While the models are trained from the the input utterance in speech structure, they are trained from the large amount of training data in TANDEM. Therefore, speech structure can be seen as an adaptive implementation of TANDEM approach. It indicates that speech structure can be applied to the applications to which TANDEM is applied and another improvement on speech structure can be obtained by applying the algorithms used in TANDEM.

Because there exist not a few approach to improve speech structure and the demand for long-term features is increasing, it is expected that some algorithms based on speech structure can be effectively combined with the state-of-the-art ASR systems.

Acknowledgments

本研究を進めるにあたり，ご指導を賜りました指導教員の峯松信明准教授に深く感謝を致します．同時に，様々なご指摘等を頂きました広瀬啓吉准教授にも感謝の意を申し上げます．

さらに，研究室のサーバ管理業務などでお世話になった高橋登技官，秘書の池上恵さんに感謝致します．

また，広瀬・峯松研究室の皆様にお世話になりました．研究に関して本当に多くの助言を頂きました，当時博士課程 (現助教) の齋藤大輔氏及び博士課程の鈴木雅之氏の両氏には言い表せないほどの感謝の意を抱いております．また，同期の千々岩圭吾君にはこれから生きていく上で非常に大きな影響を受けました．彼がいなければ，私の今後は全く違ったと思っており，非常に感謝しています．また同期の砂田宜宏君には未来ガジェットの製作及びそれに対する助言という点で非常にお世話になりました．

最後に，今日まで自分を支えてくれた友人と家族に感謝致します．

2012 年 2 月 8 日
清水 信哉

References

[1] M.J.F. Gales, PC Woodland, H.Y. Chan, D. Mrva, R. Sinha, S.E. Tranter, et al. Progress in the cu-htk broadcast news transcription system. *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 14, No. 5, pp. 1513–1525, 2006. 1

[2] S.F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig. Advances in speech transcription at ibm under the darpa ears program. *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 14, No. 5, pp. 1596–1608, 2006. 1

[3] S. Matsoukas, J.L. Gauvain, G. Adda, T. Colthurst, C.L. Kao, O. Kimball, L. Lamel, F. Lefevre, J.Z. Ma, J. Makhoul, et al. Advances in transcription of broadcast news and conversational telephone speech within the combined ears bbn/limsi system. *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 14, No. 5, pp. 1541–1556, 2006. 1

[4] S.X. Zhang and MJF Gales. Structured support vector machines for noise robust continuous speech recognition. In *Proc. Interspeech*, pp. 989–992, 2011. 3, 12

[5] G. Zweig and P. Nguyen. Scarf: A segmental conditional random field toolkit for speech recognition. In *Eleventh Annual Conference of the International Speech Communication Association*, 2010. 3, 12, 13

[6] J.T. Kao, G. Zweig, and P. Nguyen. Discriminative duration modeling for speech recognition with segmental conditional random fields. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 4476–4479. IEEE, 2011. 3, 12, 13

[7] S. Watanabe, T. Hori, and A. Nakamura. Large vocabulary continuous speech recognition using wfst-based linear classifier for structured data. In *Eleventh Annual Conference of the International Speech Communication Association*, 2010. 3, 12

[8] N. Minematsu. Yet another acoustic representation of speech sounds. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, Vol. 1, pp. I–585. IEEE, 2004. 3, 15

[9] J. Huang, K. Visweswariah, P. Olsen, and V. Goel. Front-end feature transforms with context filtering for speaker adaptation. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 4440–4443. IEEE, 2011. 3

[10] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig. fmpe: Discriminatively trained features for speech recognition. In *Proc. ICASSP*, Vol. 1, pp. 961–964. Philadelphia, 2005. 3

[11] E. Eide and H. Gish. A parametric approach to vocal tract length normalization. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, Vol. 1, pp. 346–348. IEEE, 1996. 3, 11

[12] N. Kumar and A.G. Andreou. Heteroscedastic discriminant analysis and reduced rank hmms for improved speech recognition. *Speech communication*, Vol. 26, No. 4, pp. 283–297, 1998. 3

[13] Y. Li, H. Erdogan, Y. Gao, and E. Marcheret. Incremental on-line feature space mllr adaptation for telephony speech recognition. In *Seventh International Conference on Spoken Language Processing*, 2002. 3

[14] CJ Leggetter and PC Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer speech and language*, Vol. 9, No. 2, p. 171, 1995. 3, 11

[15] N. Minematsu, S. Asakawa, M. Suzuki, and Y. Qiao. Speech structure and its application to robust speech processing. *New Generation Computing*, Vol. 28, No. 3, pp. 299–319, 2010. 3, 15

[16] N. Minematsu, K. Kamata, S. Asakawa, T. Makino, T. Nishimura, and K. Hirose. Structural assessment of language learners' pronunciation. In *Eighth Annual Conference of the International Speech Communication Association*, 2007. 3, 15

[17] M. Suzuki, N. Minematsu, Dean Luo, and K. Hirose. Sub-structure-based estimation of pronunciation proficiency and classification of learners. In *Automatic Speech Recognition Understanding, 2009. ASRU 2009. IEEE Workshop on*, pp. 574 –579, 13 2009-dec. 17 2009. 3, 15

[18] M. Suzuki, L. Dean, N. Minematsu, and K. Hirose. Improved structure-based automatic estimation of pronunciation proficiency. *Proc. ISCA Tutorial and Research Workshop on Speech and Language Technology in Education (SLaTE)*, Vol. 5, , 2009. 3, 15

[19] N. Minematsu and Suzuki M. Structure-based pronunciation assessment. *Proc. ISCA Tutorial and Research Workshop on Speech and Language Technology in Education(SLaTE), Demo Session*, 2009. 3, 15

[20] S. Asakawa, N. Minematsu, and K. Hirose. Multi-stream parameterization for structural speech recognition. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 4097–4100. IEEE, 2008. 3, 15, 19, 20, 28

[21] Y. Qiao, S. Asakawa, and N. Minematsu. Random discriminant structure analysis for continous japanese vowel recognition. In *Proc. ASRU*, pp. 576–581, 2007. 3, 15, 21

[22] Y. Qiao, N. Minematsu, and K. Hirose. On invariant structural representation for speech recognition: theoretical validation and experimental improvement. In *Tenth Annual Conference of the International Speech Communication Association*, 2009. 3, 15, 21

[23] Y. Qiao, M. Suzuki, and N. Minematsu. A study on hidden structural model and its application to labeling sequences. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pp. 118–123. IEEE, 2009. 3, 15

[24] Y. Qiao, M. Suzuki, and N. Minematsu. Affine invariant features and their application to speech recognition. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 4629–4632. IEEE, 2009. 3, 15

[25] M. Suzuki, G. Kurata, M. Nishimura, and N. Minematsu. Continuous digits recognition leveraging invariant structure. *Proc. INTERSPEECH*, pp. 993–996, 2011-8. 3, 15, 21, 22, 23, 40

[26] S. Furui. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, Vol. 34, No. 1, pp. 52–59, 1986. 3, 7, 28

[27] S. Sagayama and F. Itakura. On individuality in a dynamic measure of speech. In *Proc. ASJ Spring Conf*, pp. 3–2, 1979. 7

[28] M. Pitz and H. Ney. Vocal tract normalization equals linear transformation in cepstral space. *Speech and Audio Processing, IEEE Transactions on*, Vol. 13, No. 5, pp. 930–944, 2005. 11

[29] BS Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *J. Acoust. Soc. Am*, Vol. 55, No. 6, pp. 1304–1312, 1974. 11

[30] T. EMORI and K. SHINODA. Vocal tract length normalization using rapid maximum-likelihood estimation for speech recognition. *IEICE Transactions on Information and Systems*, Vol. 83, No. 11, pp. 2108–2117, 2000. 11

[31] M. Suzuki, Y. Qiao, N. Minematsu, and K. Hirose. Pronunciation proficiency estimation based on multilayer regression analysis using speaker-independent structural features. In *Second Language Studies: Acquisition, Learning, Education and Technology*, 2010. 17, 18

[32] T. Emori and K. Shinoda. Rapid vocal tract length normalization using maximum likelihood estimation. In *Seventh European Conference on Speech Communication and Technology*, 2001. 20

[33] ASAKAWA Satoshi, QIAO Yu, MINEMATSU Nobuaki, and HIROSE Keikichi. Isolated word recognition based on speech structures and discriminant analysis. *IEICE technical report. Natural language understanding and models of communication*, Vol. 108, No. 337, pp. 203–208, 2008-12-02. 21

[34] SAITO Daisuke, MATSUURA Ryo, MINEMATSU Nobuaki, and HIROSE Keikichi. Experimental study of acoustic modeling using speaker-invariant speech contrast as modeling unit. *IEICE technical report. Natural language understanding and models of communication*, Vol. 109, No. 355, pp. 7–12, 2009-12-14. 21

[35] H. Zen, K. Tokuda, and T. Kitamura. Reformulating the hmm as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences. *Computer Speech & Language*, Vol. 21, No. 1, pp. 153–173, 2007. 28, 30

[36] ASAKAWA Satoshi, MINEMATSU Nobuaki, MURAKAMI Takao, ISEI Toshiko, and HIROSE Keikichi. Analysis of the non-native english pronunciation based on structural representation of speech. *IEICE technical report. Speech*, Vol. 105, No. 132, pp. 25–30, 2005-06-16. 36

[37] MINEMATSU Nobuaki, TOMIYAMA Yoshihiro, YOSHIMOTO Kei, SHIMIZU Katsumasa, NAKAGAWA Seiichi, DANTSUJI Masatake, and MAKINO Shozo. Development of english speech database read by japanese and americans for call system development (¡special issue¿ educational technology research on second language learning and its assistance). *Japan journal of educational technology*, Vol. 27, No. 3, pp. 259–272, 2003-12-20. 37

[38] Shozo Makino, Niyada Katsuyuki, Mafune Yasuo, and Kido Kin'iti. Tohoku university and panasonic isolated spoken word database. *Acoustical Science and Technology*, Vol. 48, No. 12, pp. 899–905, 1992-12-01. 39, 41

[39] H. Hermansky, D.P.W. Ellis, and S. Sharma. Tandem connectionist feature extraction for conventional hmm systems. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, Vol. 3, pp. 1635–1638. Ieee, 2000. 45

[40] D.P.W. Ellis, R. Singh, and S. Sivadas. Tandem acoustic modeling in large-vocabulary recognition. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, Vol. 1, pp. 517–520. IEEE, 2001. 45

[41] Q. Zhu, A. Stolcke, B.Y. Chen, and N. Morgan. Incorporating tandem/hats mlp features into sri ' s conversational speech recognition system. In *Proc. DARPA Rich Transcription Workshop*. Citeseer, 2004. 45

Publications

Domestic conferences

[1] S. Shimizu, D. Saito, M. Suzuki, N. Minematsu, K. Hirose, “Training of robust language models by automatic sentence generation based on word replacement”, Proc. Spring Meeting of the Acoustical Society of Japan, 2010-3

[2] S. Shimizu, M. Suzuki, D. Saito, N. Minematsu, K. Hirose, “Training of robust language models by automatic sentence generation based on word replacing words with respect to their contexts”, IPSJ SIG Notes 2010-SLP-81(9), pp.1-6, 2010-5

[3] S. Shimizu, D. Saito, M. Suzuki, N. Minematsu, K. Hirose, “Automatic sentence generation for training language models based on word replacement considering the changes of word usage”, The 24th Annual Conference of the Japanese Society for Articial Intelligence, 2G1-OS3-4, 2010-6

[4] K. Takano, S. Shimizu, N. Minematsu, K. Hirose, “Experimental study on improving the performance of accent sandhi prediction using accentual phrase boundaries in a compound noun”, Proc. Autumn Meeting of the Acoustical Society of Japan, 2011-3

[5] S. Shimizu, M. Suzuki, N. Minematsu, K. Hirose, “An experimental study on dynamic features of speech structure”, Proc. Autumn Meeting of the Acoustical Society of Japan, 2011-9

[6] S. Shimizu, M. Suzuki, N. Minematsu, K. Hirose, “Reformulating Speech Structure Based on Trajectory HMM”, Proc. Spring Meeting of the Acoustical Society of Japan, 2012-3 (submitted)

[7] N. Sunada, S. Shimizu, N. Minematsu, K. Hirose, “Efficient Phase Optimization for Higher Sound Energy Density”, Proc. Spring Meeting of the Acoustical Society of Japan, 2012-3 (submitted)

[8] S. Kobayashi, S. Shimizu, N. Minematsu, K. Hirose, “Inprovement of CRF-based accent sandhi prediction by considering accentual rules of numeral phrases”, Proc. Spring Meeting of the Acoustical Society of Japan, 2012-3 (submitted)

[9] S. Kobayashi, S. Shimizu, N. Minematsu, K. Hirose, “Improvement of CRF-Based Accent Sandhi Prediction Using Rule-Based Features”, IEICE technical report. Speech, 2012-3 (submitted)

International conferences

[10] S. Kobayashi, S. Shimizu, M. Suzuki, N. Minematsu, K. Hirose, and H. Hirano “Automatic Generation of Accent Dictionary of Conjugational Words for Any Japanese Texts,” Proc. Int. Conf. on Japanese Language Education (ICJLE’2011) (2011-7)

[11] S. Shimizu, M. Suzuki, N. Minematsu, K. Hirose, “An Experimental Study On Dynamic Features Of Speech Structure”, Proc. Int. Workshop on Nonlinear Circuits, Communication and Signal Processing (NCSP ’ 2012), 2012-3 (submitted)

[12] N. Sunada, S. Shimizu, K. Hirose, N. Minematsu, “Efficient Phase Optimization For Sounds With Condensed Energy Density”, Proc. Int. Workshop on Nonlinear Circuits, Communication and Signal Processing (NCSP ’ 2012), 2012-3(submitted)