Technical University of Crete
Department of Electronic and Computer Engineering
Telecommunications Division

Master of Science Thesis

# Detection-Based Automatic Speech Recognition

by

**Spiros Dimopoulos**

Supervisor: Assoc. Professor Alexandros Potamianos
Committee Member: Professor Vasilios Digalakis
Committee Member: Professor Athanasios Liavas

Chania, 2010

# Contents

# Acknowledgements

First, I would like to thank my professor Alexandros Potamianos for the excellent collaboration we had during my Master's degree. He advised me and helped me achieve my academic goals in every aspect. Also, I would like to thank the stuff at Slate Lab, Computer Science and Engineering Dept., Ohio State University, and especially Jeremy Morris and professor Eric Fossler-Lussier, for the conversations and the collaboration we had during my stay there. Also the stuff at CSIP Lab, Electrical and Computer Engineering Dept. Georgia Institute of Technology, especially Yu Tsao and prof. Chin-Hui Lee, for helping me out during my stay there and the collaboration we had.

I would also like to thank Onassis Scholarship Foundation, for the economic support that partially funded my graduate studies at Technical University of Crete and helped me achieve this work.

I would also like to thank Eutichia Arvaniti for being on my side all the time during my graduate studies and supporting me whatever decisions I had taken. She was there sharing the fun at good times, but also helped me keep my foot on the ground and keep my mind sane at diffcult times. Finally I would like to thank my family for supporting my studies, and friends at Telecommunications Laboratory, Technical University of Crete for the good times we had.

# List of Figures

# List of Tables

# List of Symbols and Abbreviations

| Abbreviation | Description | Definition |
|---|---|---|
| ASR | Automatic Speech Recognition | page 9 |
| MLP | Multi Layer Perceptron | page 10 |
| HMM | Hidden Markov Model | page 10 |
| GMM | Gaussian Mixture Model | page 11 |
| VFR | Variable Frame Rate | page 11 |
| CRF | Conditional Random Fields | page 13 |
| MFCC | Mel-Frequency Cepstrum Coefficients | page 15 |
| PLP | Perceptual Linear Predictive | page 17 |
| LPC | Linear Predictive Coding | page 17 |
| ADC | Analogue to Digital Conversion | page 16 |
| FR | Frame-rate | page 15 |
| WS | Window-size | page 15 |
| DCT | Discrete Cosine Transform | page 16 |
| ML | Maximum Likelihood | page 18 |
| ANN | Artificial Neural Network | page 20 |
| ROS | Rate of Speech | page 32 |
| FRSU | Frame Rate Selection Unit | page 39 |
| MCE | Minimum Classification Error | page 39 |
| FFT | Fast Fourier Transformation | page 40 |
| ICSI | International Computer Science Institute | page 55 |
| KLT | Karhune-Loeve Transfrom | page 59 |
| ZCC | Zero Crossing Count | page 91 |

# Abstract

The long term goal of automatic speech recognition (ASR) is to achieve at least comparable results to human speech recognition. It has been noted, at least a decade ago, that the performance of ASR is an order of magnitude lower than human speech recognition. When the quality of captured speech signal gets degraded (noise, channel variability), then ASR performs even worse.

In this thesis, the current ASR paradigm is shifted towards a novel detection-based approach, inspired by the human understanding of speech. The main concepts discussed lie in the areas of low-level acoustic phonetic modeling and the rapid adaptation of the system to channel and talker-style variability.

Human understanding of speech has been shown to facilitate a bottom up combination of speech events in order to form a hypothesis. When this approach is transfered to ASR the detection is usually comprised of low-level acoustic-phonetic event detection and speaker and environment attributes. An efficient combinatory scheme is necessary to merge the detected events and form hypothesized transcriptions.

We lean towards a new approach to ASR that is less data driven and more event driven. This detection based approach uses state of the art model driven speech technology as basis to detect speech events and attributes and then new combination techniques to form higher level transcriptions. It uses good practices from both acoustic-phonetic and statistical-modeling (data-driven) approaches to speech recognition. It does take into consideration the knowledge that is present is speech beyond data driven statistical phonetic modeling. But does not ignore the great advances that have been achieved in data-driven approach.

## Chapter 1

# Introduction

The long term goal of automatic speech recognition (ASR) is to achieve at least comparable results to human speech recognition. It has been noted, at least a decade ago, that the performance of ASR is an order of magnitude lower than human speech recognition [36]. When the quality of captured speech signal gets degraded (noise, channel variability), then ASR performs even worse. One of the four ASR areas that were deemed as in need of improvement, was the low-level acoustic-phonetic modeling. Another area was the rapid adaptation of the system to channel and talker-style variability. The main concepts that are discussed in this thesis lie in these two areas.

Human understanding of speech has been shown to facilitate a bottom up combination of speech events in order to form a hypothesis. When this approach is transfered to ASR the detection is usually comprised of low-level acoustic-phonetic event detection and speaker and environment attributes [31]. After the detection process a merging step is necessary to combine the detected events and form the hypothesized transcription. An efficient combinatory scheme is necessary to be used in this step. The combined events and attributes are of different time resolution and quality and have redundant information [42].

Clearly we see that inspired by the human perception and processing of speech we lean towards a new approach to ASR that is less data driven and more event driven [30]. This detection based approach uses state of the art model driven speech technology as a basis to detect speech events and attributes and then new combination techniques to form higher level transcriptions. It is different from the previously tested acoustic-phonetic approaches to speech recognition. They tried to detect phonetic events as binary features and combine them in deterministic non-statistical way [50]. Instead this new approach uses good practices from both acoustic-phonetic and statistical-modeling (data-driven) approaches to speech recognition. It does take into consideration the knowledge that is present is speech beyond data driven statistical phonetic modeling. But does not ignore the great advances that have been achieved in this data-driven approach.

This thesis implements a baseline detection-based speech recognition system and then enhances it by adopting a speech rate adaptation scheme. Finally it investigates the segmental approach to speech recognition, and sets the basis for a transition modeling based segmental system.

## 1.1 Prior Work

In this section we focus on prior work, in the three domains that were stated in the last paragraph of previous section. Namely, we present classic research and the state of the art in research technology of speech event detectors, speech rate variability modeling and segmental speech recognition.

### 1.1.1 Detection of Speech Events

Detection of speech events is closely related to a detection-based speech recognition system. The current ASR technology paradigm is mostly statistical and data-driven. It ignores advances in knowledge of linguistic science and other speech related fields. To overcome these limitations, a new acoustic-phonetic, event-driven, knowledge-rich, open approach was proposed [30]. An ultimate component of this new approach is the detection of low level speech events. So a solid and rigorous event detector bank is necessary as a first step. In a similar manner to keyword spotting, the detection should detect speech attributes that are present and reject locally absent speech attributes as proposed in [35]. Also the detectors can be frame-based or segment-based with the latter giving better results when detecting manner of articulation events. Further improvement can be achieved when using discriminative learning criteria in the design of detectors.

Another approach to detection-based speech recognition proposed in [39], was to detect the words in an utterance using word detectors and then apply knowledge-based pruning of unlikely words. The pruning methods used were based on (a) duration constrains, (b) phonetic attributes detection and (c) signal features. Also this work introduced the very important concept of hypothesis combination in detection-based ASR.

In [1], various statistical modeling technologies are used to create and test phonetic attribute detectors. Multi-Layer Perceptron (MLP), Hidden Markov Model (HMM) and Support Vector Machine (SVM) detectors of acoustic-phonetic features are implemented and evaluated. Also the combination of these detected features to form higher level hypothesis is investigated.

Another important work in this area, uses output from MLPs as input to an ASR system, the so called tandem approach [2]. The novel concept in this work, is the use of MLPs, trained for articulatory feature detection. Previous tandem approaches [66] used MLPs trained for phone classification. In [46], articulatory feature detectors are implemented using Gaussian Mixture Models (GMM) and MLP. After the detection process, a combination of these features is done using another MLP and finally the tandem approach is followed. In [51], hierarchical systems of articulatory features are evaluated and their failure is reported. Also it is proposed to use joint modeling of articulatory features instead; that's because in hierarchical systems, the error is propagated from parent to child class. Finally a very important task in detection of speech events, is the creation of transcriptions of speech at the articulatory feature level. Good practices of manual labeling of speech at this level are discussed in [37].

### 1.1.2 Speech Rate Variability

Speech is a dynamic phenomenon that has stationary segments extending to 100 milliseconds in length, but also has segments of a couple of milliseconds with fast changing spectral content. Also speaker rate variability is a factor affecting the rate of change of speech content. In [63], parallel, rate-specific acoustic models are proposed to deal with the variations in rate of speech that affect both spectral features and word pronunciations. Two categories of models are used, the fast speech and the slow speech model. Rate switching is permitted at word boundaries so within-sentence speech rate variations are modeled. Following another approach in [3], a multi-rate extension of HMMs is proposed, for joint acoustic modeling of speech at multiple time scales. The usual short-term, phone-representation of speech is complemented with wide-context modeling units (syllable and stress). A similar approach is followed in [11] and [19]; multi-stream HMMs are used to incorporate multiple time scale information as independent streams. In the former work, a combination of phone and syllable scale processing is used as previously described. In the latter, different time scale processing of input features is used to create multiple streams and then combine them in a multi-stream HMM framework.

One method is presented in [15], where a typical fixed-rate recognizer is augmented with information from multiple rate spectral models. The appropriate model for each segment of speech is determined by using the hypotheses generated by the fixed-rate recognizer. To achieve this, N-Best list rescoring is done with acoustical models using different temporal windows. Other methods that don't need modification in the underlying statistical models have been tested. In [64], a Variable Frame Rate (VFR) algorithm is introduced that results in an increased number of frames for rapidly-changing segments with relatively high energy and less frames for steady-state segments. Finally in [62], an improvement is proposed for the previous VFR algorithm that uses the entropy of the signal instead of an Euclidean distance measure to compute the frame-rate.

### 1.1.3 Segmental Speech Recognition

Recently, new models are researched to overcome the shortcomings of HMMs. Segmental modeling and processing of speech is widely popular and researched by many groups. It is an extension to the standard HMM approach using higher order models and modeling explicitly the duration of segments. Due to the similarity with HMMs the same training and decoding algorithms can be used, but with increased computational cost. It is necessary to include all possible segmentations of the utterance into the search space. A perfect introduction to this area is available in [47]. In [34], a major problem in segmental recognition, the segmentation, is considered and a probabilistic solution is proposed. A high quality segmentation is necessary to include as many actual segments as possible. In [55], segmental acoustic models are trained using discriminative training algorithms. Another problem of segmental processing, the reduction of possible segments that are considered during decoding but without losing the clarity of including the entire graph of segments is addressed in [4]. Finally in [44], an extension to HMM is introduced that can explicitly model the phoneme boundaries using an acoustic feature set that is associated with state transitions.

## 1.2 Thesis Objective

Current speech recognition paradigm is mainly statistical-based and data-driven. It uses statistical models (HMM) to model the speech units and bigram and trigram models for language modeling. More or less, these models work as black boxes that are trained and used for decoding during recognition. Except for choosing the number of internal model parameters, there are not many possibilities of altering the model structure. This is considered a hindering factor in further advancement of ASR technology [1, 30].

This thesis introduces an ASR paradigm that is inspired by current theories on human perception of speech [58, 53, 7, 23, 6] and by recent advances in statistical ASR [2, 25, 18, 42]. Human perception of speech works in a bottom-up fashion, analyzing and processing information from the signal, sup-phonetic, phonetic, syllable, word and sentence levels. All this work is done in parallel and the information is integrated forming possible hypotheses. These hypotheses are constantly being verified until a solid decision is reached [12, 22, 30]. In this process knowledge available at all levels is extracted and exploited in order to have as accurate results as possible. We propose an open system that can accept further improvements that could help the recognition process. Such additions can be speaker accent, gender, rate of speech etc. and speaking environment conditions, such as noise.

In figure 1.1 we can see the main components and relations of the proposed system. First the speech signal is analyzed and processed to extract useful features.

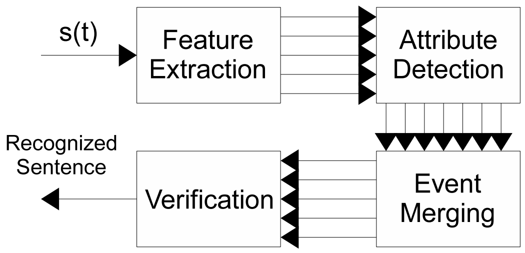


Figure 1.1: Main component flowchart of the detection-based ASR paradigm.

Next is the detection of speech events at different levels of speech hierarchy. This is the main and most important process of the system, so the whole paradigm is called detection-based ASR. Accurately detection and time-placement of speech events is necessary to successfully proceed to next step. After this, the combination of speech events is performed to form hypotheses. Last thing, the verification of generated hypotheses and final utterance recognition are completed. The system presented is an open system in the sense that different implementation can be used for each component. We test many different implementations and keep the best combination of features, events, merging and verification techniques. Furthermore, in future when linguistic knowledge and statistical techniques advance, we can substitute a part of system or add a new component, without changing the others.

Chapter 2 introduces the state-of-the-art HMM-based ASR methodology. Also a new type of features is introduced that better describes speech events. Finally non explicitly HMM-based speech recognition techniques are presented.

Chapter 3 presents Conditional Random Fields (CRF) framework as a proposed event merger and verifier. The framework is a contribution of the Speech and Language Technology Lab, Computer and Software Engineering Department, Ohio State University.

Chapter 4 experiments with the idea of adding more temporal resolutions into the system to catch events of different time-scale and compensate for the speech rate variability.

Chapter 5 introduces the segmental processing of speech recognition. An initial approach of inserting segmental processing capabilities into the detection-based system is presented and discussed. This Chapter was written in close collaboration with Chin-Hui Lee, CSIP Lab, Electrical and Computer Engineering, Georgia Institute of Technology.

Chapter 6 presents the results from the experiments conducted during the research of the problems presented in previous chapters.

Chapter 7 presents the conclusions and proposes further improvements.

Chapter 2

# Speech Recognition and the detection-based approach

This chapter introduces state-of-art statistical modeling techniques in speech recognition. Current ASR paradigm is based in spectral-based feature extraction and HMM modeling and their corresponding algorithms. Although quite restrictive, it has been successful so far. Later in the chapter, a new family of acoustic features is presented, based on acoustic-phonetic events. The last section introduces novel modeling approaches that use a combination of statistical modeling techniques beyond HMMs.

## 2.1 Spectral features for speech signal representation

In this section, we introduce the standard and most popular transformation of speech signal into meaningful and useful parameters. This transformation uses spectral information present in the speech signal, in order to extract spectral-based features. These feature represent the speech signal in the recognition process. The representation must be as compact as possible but without losing critical information from the original signal. A trade-off has been reached to the most appropriate representation and is called Mel-Frequency Cepstrum Coefficients (MFCC) [60].

The core of the transformation process is based on a filterbank analysis of the speech signal spectrum. The whole process is shown in Figure 2.1. The processing of the signal is done in windows, which are usually overlapping. The frame-rate (FR) and window-size (WS) are important parameters of the process and are expressed in milliseconds. Pre-processing of speech is necessary to make it ready for further processing. The pre-processing step usually consists of three tasks:

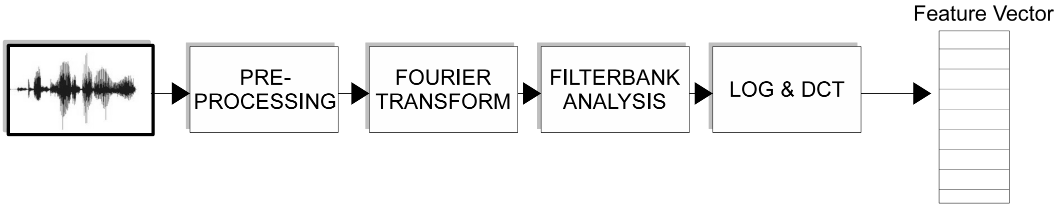


Figure 2.1: MFCC parameter extraction process from speech signal

DC removal, pre-emphasis and windowing [13]. Analog to Digital Conversion (ADC) of speech signal can add a DC offset. It is useful to remove this DC offset from each window before further processing. Also a pre-emphasis of the signal is common practice by applying the equation:

$$s'_n = s_n - \alpha s_{n-1} \tag{2.1}$$

where $s_n$ are the samples in each window $n = \{1, ..., N\}$ with $N$ the WS, and $\alpha$ the pre-emphasis coefficient in the range $0 \leq \alpha < 1$. Final pre-processing step is to apply a Hamming window to the samples of the window by the following transformation:

$$s'_n = \left\{0.54 - 0.46cos\left(\frac{2\pi(n-1)}{N-1}\right)\right\} s_n \tag{2.2}$$

again where $s_n$ are the samples in each window $n = \{1, ..., N\}$ with $N$ the WS.

After pre-processing the signal is transformed in the frequency domain by applying a Fourier Transform. Next step is to perform a filterbank analysis on the spectral representation of the speech signal. Filterbank analysis is non-linear and is based on the assumption that the human ear resolves frequencies in non-linear fashion. The analysis is done by first creating a set of triangular filters on the mel-scale which is defined by:

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

with $f$ the frequency. Lowest and highest cut-off frequencies can be set to leave out unwanted spectral regions of the signal. Then the magnitude coefficients from the Fourier analysis of the signal are multiplied by the corresponding filter gain and the results are accumulated in bins in each filterbank channel.

Because the filterbank coefficients are highly correlated further processing is necessary. The filterbank coefficient are logged and then a Discrete Cosine Transform (DCT) is applied. The DCT is described by the formula:

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j cos(\frac{\pi i}{N}(j - 0.5)) \tag{2.4}$$

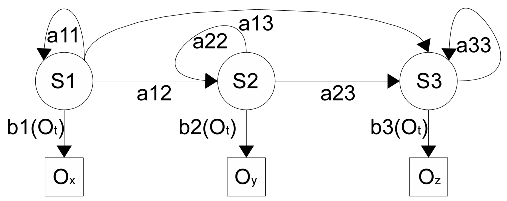


Figure 2.2: 3-state HMM visual representation

where $N$ is the number of filterbank channels and $c_i$ the cepstral coefficients. The number of filterbank channels and the number of cepstral coefficients are parameters of the process.

There are other methods of speech signal transformation and feature extraction, based on Linear Predictive Coding (LPC). A method used in this thesis is the Perceptual Linear Predictive (PLP) analysis of speech. This a method is more consistent with human hearing that the simple LPC. It uses an autoregressive all-pole model to approximate the audio spectrum. A 5-th order model is appropriate for speaker-independent speech recognition application because it suppresses the speaker-dependent details. More details can be found in [20].

## 2.2 Hidden Markov Model (HMM) recognition system

State-of-the-art speech recognition systems are based on HMMs. These statistical models are used in many applications from image recognition to intrusion detection and genomic sequence analysis. The underlying philosophy is simple and is based on Markov process modeling. HMMs are Markov models with hidden states. Only the observation is available but not the state in which is the model. To better understand the HMM philosophy we present a simple 3-state HMM that is frequently used in speech recognition in Figure 2.2.

The model consists of 3 states which are not directly observable. Between these states there are possible transitions which are marked by the arrows. We can see that this type of model used in speech recognition has only left-to-right transitions. The transitions are probabilistic and use the symbols $aij$ with $i$ the source state and $j$ the destination state. From every state the outgoing probabilities must sum to 1. This is a requirement of the state transition probabilities and mathematically is expressed by equation:

$$\sum_{k=1}^{N} a_{ik} = 1, \forall i \in N \tag{2.5}$$

where $N$ is the total number of states of the HMM. We have also the emission probabilities $bj(O_t)$ which are associated with each state. This probability expresses how probable is for a state $j$ to have emitted the symbol $O_t$. Note here that each time frame is associated with a symbol. Symbols can be vectors and are directly connected with speech signal parameter vectors that emerge from feature extraction of Section 2.1. For emission probabilities also holds the requirement that all possible symbol emission probabilities for a state must sum to 1. The mathematical formula is:

$$\sum_{k=1}^{M} b_j(O_k), \forall j \in N \tag{2.6}$$

where $M$ is the total number of symbols associated with the HMM and $N$ is the total number of states of the HMM.

After giving the formulation of the HMM we have to consider the three canonical problems of the HMMs. A summary of each one of the problems is given below:

**Evaluation** Given the parameters of the model and the observation sequence, compute the probability of the sequence.

**Estimation** Given the observation sequence, find the state sequence that is more likely to have generated the observation sequence.

**Training** Given a set of observation sequences, find a set of model parameters that maximizes the probability of the observation sequence.

The solution to the evaluation problem is given efficiently by the Forward-Backward algorithm. The estimation problem is efficiently solved by the Viterbi algorithm. The training problem does not have a known analytical solution, but the Baum-Welch algorithm finds a Maximum Likelihood (ML) estimate of the unknown parameters. Detailed description of the algorithms that solve the HMM problems can be found in [13] and [49].

In order to use the HMM framework (model and algorithms) into the speech recognition area, we have to map the ASR problems to HMM solutions. At first we choose the basic speech unit we want to model. According to the application it can be word, digit, phoneme etc. Then we have to train the parameters of each HMM using a sequence of observation with already known correspondence to speech units. This correspondence is given by the transcription of the observed sequence. This is the **training** problem of HMM. With the trained models, we can find the best model describing an isolated-speech segment, in case we have already correct segmentation of the speech utterance. This is the **evaluation** problem of HMM. In case we have an utterance with unknown segmentation, we concatenate our models forming all possible combinations and consider this as supermodel. In this model we search for the best state sequence to describe the observation sequence, and this is the **estimation** problem of HMM. With this mapping we can proceed to use HMMs for speech recognition.

## 2.3 Features beyond Cepstral Coefficients

Back in the first years of ASR research, an acoustic-phonetic approach to speech recognition was proposed [50]. It used phonological attributes, described in this section, as binary features. Binary features were either present or absent. It was soon abandoned for the statistical data-driven approach that dominates the field until today. The main problem was the lack of successful detectors at that time. In the case of a false detection of an attribute, the whole integration process was mislead. Also the lack of research in combination methods was a hindrance. Now with the advances in statistical modeling and pattern recognition [5], we can return to this acoustic-phonetic approach and reconsider it in a new way as presented in this chapter and Chapter 3.

Speech signal representations are not limited to those described in Section 2.1. Different speech signal analysis methods can be used to extract useful information. As it will be presented later in Chapter 5 Section 5.2 different signal processing methods can be applied based on the application (i.e. boundary detection and segmentation).

Also based in already described spectral-based parameters of section 2.1, one can define acoustic-phonetic representations of speech. They are widely known as phonological attributes, acoustic-phonetic attributes or articulatory features. These phonological attributes are based on and supported by linguistic knowledge. Every distinct phonetic unit in speech, is generated by a certain configuration of all the physical organs of the speaking person that are involved in speech production. The phonological attribute set constitutes an abstract description of this configuration. Every phonetic unit, phone or phoneme is associated with a set of phonological attribute values. This set of values represents the configuration state of the generative physical units of the speaker.

It is widely accepted that using phonological attributes in ASR can improve the performance. This has been either proved by using direct measurements of the speech production signals by physical measurement from electromagnetic articulograph [59], or by using phonological attribute values estimated from the speech signal into an ASR system setup [46, 54, 38].

To use these phonological (acoustic-phonetic) attributes in a recognition system, they have to be estimated from the speech signal in lack of another viable method to detect them; electromagnetic articulograph cannot be considered an option in normal ASR applications. The detection of phonological attributes is a process of reverse engineering the creation of speech sounds existing in speech signal in order to find the original configuration of the speakers physical units

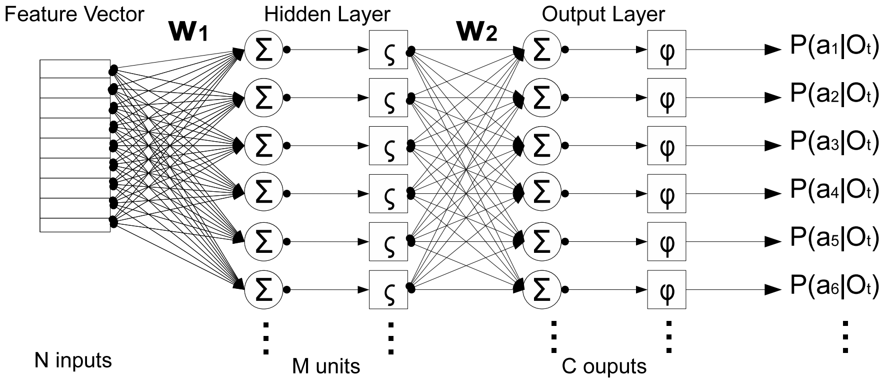


Figure 2.3: MLP detector of phonological attributes giving posterior probability of presence given the input speech parameter vector

(tongue, mouth, chords etc.) that created this sound. If this configuration is known, it can be related to a phonetic unit, according to prior linguistic knowledge about the creation of speech.

At this point, it should be obvious that to implement a successful detection-based ASR system, first of all effective detectors of lower level phonological attributes have to be designed. Phonological attribute presence can be binary, thus binary detectors of either presence or absence of an attribute are necessary. Binary attributes have some complications when trying to combine them and have proved to be ineffective [50]. A better approach is to compute the probability of the presence or absence of a phonological attribute and use it as a feature for phonetic unit recognition. A multitude of statistical methods to design probabilistic phonological attribute detectors can be used [1]. Most frequently Artificial Neural Network (ANN) [38], HMM [35] or GMMs [46] detectors are used. All methods are data-driven and train the detectors to as many examples of the attribute as possible. In our experiments we use ANN detectors and more specifically MLPs [65]. An example of an MLP detector in action is shown in Figure 2.3. It uses speech signal parameters of Section 2.1 as input and gives phonological attribute probability presence as output. It has an intermediate hidden layer of M units and a final output layer of C units. Each layer is connected to the next layer with a vector $\mathbf{w}_{\{}\mathbf{1}, \mathbf{2}\}$ of weights that indicate the contribution of each node to the associated sum. After the sum there is the activation function that triggers the output of the layer. The hidden layer has a sigmoid activation function and the output layer has a softmax activation function. Softmax is used to convert the output to posterior probabilities. A more complete description of ANNs and their applications can be found in [10].

In Table 2.1, there are 8 groups of phonological attributes. Each group is an International Phonetic Association (IPA) class and can take several values. The total number of attributes is 44 and are estimated using MLPs. One multi-class MLP (3-9 classes) is created for each group, totaling 8 MLPs. In the output of the MLPs, a posterior probability estimate for each attribute is generated.

| Class | Attributes |
|---|---|
| **Sonority** | Obstruent, Silence, Sonorant, Syllabic, Vowel |
| **Voicing** | NA, Voiced, Voiceless |
| **Manner** | Approximant, Flap, Fricative, NA, Nasal, NasalFlap, Stop-Closure, Stop |
| **Place** | Alveolar, Dental, Glottal, Labial, Lateral, NA, Palatal, Rhotic, Velar |
| **Height** | High, Low-High, Low, Mid-High, Mid, NA |
| **Backness** | Back, Back-Front, Central, Front, NA |
| **Roundness** | NA, NonRound, NonRound-Round, Round-NonRound, Round |
| **Tenseness** | Lax, NA, Tense |

Table 2.1: IPA phonological attributes

To make the detectors speaker independent, an audio database that is comprised of different speakers has to be used. Another important aspect of the database is that it has to contain speech examples transcribed in the phonetic unit level. A widely used database with these features is the TIMIT database [16]. Timit uses 61 phonemes to transcribe the utterances. Also a mapping of phonetic units to phonological attribute sets that are present during the generation of the phonetic units is necessary. The latter is not an engineering problem as it is studied and resolved by linguistics. In Appendix A.1 there is a complete description of the TIMIT database used in this thesis. In Appendix A.2, there is the complete map associating 48 phoneme labels of TIMIT to 44 phonological attributes.

Designing effective attribute detectors is a difficult task. It can be compared with keyword spotting, only that phonological attributes are much shorter in time and less stable to detect. It is important to detect phonological attribute positions in time as accurately as possible. Every false detection can be carried over to the next level of phonetic recognition. Also segmental attribute detectors have been designed using HMMs and discriminative training, but are not used in this thesis. [35]

## 2.4 Hybrid Recognition Techniques

A recognition system that uses more than one statistical method (e.g. HMM and ANNs) to model the phonetic units is usually called a hybrid system. HMM systems make unrealistic assumptions about the observation vector statistical properties. The independence assumption is the most popular and is better described in Section 3.1.

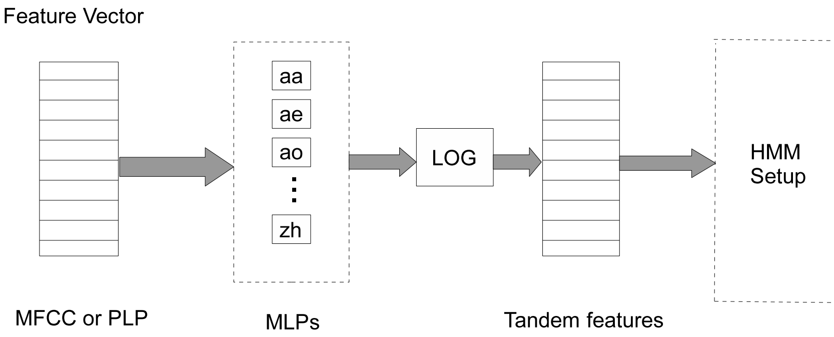


Figure 2.4: The tandem connectionist approach

To overcome these obstacles, some proposed the HMM-ANN hybrid approach. This approach uses ANNs instead of GMMs in the HMM's states to model the observation densities [56]. The HMM structure with the transition probabilities and the associated algorithms already described in Section 2.2 are used. ANNs are used to learn and estimate the emission probabilities discriminatively.

A more recent approach is the ANN-HMM approach, also known as the tandem approach [21]. Figure 2.4 shows the main process components of the tandem approach. Between the speech parameter extraction and the HMM recognition components is inserted an ANN classifier of phonetic units, usually phonemes. The ANN classifier uses the speech parameters as input and discrminatively learns the posterior probabilities of phonetic units. Then these posterior probabilities are linearized using a log functions and used as input features in an HMM recognition setup. This approach has proved to be quite effective in boosting speech recognition performance. Another variant of the tandem approach is the articulatory feature tandem approach. The MLP classifies articulatory features, or else known phonological attributes, instead of phonemes. Then articulatory feature posterior probabilities are linearized and used as input features into the HMM setup.

The detection-based system presented in this thesis, is greatly influenced by these hybrid methods. It uses a similar architecture to ANN-HMM, but with the addition of another statistical modeling toolkit, the CRFs. Chapter 3 provides further information of the CRF framework that does the task of event merging, shown in Figure 1.1.

## Chapter 3

# Conditional Random Fields (CRF) for feature combination

In this chapter, a framework is introduced, that has been applied recently to the ASR domain. The Conditional Random Fields (CRF) framework is a novel approach for the combination and effective usage of discriminatively created features, highly correlated in general [42].

Unlike HMM that is a generative model, CRF is a discriminative model and does not suffer from the feature independence restrictions of the former. It can very well model long-range dependencies between states and observations. It is based on exponential distribution functions associated with states and transitions. These functions use information from the input feature vectors and trainable weights to estimate the posterior probability of an output label sequence given the input feature sequence. Although arbitrary dependencies can be modeled using CRFs, a Markov structure is imposed on state sequence in order to apply a Viterbi algorithm for decoding. Contrary to HMMs, CRFs do not have normalized probability distributions for transition and emission probabilities. So they do not need any special purpose algorithms during training. They can be trained using direct optimization techniques or stochastic gradient descent [18].

## 3.1 The feature independence assumption

Speech modeling using HMMs holds a couple of unrealistic assumptions about observation features. The first assumption is due to the HMM model structure and definition in general and states that the features of successive frames have to be independent. This is not true; in contrary, features of successive frames are correlated. Especially discriminative features of Section 2.3 are highly correlated

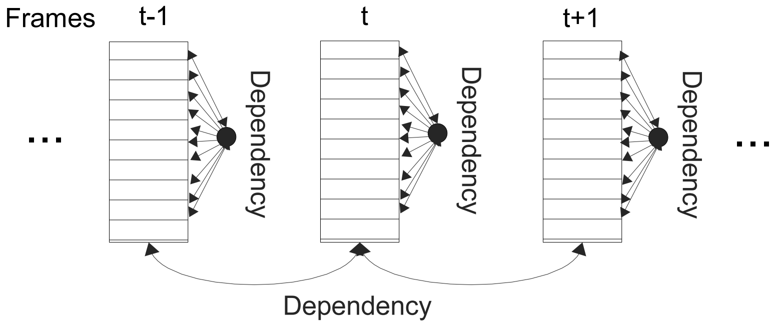


Figure 3.1: Observation feature dependencies ignored by HMMs

and pose a serious threat to the correctness of methods such as the tandem approach. The second assumption is that the features are independent inside the frame feature vector. This is not true either; but to model this correlation inside the frame using HMM would require a full covariance matrix and a huge increase in model parameters [40].

CRF overcomes this unrealistic limitation because it does not use generative probability densities of observations in each state of the model. It directly computes the posterior probability of an output label sequence. In fact it uses an exponential distribution function to compute the conditional probability of the entire output sequence given the full sequence of observations. Thus not requiring any special assumptions about the observation dependencies [18]. In Figure 3.1, the dependencies that hold between observation features are shown. They are unholily ignored by HMM modeling. The circle on the side of the feature vector and the arrows from and to all features in a frame mean that all features are somehow correlated to each other.

## 3.2 Mathematical foundations

The visual representation of a linear chain CRF used for the purpose of speech recognition in this thesis is given in Figure 3.2. One can see the linear structure of this CRF. It allows sequential steps from state $t-1$ to $t$. Each state is identically mapped with an output label. For example state t corresponds to label $Y_t$ in the output label sequence. Also the connection of output labels $Y_t$ with a frame of observation features $X_t$ is shown.

CRF models are exponential models that use functions of the input features and a trainable weighting scheme of these functions to model the phonetic units

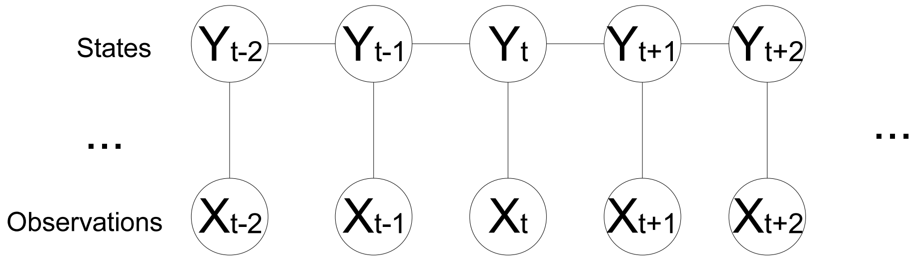


Figure 3.2: Linear chain CRF visual representation

[40]. State functions are associated with states and state features. These are used to compute the likelihood of being in a certain state. In addition, transition functions are associated with transitions between states and transition features. The posterior probability $P(\mathbf{y}|\mathbf{x})$ of a phonetic unit label sequence $\mathbf{y}$ given an input feature sequence $\mathbf{x}$ is given by:

$$P(\mathbf{y}|\mathbf{x}) \propto exp \sum_i (S(\mathbf{x}, \mathbf{y}, i) + T(\mathbf{x}, \mathbf{y}, i)) \tag{3.1}$$

where

$$S(\mathbf{x}, \mathbf{y}, i) = \sum_j \lambda_j s_j(y, \mathbf{x}, i) \tag{3.2}$$

$$T(\mathbf{x}, \mathbf{y}, i) = \sum_k \mu_k t_k(y_{i-1}, y_i, \mathbf{x}, i) \tag{3.3}$$

and $i$ is over all frames of the input sequence, j over all possible state functions and k over all possible transition functions of a CRF.

Each state feature function $s(\mathbf{y}, \mathbf{x}, i)$ is associated with a phonetic unit label and an input state feature and also has an index pointing to a position in the feature sequence. For example if we want to establish a state function for output label /aa/ (phoneme aa) and the 8-th MFCC on frame $t$ we would define:

$$f_{/\mathrm{aa}/, MFCC_8}(y, \mathbf{x}, t) = \begin{cases} g(MFCC_8(t)), & \text{if } y_t\text{=/aa/,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.4}$$

where $g()$ is a function transforming the 8-th MFCC value into an appropriate value for the implementation. Different functions can be used in place of $g()$. Usually for spectral features like MFCC the identity function is used. For phonological features created with MLPs, the posterior outputs are used [42, 40, 1].

Similarly, each transition feature function $t(y_{i-1}, y_i, \mathbf{x}, i)$ is associated with a phonetic unit transition and a transition input feature and also has an index in the feature sequence. To define a transition from label /b/ to label /ae/ associated with input phonological feature of "voiced" we have:

$$f_{/\mathrm{b}/,/\mathrm{ae}/,\mathrm{voiced}}(y, \mathbf{x}, t) = \begin{cases} g(voiced(t)), & \text{if } y_{t-1}\text{=/b/ and } y_t\text{=/aa/} \\ 0, & \text{otherwise.} \end{cases} \tag{3.5}$$

Again $g()$ function can be chosen according to the implementation as in state functions.

Trainable weights $\lambda$ and $\mu$ learn the importance of the association of each phonetic unit label or transition with the state or feature function in the final probability calculation. In the above examples, the weight of a transition feature function associating voiced attribute with transition /b/ to /aa/, should learn that the voiced attribute should be present. In another example, the weight of a state feature function associating the nasal attribute with the label /k/, should learn that the nasal attribute must be absent. The training in CRF is done using quasi-Newton gradient descent or stochastic gradient descent optimization algorithms [42, 43]. The gradient of the likelihood function must be calculated. For a set of $K$ label/observation pairs $(y_k, x_k)$ the gradient is:

$$\nabla L = \sum_{k=0}^{K} \left[ \mathbf{F}(y_k, x_k) - \sum_{Y} \mathbf{F}(Y, x_k) \cdot E_{P_\lambda(\mathbf{Y}|x_k)} \right] \tag{3.6}$$

where

$$\mathbf{F}(\mathbf{y}, \mathbf{x}) = \sum_{t=0}^{T} f(\mathbf{y}, \mathbf{x}, t) \tag{3.7}$$

is a vector of all feature functions of input sequence $\mathbf{x}$ and label sequence $\mathbf{y}$ ordered together and

$$E_{P_\lambda(y|x_j)} = \frac{exp\lambda \cdot \mathbf{F}(\mathbf{y}, \mathbf{x_j})}{Z(x_j)} \tag{3.8}$$

is the probability of sequence $y$ given $x_j$ and $\lambda$ is the vector of weights corresponding to feature function vector $\mathbf{f}$ with

$$Z(x) = \sum_{\mathbf{y}} \lambda \cdot \mathbf{F}(\mathbf{y}, \mathbf{x}) \tag{3.9}$$

the normalization value. For linear-chain CRF, there is a variation of the forward-backward algorithm calculating the gradient. More information on the training process can be found in [57].

Finally the decoding step finds the label sequence $y$ that maximizes Eq. (3.1) over input sequence $x$:

$$\hat{\mathbf{y}} = \arg\max_{y} \lambda \cdot \mathbf{F}(\mathbf{y}, \mathbf{x}) \tag{3.10}$$

| Low-level | Mid-level | High-level |
|---|---|---|
| MFCC | obstruent | /aa/ |
| MFCC delta and acceleration | silence | /ae/ |
| Degree of Voicing | sonorant | /ao/ |
| Spectral Centroid | syllabic | /b/ |
| Spectral Roll-off | ... | .. |
| Spectral Flux | tense | /zh/ |

Table 3.1: Speech events used in experiments

## 3.3 The CRF as a discriminative combination method

The primary use of CRFs in this thesis is to merge lower level attributes and features described in Chapter 2 into forming phonetic unit sequences. The phonetic units are the 61 TIMIT phonemes reduced to 48 phonemes as presented in Appendix A.

Going back to Figure 1.1 which presents the main components of the detection-based ASR system, the CRF does the job of Event Merging and Verification combined. The event merging is done with the exponential distribution modeling of the posterior probability of the output label (phoneme) sequence given the input features. The input features are provided by the two previous components of Feature Extraction and Attribute Detection. The Verification is partially done with the training of the CRF weights on a already transcribed set of speech utterances. The weights learn the correspondence of the inputs to the phonetic units. During decoding the weights are used in the exponential CRF model to find the most probable output label sequence.

The detection-based system provides speech events to the Event Merging component. The interpretation of speech event term is not restrictive and can include lower level spectral events, to higher level phonetic units presence clues. In this thesis we decided to include 3 levels of speech events, low-level speech events directly extracted from speech signal, mid-level phonological attribute events and high-level phoneme presence events. Low-level events are extracted from the speech signal using feature extraction algorithms. We have already described the MFCC and PLP computation process. Other low-level speech events used are shown in Table 3.1. The signal processing algorithms to compute these features are given in Appendix B.1. Mid-level phonological attributes were already presented in Section 2.3. High-level phoneme presence events are computed using a process similar to tandem approach: MLPs are used to compute the posterior probability of phonetic units (phonemes) given some spectral feature input (MFCC or PLP).

Different combinations of speech events are used in this thesis. Possible configurations are shown in Figure 3.3. The first configuration in Fig. 3.3a, uses phonological attribute posterior probabilities from MLPs as inputs to the CRF. The second configuration in Fig. 3.3b is a variation of tandem. But instead of HMMs, CRFs are used in the final combination stage. The third configuration in Fig. 3.3c, is using all three-level speech events, from low-level spectral-based events to high-level phoneme presence posterior probabilities. The detection of low-level events is placed in the feature extraction component of the setup. It could be placed in the event detection component, but it is basically a low-level signal processing process and has more common with MFCC or PLP extraction. Nevertheless, these low-level events (or features) are used in tandem with the mid- and high-level events detected by the MLPs. Due to the exponential distribution used in the CRF, it is adequate to concatenate the different speech event cues in a bigger vector. One restriction that must be met is that they must have a dynamic range that is similar enough. For example to combine posterior probabilities that are in the range of 0 to 1 with MFCCs, the MFCC must be mean and variance normalized.

The output of the CRF is given after the application of the exponential distribution model that is trained to give the most probable output label sequence. Most probable in the sense of highest posterior probability of the sequence - which is usually comprised of phoneme labels - given the input vector of events.

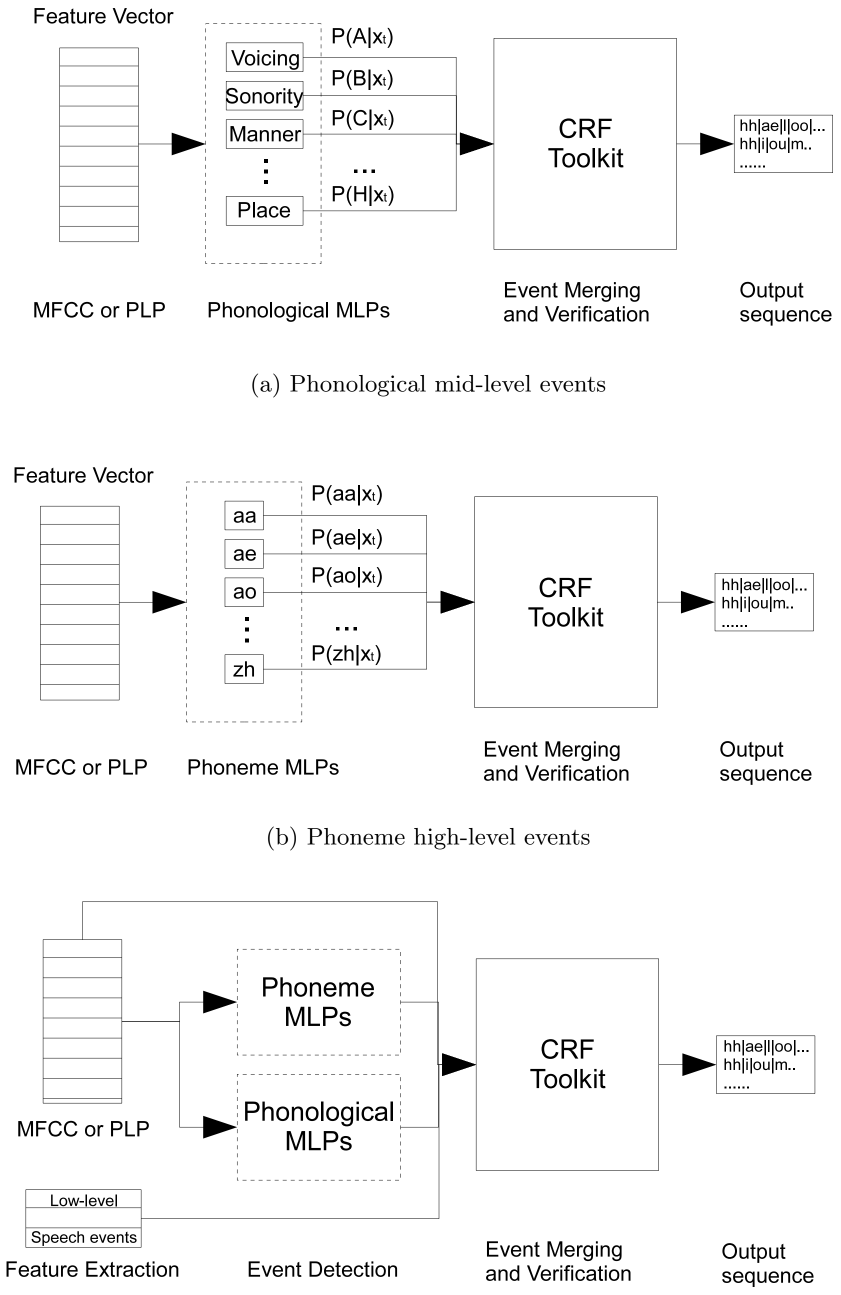
Various experiments are conducted for this thesis, using different configurations. Chapter 6 is dedicated to the presentation of these experiments and the discussion of the results.

## 3.4 Towards a complete recognition toolkit

CRFs alone cannot constitute a complete recognition toolkit. Embedded in the detection-based system are a successful component for speech event merging and partial verification. In the output of the CRFs we have a sequence of phonemes that of course is not adequate in normal ASR systems. Further merging and verification is necessary to recognize speech units that are higher in the speech hierarchy, such as words and sentences. There is no solid solution at the moment. An approach has been proposed recently in [14]. More details are given in Chapter 7.

In this chapter and Chapter 2, the detection-based system used in this thesis, has been presented in detail. The components of the system including the inputs and the outputs were given. Each component is based on a technology that was analyzed and discussed and no black boxes were used. We can now use the detection-based system as basis and try to achieve recognition performance improvements. Towards this goal, we will:

(a) try different speech event detectors and combinations



(a) Phonological mid-level events

(b) Phoneme high-level events

(c) Various level events

Figure 3.3: CRFs combining different speech events

(b) extend the system to support the modeling of speech rate variability

(c) extend the system to support the inclusion of speech segment information

Goal (a) can be achieved with the already presented detection-based system setup. Experiment details, results and discussion are presented in Chapter 6. To support the other two goals, some modifications on the detection-based system will be needed. Chapter 4 is dedicated to the analysis of the theoretical foundations of speech rate variability and how it can be addressed. Also the modifications to the detection-based system are given. Chapter 5 describes the segmental approach to speech recognition and how we can benefit by including some of its concepts into the detection-based system.

## Chapter 4

# Modeling the variability in speech rate with multi-scale analysis

In this chapter, the detection-based system is modified in order to support modeling of speech rate variability inside the utterance. First the speech rate variability is studied in general and a theoretical supportive base is established. Then, an approach to model the inside the utterance speech rate variability is introduced and discussed. This approach uses multiple temporal scales during various phases of the recognition system, from speech signal processing to viterbi decoding. The necessary modifications to the detection-based system components are discussed. Finally, the problem of multiple scales integration to form the final output hypothesis is discussed and novel approaches developed for this thesis are introduced.

## 4.1 Modeling the speech rate variability

Speech production is a dynamic phenomenon and the variability is inherent in various aspects of the process. The speech rate is such a variable attribute that must be taken into consideration when designing ASR systems. This variability can be caused by the different spectral characteristics of speech during different pronounced phone classes. This means that we can have stationary segments of speech signal that can reach 100 msecs during some vowels. However, we can have drastically changing spectral contents in the scale of a couple of milliseconds during stop consonants and phoneme transition segments. Another factor that causes rate variability is the speaking rate variability that is very common in conversational speech. Speaking rate can change between speakers or inside the utterance. It affects various aspects of speech such as co-articulation and reduction, causing even more non-stationary segments and confusion.

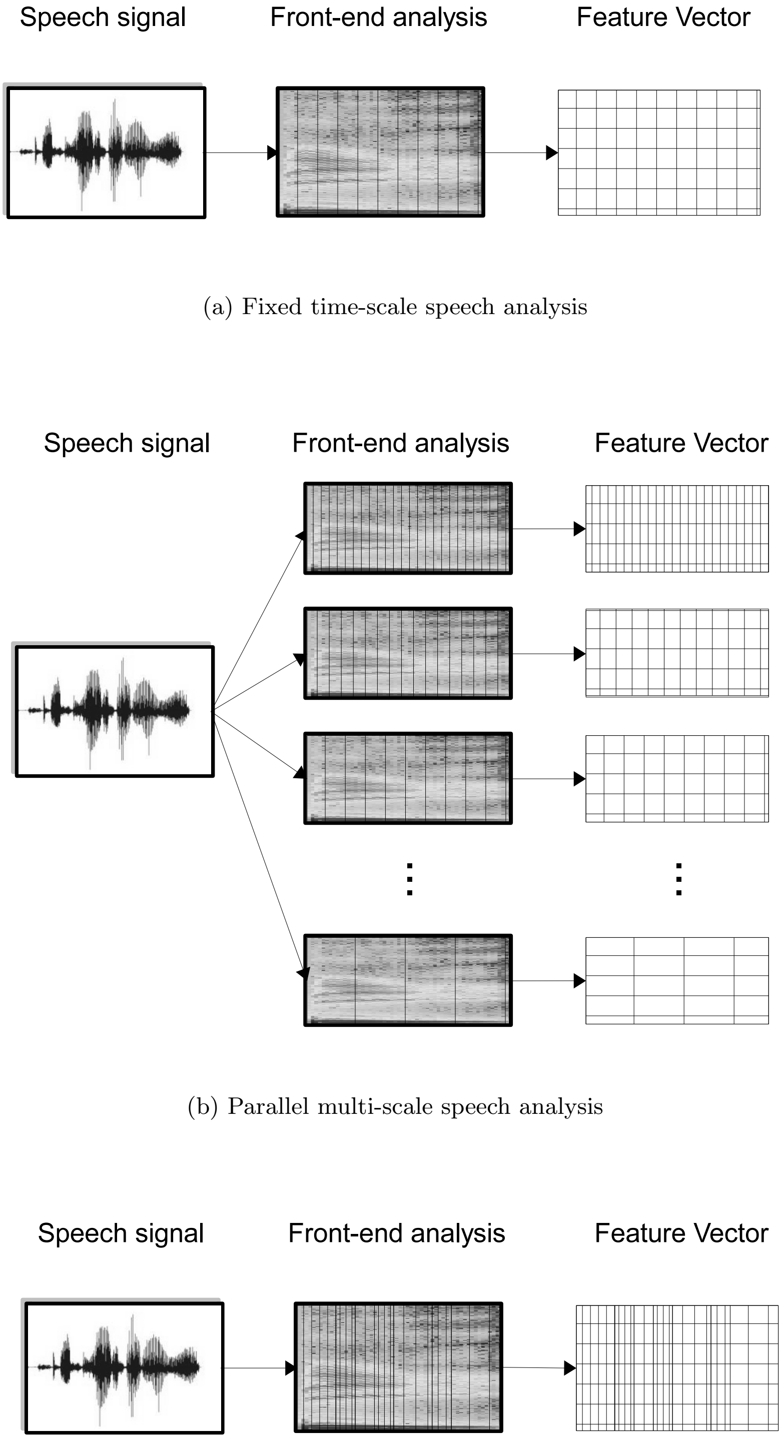
Currently, different approaches have been proposed to deal with the variability of speech rate. Seen from a speaking rate perspective, different acoustic models are trained for each supported rate of speech [63]. Usually two variants of models are created: one for slow and the other for fast speech and speakers. The rate selection can be set at the utterance or word levels. At the utterance level, a rate of speech (ROS) estimator is used to classify the sentence as slow or fast speaking. Then a system that uses either slow or fast models is fed with the corresponding utterances. A more robust method is selecting the rate at the word level. Each word is given two pronunciations, a slow and a fast one by using appropriate rate-specific sub-word units (phonemes). Then the selection of the optimal pronunciation is left to the decoding algorithm. This method does not need any pre-recognition rate-of-speech classification. Another perspective to look at the speech rate variability issue is the variability in the spectral contents in time [15]. During stationary segments we have a slow rate of change. This fact permits the signal analysis to be made in longer segments, because no significant change in spectral content happens. But during transitional segments, we have fast changing spectral content that needs to be extracted. In this case, short segments must be used during the analysis of speech in order to conserve fast changing information. Otherwise the information will be lost, as it will be smoothed out by the adjacent stationary segment contents. Introducing multiple time scales in the analysis of speech is a proposed solution to the problem of variability of speech rate of change as described. Recent work on this area uses various modifications and tweaks in the already developed set of recognition tools to support the multiple time scale concept. In [19], the multi-stream HMM framework is utilized to incorporate multiple time scale information as independent streams. Independence assumption is faulty assumed once again as described in Section 3.1. In another work [15], the rescoring of N-best lists is done by a phone-dependent posterior-like score. Phoneme clustering is done in 11 clusters based on the characteristics of each phoneme. Then using the mapping of phonemes to clusters, a selection of the appropriate rate model of a phoneme for rescoring is done. In yet another work [64, 62], the selection of the appropriate rate of analysis is done using rate of change metrics. The weighted Euclidean distance of consecutive MFCC vectors is used as a metric to select the optimal rate of analysis from a predetermined set. In practice, the rate of analysis is determined by a frame picking algorithm that is based on hard limits of the metric value. Another metric, used in the same work, is the entropy of the speech signal. To determine the speech rate, again hard limits on the entropy value are used.

It is obvious that there are different approaches to model speech rate variability, depending on the point of view. In this thesis, we concentrate our efforts into tackling speech rate variability inside the utterance. More specifically the rate of change of speech is estimated using statistical techniques and consequently the rate of analysis is determined. In the following sections of the chapter, details of the approach followed are given: from theoretical conception to actual implementation using the detection-based system.

## 4.2 The idea of multi-scale analysis

Speech events and phonemes can have shorter or longer duration. In a speech recognition system, it is necessary to detect as many events as possible and recognize phonemes of different spectral contents and duration. The analysis front-end of speech signal was given in Chapter 2. It was shown that the analysis is performed in window frames. Each window has length that is defined in milliseconds and usually all windows have the same fixed length in a recognition system. Also another parameter is the frame-rate of analysis, meaning the step at which windows are applied on the speech signal. These two parameters are important because they set the time scale at which the speech signal is processed and thus the potential speech events and attributes that can be successfully detected. Current trade-off sets the window size at 25 msec and the frame-rate at 10 msec.

One can guess that by including in the system multiple time scales, the potentially detectable speech events and recognizable phonemes are increased. Based in this assumption and the knowledge of the inherent variability of speech events in time, multiple time scale methods are tested in this thesis. These methods can be seen as an extension to the current trade-off to include information from other time scales. In Figure 4.1a, the analysis of a speech utterance with fixed windowing is shown. Then in Figures 4.1b and 4.1c, two approaches for inclusion of multi-scale information are shown: first a multi-scale analysis of the speech signal is run in parallel and second a variable scale analysis is used. The parallel multi-scale analysis is quite obvious in conception, in the sense that different and independent front-ends are used to analyze the speech signal. Each front-end has its own scale of processing the speech signal and the features that are extracted contain information of events at that scale. The variable scale analysis task is more complex. Switching between different time-scales is done inside the utterance at various points in time. The most appropriate scale of analysis for a certain segment of speech is selected and used. Here emerges the problem of finding the most appropriate scale of analysis, i.e. the time-scale that gives the best performance when the extracted features are used to either detect speech events or recognize phonemes. An objective method must be used to for this purpose. As already described in Section 4.1, the task of selecting the appropriate scale or rate of analysis has been achieved by using a clustering of phonemes, a spectral distance metric and signal entropy metric. In this thesis, a classifier is used



(a) Fixed time-scale speech analysis

(b) Parallel multi-scale speech analysis

(c) Variable scale speech analysis

Figure 4.1: Speech front-end processing with speech rate variability awareness

to select the appropriate time-scale. The process of selecting the most efficient time-scale for a segment of speech is approached as a multi-scale combination task. More details are given in Section 4.4.

## 4.3 Multiple temporal analysis in a detection based system

The detection-based system presented in Figure 1.1 and described in Chapters 2 and 3, supports single time-scale analysis of speech and subsequent processing. The feature extraction component works in a single time-scale and provides the attribute detectors with single scale features. The attribute detection component detects events on a fixed time-scale, although the events can expand on multiple frames in time. The event merger and verification work also in a fixed time scale, combining and verifying frame-based events. In this section, the necessary extensions to the implementation of the detection-based system are given, in order to support multiple time-scale analysis; we can call it multiple time-scale detection-based speech recognition. In fact, extending the detection-based ASR into supporting multiple time-scales comes in naturally. Speech events tend to be unstable and variable in length, so incorporating into the system as much information as possible is a desirable feature.

Feature extraction component is the first that should be modified. In Figure 4.1 is already shown a speech signal front-end that supports multiple time-scales. Getting into a detailed description, and having in mind what was presented in Section 2.1, Figure 4.2 shows an MFCC implementation extracting speech parameters at multiple time scales. Different FR and WS pair values are used to create multiple size windows and multiple step sizes. The commonly used trade-off values are $FR = 10msec$ and $WS = 25msec$. In this thesis, information from time-scales that are nearby the trade-off value are used. We want to catch shorter as well as longer possible speech events, so we use the following pairs:

- $FR = 2.5msec$, $WS = 6.25msec$
- $FR = 5.0msec$, $WS = 12.50msec$
- $FR = 7.5msec$, $WS = 18.75msec$
- $FR = 10.0msec$, $WS = 25.00msec$
- $FR = 15.0msec$, $WS = 37.50msec$

By processing the speech signal in parallel, we get 5 streams of information that are highly redundant. The stream using the smaller pair values has fine details of spectral content, though it is unable to extract longer spectral patterns. As the pair values increase, the fine details of the spectrum are smoothed out and

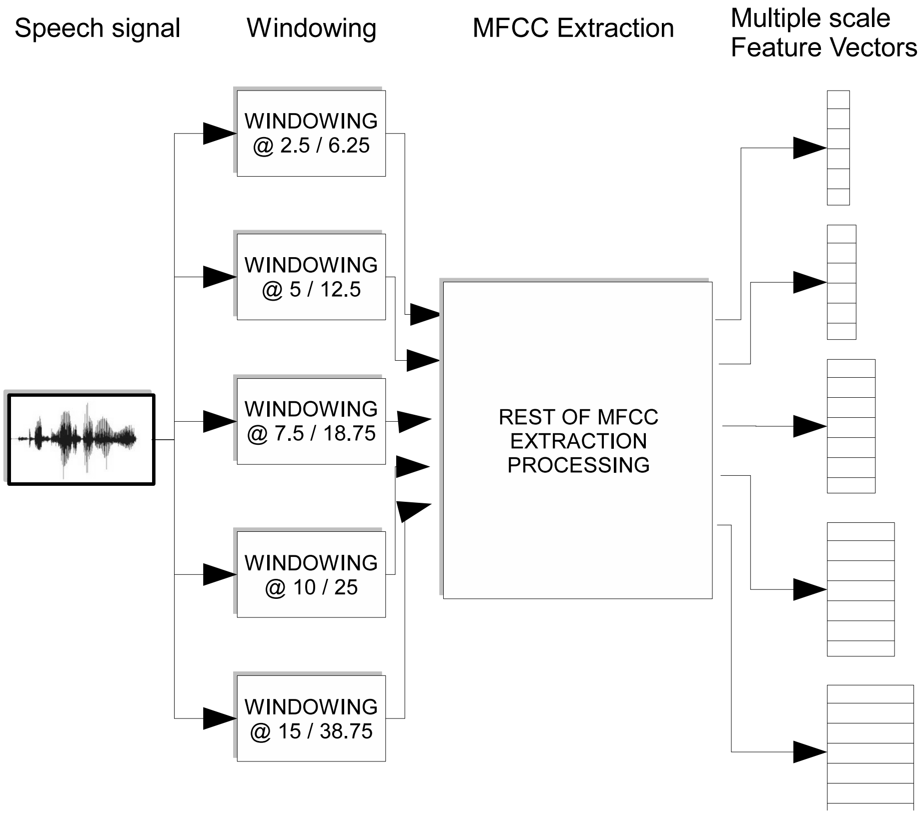


Figure 4.2: Multiple time-scale MFCC feature extraction

longer patterns are extracted. The extracted speech parameters are used as input features to the next component in the detection-based ASR setup.

Figure 4.3 shows the event detection on multiple time scales. The approach used in this thesis expects speech events to have presence at different time-scales. Being not quite sophisticated, does not restrict a speech event to a specific time-scale. It is assumed that a speech event can be present at any of the available time scales. Speaking optimistically, this adds considerable effort and redundant information to the system. The issue is planned to be resolved in the future as stated in Chapter 7. The actual MLPs, used for detection, don't need any special modification and they follow the structure shown in Figure 2.3. The difference is in the rate of the input vectors, so the transcription labels must use a similar time scale during training and detection.

The next component of the detection-based system is the event combination. Multiple time-scale support for this component is a more complex task and is described in Section 4.4.

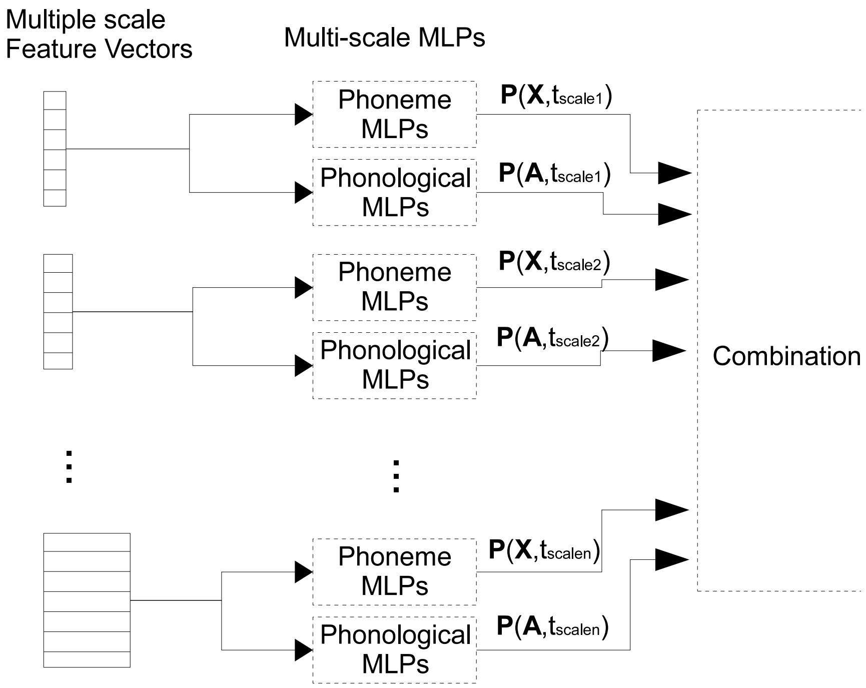


Figure 4.3: Detection of speech events at multiple time-scales

## 4.4 The problem of multi-scale combination and solutions

Up to this point, a detection-based system which supports speech events at different time-scales has been introduced. These multi-scale events must somehow be combined and used. In Chapter 3, CRFs have been proposed as a combination and partial verification toolbox of speech events. In this section, a couple of methods to combine the multiple time-scale streams will be proposed in conjunction with the CRF toolbox. The problem in general is shown in Figure 4.4. In this thesis, the combination is done before giving the speech event lattice to the CRF toolbox for final integration.

The first method we use is quite straightforward. It takes speech event presence vectors computed at different time scales and combines them all together. It is better described in Section 4.4.1. This method does not use any indicator of the most appropriate time-scale. In contrast, the other method used in this thesis, does a selection of a time-scale and ignores the other scales. The information that are left out are considered either redundant or at best confusing. The worst case is when useful information that are not present in the time-scale elected as optimal, are left out. The optimal scale indicator is used to select the optimal time-scale for a particular speech segment, word or whole utterance. The realization of the optimal scale indicator, an indicator of the time-scale which when

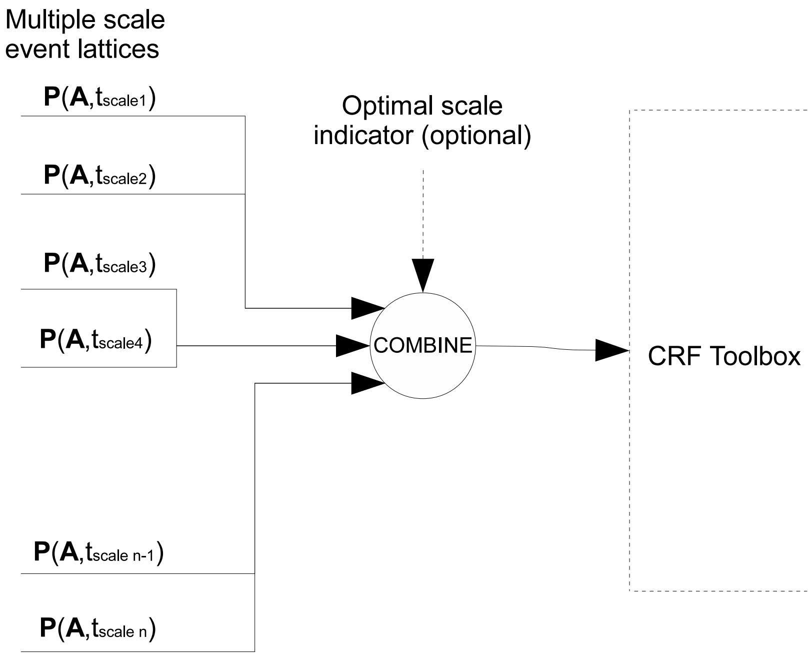


Figure 4.4: Multiple time-scale combination in general

used gives the best overall performance in the recognizer is a problem studied in Section 4.4.2.

### 4.4.1 All-inclusive method

This combination method is quite straightforward in its implementation. It is mostly a concatenating feature combination method [48]. Every detected event at any time-scale is considered equally important to any other. In Figure 4.5, the method is shown as a visual representation of a box of features [9]. This box includes features from different time-scales and is a concatenation of the latter. In fact, the box is a feature vector and the sequence of these vectors is given the frame-rate of the largest time-scale included. By taking the frame-rate of the largest time-scale, we manage to have a fixed-rate system that includes speech events from all possible time-scales.

The most important drawback of the box method is the large dimensionality of the vector created. It adds to the overall complexity of the subsequent recognition process. Another drawback is the redundancy of the vector components and their high correlation. A de-correlation and dimensionality reduction step can be used to compensate for these issues. Also by using CRFs instead of HMMs, we have shown that these drawbacks are scaled down due to the CRFs ability to efficiently integrate highly correlated features [9].

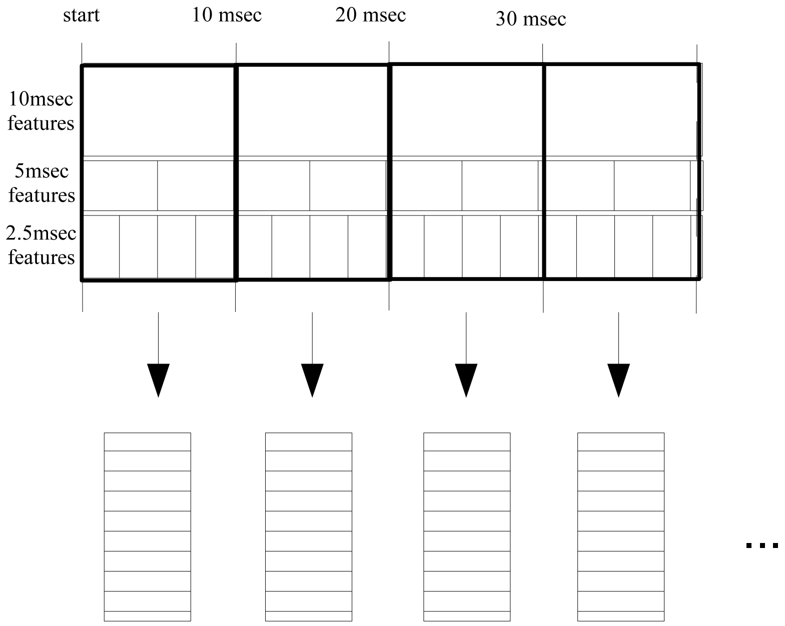


Figure 4.5: The all-inclusive box combination method

### 4.4.2 Optimal scale selection method

A more sophisticated method does not use events from all time-scales, but selects an optimal scale in the sense that maximizes the final recognition performance. Switching time-scales can be done in a speech frame level, segment, word or utterance. In this thesis, the optimal time-scale selection in a speech segment level has been investigated. A 3-frame segment has been chosen, and switching time-scales was allowed when moving from a segment to the next. The main problem is to find an indicator that would connect the recognition performance of a time-scale with a metric showing the rate of speech in a segment. This would enable a selection process to choose the optimal time-scale for a segment. The method used in this thesis and in [9] is shown in Figure 4.6.

The system works in a closed-loop in order to select the best frame rate for each temporal segment of speech. The main unit of interest is the Frame-Rate Selection Unit (FRSU ) which is trained to select the best frame rate for a specific segment of speech. The training of the FRSU's parameters is done using ML or Minimum Classification Error (MCE ) discriminative training method [25, 26, 27].

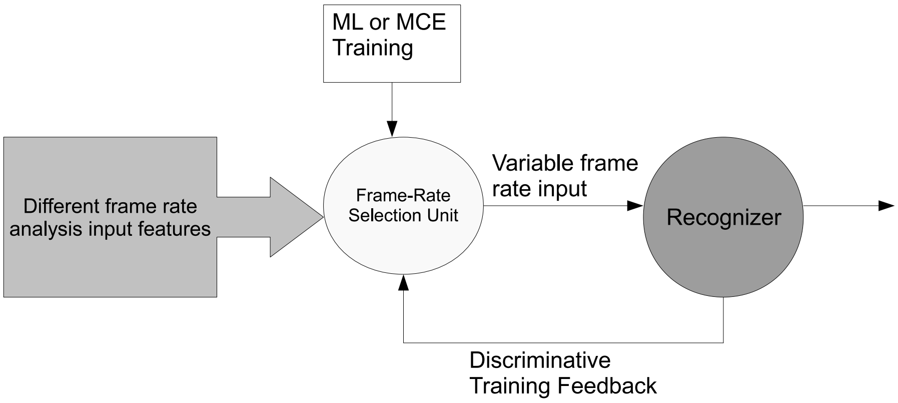

ML or MCE Training
Different frame rate analysis input features
Frame-Rate Selection Unit
Variable frame rate input
Recognizer
Discriminative Training Feedback

Figure 4.6: Variable scale selection method

### Spectral Change Metric

First task it to compute a metric for each subspectral region of a segment of speech signal that indicates the rate of change for that spectral region. By using spectral regions we include information about the rate of change of different regions. The combination of these metrics forms a global spectral distance metric that models the rate of change of speech. The rate of change can be computed from a Fast Fourier Transformation (FFT) analysis and then taking the first-order time difference for each region. The combination of these partial spectral change metrics benefits from a non-linear combination method such as the product rule with weights that can be trained with a ML or MCE method. Non-linear method can detect changes in minor spectral regions which indicate a transition segment compared to a rather smoothed result of a linear combination method. The global spectral distance is computed by the equation:

$$D = \prod d_i^{w_i} \tag{4.1}$$

with $w_i$ the trainable weights and $d_i$ the distance for the i-th spectral region. The weighting parameters can be trained in a first-pass MCE training of the FRSU unit.

### Mapping to Rate Of Speech

Next is to map the computed distance metric to the optimal time-scale pair values (FR,WS). The mapping function of choice is a sigmoid with a few parameters that can be trained in a second-pass MCE training phase:

$$ROS = a + \frac{c}{1 + e^{-b(D+d)}} \tag{4.2}$$

with ROS meaning the Rate Of Speech, D the global spectral distance computed above and a,b,c and d trainable parameters learning the non-linear mapping from the spectral change distance metric to the optimal rate of speech. After these two processing steps, we have a continuous function of the rate of change of speech. This can be used to select the appropriate FR/WS pair for each speech segment.

### Training of parameters using an MCE method

Now that we have described the transfer function of the FRSU unit, we can proceed in describing the learning process that we use to train the free parameters on Eq. (4.1) and (4.2). An obvious solution to the learning process is the ML estimation of the parameters. This estimation method is used as a baseline in this thesis. Given the ML estimated model parameters, MCE training method is used to improve the parameter estimates.

The task of MCE training can be divided into three main steps:

a) Choose a discriminant function for the description of the classification task.

b) Create a misclassification measure to express the classifier decision process.

c) Form a cost function that would be an indicator of success of the classification.

All the above quantities must be continuous and differentiable with respect to the estimated parameters. The classifier can be designed to have a simple discriminant function of the form:

$$g_j(X; \lambda) = |ROS_j - ROS_X| \tag{4.3}$$

with $ROS_j$ indicating the j-th Rate Of Speech prototype value, and $ROS_X$ the Rate Of Speech Metric computed as shown in Sections 4.4.2 and 4.4.2. With the previous formulation of the discriminant function, we can state the decision process of the classifier as:

$$C(X) = C_i, \quad \text{if } g_i(X; \lambda) = min_j(g_j(X; \lambda)) \tag{4.4}$$

The next step in MCE formulation is the definition of a class misclassification measure, which in fact expresses the decision rule in Eq. (4.4) in a functional form [25]. We choose the rather frequently used measure:

$$d_j(X; \lambda) = g_j(X; \lambda) - \left[ \frac{1}{M-1} \sum_{k,k \neq j} g_k(X; \lambda)^{\eta} \right]^{\frac{1}{\eta}} \tag{4.5}$$

with $\eta$ a positive smoothing constant and $M$ the number of classes. When the misclassification measure is way below zero, this indicates a correct classification. Instead when it is positive, it indicates an incorrect classification.

After we have defined the misclassification measure, we create the cost/loss function. The function must be continuous and indicative of success/error rate of the classification . We choose the sigmoid mapping function which is bound between 0 and 1:

$$l_j(X;\lambda) = \frac{1}{1 + exp(-\gamma d_j(X;\lambda))}, \gamma > 1 \tag{4.6}$$

with $\gamma$ the scaling factor of the sigmoid. This loss function is a smooth and continuous measure of success of the classification task. When sample $X$ is correctly classified then the misclassification measure decreases way below 0 and the loss function approaches 0. When it is incorrectly classified the misclassification measure indicates the level of failure and the loss function approaches 1. Now we can evaluate the performance of the classifier on an unknown sample $X$ using the following smooth function:

$$l(X;\lambda) = \sum_{i=1}^{M} l_i(X;\lambda) 1(X \in C_i) \tag{4.7}$$

where 1() is the indicator function and is 1 when sample X belongs to class i else is 0.

The next concern is a minimization method for the expected loss of the classifier during training, in order to estimate the appropriate values for the free parameters on Eqs. (4.1) and (4.2). We want to minimize the expected loss which is:

$$L(\lambda) = E_X\{l(X;\lambda)\} = \sum_{i=1}^{M} \int_{X \in C_i} l_i(X;\lambda) p(X) dX \tag{4.8}$$

with $X$ summing over all samples of a training set. We use the GPD algorithm with parameter space transformations in order to impose constrains on the free parameters [25]. In practice we minimize the empirical loss assigning equal probability mass to each sample. The empirical loss will converge to the expected loss if a training set of sufficient size is used. The general update equation of the parameter set we are training ($\lambda$) at a given iteration of the process (t) is:

$$\lambda_{t+1} = \lambda_t - \varepsilon \nabla l(X;\lambda)|_{\lambda=\lambda_t} \tag{4.9}$$

with $\epsilon$ the learning coefficient. We can use a 2-pass training procedure. In the 1st pass, keep the parameters of Eq. (4.2) constant and train the parameters of Eq. (4.1). In the 2nd pass use the values found during 1st pass and train the parameters on Eq. (4.2). As an example of MCE/GPD iterative update, the update equation for parameter $b$ on Eq. (4.2), when sample $X \in C_i$, is given:

$$b_{t+1} = b_t - \varepsilon \frac{\partial l_i(X;\lambda)}{\partial b} \tag{4.10}$$

with

$$\frac{\partial l_i(X;\lambda)}{\partial b} = \frac{\partial l_i}{\partial d_i} \cdot \frac{\partial d_i}{\partial g_i} \cdot \frac{\partial g_i}{\partial ROS_X} \cdot \frac{\partial ROS_X}{\partial b} \tag{4.11}$$

A similar partial derivative chain rule can be used in order to derive the update equations of the other parameters of the FRSU module.

## Chapter 5

# Segmental Processing of Speech

In this chapter, an introduction to segmental speech recognition is given. The various problems of this approach are presented and special insight into the segmentation task is given. Although this approach is not fully followed in this thesis, some important ideas are adopted in the detection-based system.

## 5.1 Segmental speech recognition

The detection-based ASR system already described, uses frame-based analysis and observation modeling. Although it departs from the standard HMM frame-based techniques, still uses states and observations associated with them. Segmental speech recognition models larger units than frames. The start and end times of a segment would ideally coincide with the boundaries of a phonetic unit. Thus a segment analysis and feature extraction would give the maximum available information of a phonetic unit. Subsequent recognizer would benefit from this rich information in order to identify better the underlying phoneme. The main advantages of the segmental method over the standard frame-based are:

- Unrealistic assumptions about the independence of the features are unnecessary [47],

- The duration of the phonetic unit can be a useful feature in the recognition process [47],

- Transitions between phonemes contain useful information which can be included in segment boundaries [4].

All three issues have been addressed in the HMM frame-based system with some extensions. We have also seen that the first problem is not relevant in a detection-based system using CRF modeling. The other two are good candidates for implementation, as extensions, in the detection-based system.

Before moving on to the actual implementation, further details of the segmental approach are necessary. Some major concerns when using segments are:

**Segmentation** The first and most important step is the segmentation of the utterance into speech segments. It affects all subsequent steps of the process. The segments must encapsulate the phonetic unit position and duration as accurately as possible. An effective segmentation reduces the search space of the decoding step as less paths must be accounted for in the graph of segments. Usually an acoustic segmentation algorithm is used to define the segment graph of an utterance. Poor alignment, insertions and deletions of speech events are a common issue of a segmentation algorithm. These errors cannot be corrected in subsequent steps and lead to recognition errors. A trade-off between performance and computation must be established [4]. More segments mean less errors of the kind described above, but more difficult task for the search algorithm. On the other hand, less segments don't create much burden during searching on the graph, but increase recognition errors. Section 5.2 is dedicated to the segmentation process and the implementation used in this thesis.

**Decoding** The decoding step is a search on a probabilistic framework for the best path. Standard dynamic programming techniques are used that were described in Chapter 2. The difference is that now the sequence of states and models is substituted by a segment sequence. So the most probable segment sequence for an input utterance is computed. Each segment has its own feature vector, so each sequence of segments accounts for a specific set of vectors. It has been proved that it is necessary that a path in search space must account for all feature vectors [17]. To overcome this restriction, a series of extensions have been developed in segmental modeling, Anti-Phone modeling and the Near-Miss modeling [4]. They create segment models that include feature vectors of segments that are off the current search path. In this thesis, we don't follow the segmental approach in searching. Instead we use the CRFs ability to process external segmentation information as transition features. More details are given in Section 5.3.

**Modeling** Another important issue is the model used to represent the segments. Although HMMs are allowed in segment modeling, they are not optimized for segmental processing. Usually models that have more general distributions are used. They must have the ability to explicitly or implicitly model the feature dynamics [47]. Also another possibility is to model the transitions between segments, instead of the segment themselves. This approach creates transition models between probable phonetic units.

After describing the major issues and hurdles of segmental processing of speech, we will concentrate on the detection-based system. The segmental approach is not fully followed in this thesis, but some knowledge is adopted and used to extend the ability of the detection-based system. The work done in this thesis is considered an initial approach to implement segmental processing abilities to the proposed detection-based system.

## 5.2 Locating the phoneme boundaries: Transition features

The first concern is the segmentation. An effective segmentation requires an accurate detection of the phoneme boundaries. Low quality segmentation leads to poor alignment, insertion and deletion of speech events. We investigate different features and implementations in order to achieve a successful segmentation of the speech utterance.

### 5.2.1 Spectral and energy domain

A common feature for the task of segmentation is a measure of spectral change. This feature was already used, as described in Chapter 4, to adapt the detection-based system to the rate of change of speech signal. In [28], a similar feature set was used for automatic segmentation in a speech synthesis application. In this thesis, the Mel-Scale Spectral Magnitude was used to compute the spectral region differences of 20 filterbank channels. Then these sub-spectral differences were combined using the product rule with equal weights. The spectral change metric for a frame of speech $i$ is computed by the equation (average over three frames):

$$D(i) = \frac{\sum_{j=i-1}^{i+1} \prod^{K} d_j(k)}{3} \tag{5.1}$$

where $d_j(k)$ the distance for $j$-th frame and $k$-th spectral region, and $K = 20$ is the number of spectral regions used.

A similar feature is the Spectral Flux (Fss) [33]. The Spectral Flux is the difference between the amplitudes of successive magnitude spectra. This feature was also used as a speech event in CRF event merging of Section 3.3. Details about its computation are given in Appendix B. A smoothed flux measure is derived by averaging over neighboring flux measurements:-

$$F_{ss}(i) = \frac{\sum_{j=i-1}^{i+1} F_{ss}^{0}(j)}{3} \tag{5.2}$$

Another important group of spectral features is the Spectral Centroid difference, the Spectral Roll-off difference and the Zero-Crossing Rate difference. The Spectral Centroid and Spectral Roll-off were also used as speech events. The Spectral

Centroid is the frame-to-frame difference of the center of mass of power spectrum. The Spectral Roll-off is the frequency below which the 95% of the power spectrum is concentrated. Finally, the Zero-Crossing Rate is the rate of sign changes (positive to negative and back) of a signal and can be computed in the time-domain. Computation details are given in Appendix B. The (smoothed) time difference of each of these features was used as follows:

$$X_{ssd}(i) = \frac{\sum_{j=\{1,2\}} X_{ss}(i+j) - X_{ss}(i-j)}{2} \tag{5.3}$$

where $X_{ssd}()$ is the difference feature and $X_{ss}()$ one of the previously described spectral features for frame $i$.

Finally, the frame-to-frame Energy difference of the signal was also used for boundary detection. The same regression formula was used as in Eq. (5.3).

To evaluate the proposed features, a linear classifier which worked as a detector/rejector of transition regions was used for each feature. Adjusting the operation point, for the boundary detector the optimal hits to false positive ratio is reported, while for the boundary rejector the optimal true negative to miss ratio is reported in Table 5.1. Overall, spectral change and spectral flux perform better as rejectors of frames as possible transitions, especially for frames in the same phonetic class, e.g., STOP ⇒ STOP. Spectral Centroid difference, Spectral Roll-off difference, Zero-Crossing Rate difference and Energy difference are better detectors. Best results were obtained for transitions between VOW ⇒ {s,z}, {s,z} ⇒ VOW and n ⇒ {s,z}

| Features | Detector Ratio | Rejector Ratio |
|---|---|---|
| Spectral change | 2.56 | **5.78** |
| Spectral flux | 1.31 | **14.45** |
| Spectral centroid diff | **6.72** | 2.35 |
| Spectral roll-off diff | **4.21** | 2.94 |
| Zero crossing rate diff | **10.16** | 2.03 |
| Energy diff | **3.53** | 1.24 |

Table 5.1: Spectral and Energy feature evaluation

### 5.2.2 Phonological and MFCC deltas

Phonological features have long been used to describe whether phonological attributes of segments, such as the consonant manner, place of articulation and voicing, sonority, or vocalic attributes, are present within a speech frame ([41, 29] inter alia). These attributes are associated with a group of phonetic units; each unit can be thought of as a bundle of features. The relationship between segment boundaries and phonological features can be complex: while some features can extend across boundaries (as in the nasalization of vowels), many features will transition in unison at segment boundaries. The degree to which features transition in concert depends particularly on the type of segmental transition.

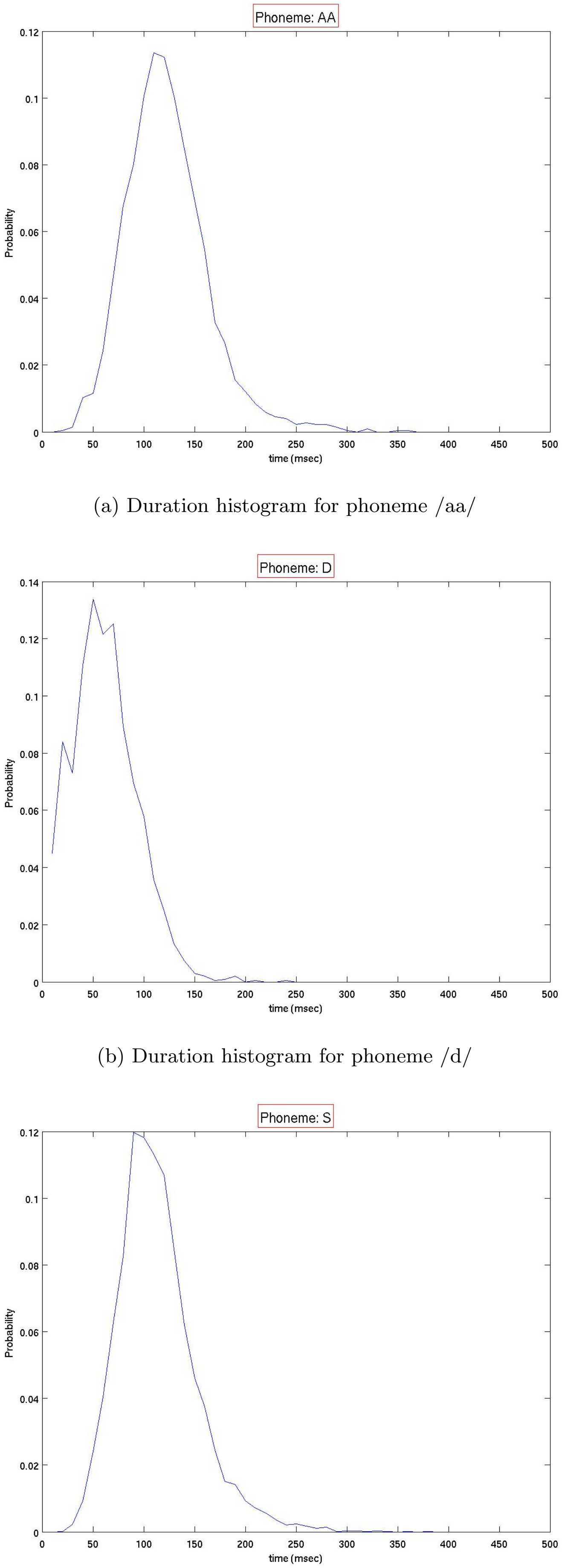
Deltas of phonological features can be used to estimate the rate of change of the phonological attributes: High values of phonological deltas indicate a phonological attribute transition. We used phonological deltas as transition features, accepting the risk that we will have a small increase in false positive detection of boundaries (and consequently in the insertions of the final recognition task). In Figure

The common MFCC feature vector was computed with a frame-rate of 10 msec and a window size of 25 msec. MFCC deltas is a commonly used group of features that estimate spectral change in time, in the transformed cepstral domain. MFCC deltas were also used as transition features.

## 5.3 Segmentation and Modeling

The segmentation of the speech utterance creates the graph of segments. Two segmentation families are used, the acoustic segmentation or the probabilistic segmentation. In an acoustic segmentation, transition features extracted from the speech signal are used. After having a set of features that give exclusive information for the phoneme boundaries and thus for the segmentation, one would proceed to define the actual segmentation algorithm. This would include either a peak locating algorithm in the transition feature curve, or a phonetic boundary classifier using these features. This would create a segmentation of the speech utterance and thus the creation of a graph of segments. Acoustic segmentation, tends to hypothesize more segments than necessary, although this factor can be tuned by certain parameters. In a probabilistic segmentation, a first pass decoding of the utterance, by a frame-based recognizer, is necessary. The segment boundaries are defined by combining the N-Best path information from the first pass. This method creates segmentations characterized by good quality, but suffers from computational burden from the two step recognition. In [34], some efforts are presented towards creating a real-time probabilistic segmentation algorithm. In this thesis, there is no need to define an actual segmentation algorithm, as the transition features are used directly in the CRF model formulation. Due to the ability of the CRF model to accept transition features, tied with transition functions, the task of finding the most probable segmentation is left to the recognizer. Also there will be no need to exclusively model the segments for the same reasons. This can be seen as a segmental processing system combining frame-based scores and segmentation in one modeling framework. Nevertheless, some basic segmental modeling techniques will be presented.

The general segment modeling framework is a joint model for a random-length sequence of observations generated by a phonetic unit. For instance, if the



(a) Duration histogram for phoneme /aa/

(b) Duration histogram for phoneme /d/

(c) Duration histogram for phoneme /s/

Figure 5.1: Duration histograms in TIMIT dataset

observation sequence is $\mathbf{y^l} = [y_1, ..., y_l]$ and the phonetic unit $a$ then the density is:

$$p(y_1, ..., y_l, l|a) = p(y_1, ..., y_l, a)p(l|a) = b_{a,l}(y_1^{\prime})p(l|a) \tag{5.4}$$

where $l$ is the length of the segment The main characteristics of the segment model are the duration distribution $p(l|a)$ and the family of observation densities that describe observation sequences of different lengths $\{b_{a,l}(y_1^l)\}$.

The duration distribution is usually modeled by a Poisson distribution or a Gamma distribution. In Figures 5.1a to 5.1c, histograms of the duration of three phonemes are presented, /aa/, /d/ and /s/. We have computed the average duration of a phoneme in the TIMIT database and found it to be 87.9 milliseconds. If the segments are phone-sized then any reasonable assumption about the distribution works well, because the contribution of the duration model is small relative to the segment observation probability [47].

The segment is divided in separate regions in time. Inside these regions the distribution parameters of the observation densities remain invariant, making it look similar to an HMM state. Also a distribution mappings collection associates each observation with one of the model regions. These two factors constitute a means of specifying the family of observation densities for a segment. The mapping can be either static or dynamic. In the first case, the computation is lower and is adequate for phoneme-sized segments. The dynamic case requires trajectory sampling and is more sophisticated as it does not assume piecewise constant dynamics. An example is given on how to compute the probability of a phoneme sequence, given the observation sequence, using segmental speech recognition. If the phoneme sequence is $a_1^N$ and the observation sequence is $y_1^T$ then:

$$p(y_1^T|a_1^N) = \sum_{l_1^N} p(y_1^T|l_1^N, a_1^N)p(l_1^N|a_1^N) \tag{5.5}$$

$$= \sum_{l_1^N} p(y_1^T|l_1^N, a_1^N)p(l_1^N|a_1^N) \tag{5.6}$$

$$= \sum_{l_1^N} [\prod_{i=1}^{N} p(y_{t(i-1)+1}^{t(i)}|l_i, a_i) \prod_{i=1}^{N} p(l_i|a_i, l_{i-1}, a_{i-1})] \tag{5.7}$$

where $t(i)$ is the ending time of the i-th segment, and $l_i = t(i) - t(i-1)$ is the segment length.

The highest benefit when using segmental models compared to the HMMs is achieved when using a modeling structure to take into account the feature dynamics. A variety of distribution families can be used for this purpose. When using distribution regions inside a segment model, then the model is called constrained mean trajectory. The trajectory can be either parametric, so the mean is specified by a polynomial, or non-parametric, so the distribution parameters are estimated separately for each model region. Another type of modeling is done using a stochastic, linear dynamical system. Without getting into much detail, the modeling for a multi-region segment model is done by defining a system parameter set for each region of each model. The assumption that must hold is that the system parameters are time-invariant within the region of definition.

Other modeling techniques can be used, such as Conditionally Gaussian Models, Non-Linear Models and Segment-Level mixtures. More details can be found in [47].

## 5.4 Segmental processing in the detection-based system

The approach, used in this thesis, is to keep the frame-based CRF model and augment the system with transition-based modeling through appropriate features [8]. The benefit of including transition information into the state models has been recognized in past work. In [45], an acoustic feature set that captures the dynamics of the speech signal at the phoneme boundaries was introduced in combination with the traditional acoustic feature set representing the periods of speech that are assumed to be quasi-stationary. In [52], a extended hidden Markov model that integrates generalized dynamic feature parameters into the model structure is developed and evaluated.

Prior work in CRF phonetic recognition has either ignored or used a simplified approach in the transition function implementation. In [41], the transition functions were binary, evaluating to 1 when the phonetic unit label pair matched the values for the defined function and 0 otherwise. This left out any transition clues that where present in the input and let the Viterbi decoding decide which transitions maximized the final probability of the sequence. In [14], a feature set was used for both state and transition features. This feature set was not optimized for boundary detection, thus allowing only a small increase in overall recognition performance. Explicit boundary detection (using a single MLP detector to detect segmental boundaries) was found to be mildly effective, but an expensive transition feature for CRF transitions in [61]. In this work, we examine a wide range of boundary features, already presented in Section 5.2, which are designed to detect as many boundaries as possible without adding a considerable amount of insertions to the system.

In Figure 5.2, the full CRF model structure is given, including the transition features associated with transitions between states. Transitions are abstract units and can be either inside the phoneme, connecting states of the same phoneme, or between phonemes, connecting the last state of the previous phoneme to the first state of the next phoneme. Interesting transitions are the ones connecting two adjacent phonemes, because they indicate phoneme boundaries. The segmentation information present in the transition features can be used to supplement

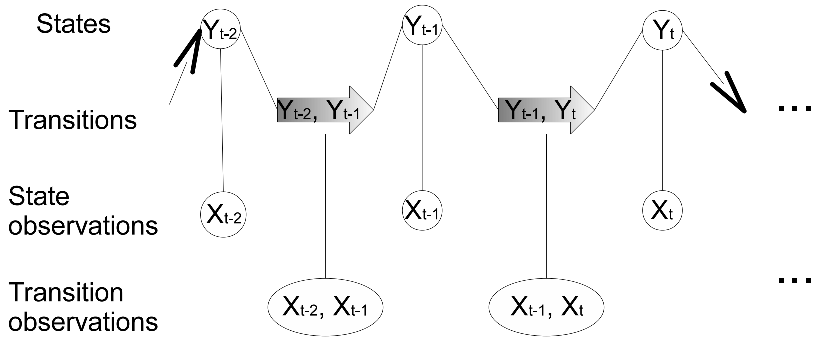


Figure 5.2: Full CRF formulation using transition features

the detection-based system. This external source of information about phonetic unit boundaries is incorporated into the system during the training and decoding processes. During the training process, the association weights of each transition feature with every possible transition (inside/between phoneme) are estimated. Then during decoding, the segmentation information is used to identify better the possible transitions when searching for the best path. The decoding is still frame-based, but benefits from clues normally unavailable in standard HMM recognition, where only state associated features are available. Like the state features, the transition feature increase their positive contribution when having high degree of discrimination between different types of transitions (between different phoneme pairs). One can research the best possible transition set for every type of transition. But even in a bad scenario, where a transition feature can only discriminate between two events, phonetic boundary or not boundary, it still provides useful information in the decoding process. In Chapter 6, the results from different combinations of transition features are given.

Including specialized transition features into the detection-based system, in no way constitutes an implementation of a segmental speech recognition system. It is just an initial step to identify possible features for an efficient segmentation and incorporate this information into the proposed system developed for this thesis. Towards the goal of full segmental processing capabilities, one can develop transition models that are placed on the possible boundaries provided either by an acoustic or a probabilistic segmentation. More details on this in Chapter 7.

## Chapter 6

# Experimental Setup and Results

In this chapter, the experiments conducted during this thesis are presented. Each section is dedicated to a detailed explanation of the setup using any figures when needed. Also the result of each experiment is given together with the analysis of the result. First a baseline detection-based setup is given that is used throughout all experiments for comparison. Then other experimental setups are presented, including extensions and improvements to the baseline system. Different degrees of sophistication are used in each experiment, thus creating a big set of experimental results in each case. Next some common characteristics of the experimental setups are presented. The characteristics that are not shared among all setups, are presented in each individual section.

For all experiments, the TIMIT speech database is used and is presented in Appendix A.1. The si and sx utterances are used to form 1 training set and 1 cross validation set and 2 test sets. The rules used for the subdivision of the dataset are also described in Appendix A.1.

As already presented in Section 2.3, ANN MLPs are used as detectors of speech events. Various speech events are used in this thesis, so a variability of MLP outputs is used for speech event presence estimation. Nevertheless, all MLPs share the same construction properties, morphology and training procedure. For all MLP detectors, the International Computer Science Institute (ICSI) Quicknet toolbox for ANN is used to implement them. From the developers of the toolbox, the Quicknet package is a general purpose MLP toolkit and has been used for tasks other than ASR, including handwriting recognition. It has been used for both hybrid MLP/HMM ASR, in which only MLPs are used to model state emission probabilities, and tandem ASR, which uses MLPs as a kind of nonlinear discriminant analysis prior to Gaussian mixture modeling. In this thesis our approach is to use the MLPs in tandem with either HMMs or CRFs in the final level of event merging and verification. More information about the

Quicknet software package can be found in [24]. The input features used to feed the MLPs are PLP coefficients plus first and second-order deltas. A nine-frame window of features is used in the input thus creating a 351-sized input vector. The structure of the MLPs contain a 351-node input layer, one hidden layer of 1000 hidden units and the output layer as described in Section 2.3 - dependent on the speech event detection. The training process of the MLPs uses a random selection of 416 speakers taken from the training set of all dialect regions and the convergence is evaluated on a cross-validation set of 46 speakers from the training set.

In the final step of event merging, the CRF modeling framework is used extensively. Different implementation of the framework are available online. In this thesis, a C++ implementation is used, developed in the Ohio State University Speech and Language Technologies Lab. The speech event outputs are used as inputs to a CRF modeling framework, that is trained with the TIMIT training set. The training process is stopped after 50 iterations and the iteration that gives the best accuracy on the cross validation set is selected. Also the HTK HMM Toolkit [13] is used when HMM modeling is needed, usually for comparison to the CRF modeling approach.

The recognition task for all experiments is phoneme recognition on the TIMIT dataset. The original 61 phoneme labels of TIMIT are reduced to 39 corresponding phoneme labels as proposed in [32]. The conversion is done with finite-state machines and the rules are presented in Appendix A.3. We measure hits, deletions, substitutions and insertions of phoneme labels in the total labels of the core test set and the extended test set. To evaluate each setup, two metrics are used: the correct percentage and the accuracy percentage. The correct percentage measure, ignores the insertion errors and is given by:

$$C = \frac{T - D - S}{T} \cdot 100\% \tag{6.1}$$

where $T$ = total number of labels, $D$ = number of deleted labels, $S$ = number of substituted labels. The accuracy is a more representative measure and is given by:

$$A = \frac{T - D - S - I}{T} \cdot 100\% \tag{6.2}$$

where $T$ = total number of labels, $D$ = number of deleted labels, $S$ = number of substituted labels and $I$ = number of inserted labels.

## 6.1 Baseline detection-based experiments

The first detection-based setup of this thesis uses 44 phonological event detectors of Table 2.1. The output of the detectors is a frame-based posterior probability of the presence of each of these events. Each event belongs to one of the 8 groups of events. The probability of the events that belong to the same group sum to 1, as they are mutually exclusive. No transition speech events are used. The results of the baseline setup are given in Table 6.1.

| SETS | Correct % | Accuracy % | Hits | Del | Subs | Ins |
|---|---|---|---|---|---|---|
| C.V. | 72.82 | 69.85 | 9882 | 1289 | 2400 | 403 |
| Core Test | 70.82 | 67.95 | 4613 | 694 | 1207 | 187 |
| Ext Test | 72.08 | 68.89 | 23194 | 3094 | 5892 | 1024 |

Table 6.1: Baseline detection-based experimental setup

The convergence of the training process was achieved after 30 iterations. For comparison we can take into consideration an HMM tandem system using the same phonological speech event detectors. The tandem system uses a 16 Gaussian mixtures per state HMM and 3 states per phoneme. Also The results are reported in [40] for a Core test set: $Correct\% = 72.42$ and $Accuracy\% = 66.85$. We can see that the one-state CRF model of 44 phonological event inputs has comparable results to an 3-state, 16-mix HMM model. To understand better this comparison, in Table 6.2, the number of parameters of each modeling framework are given. These parameters take values during the training process and give a direct indication of the complexity and computation time of the method. CRFs have significantly less parameters to achieve comparable performance to the HMMs.

| Model | states per phone | number of parameters |
|---|---|---|
| Tandem HMM (16-mix) | 3 | 205,350 |
| CRF | 1 | 4464 |

Table 6.2: Same performance different complexity: CRF and HMM modeling comparison

## 6.2 Various level speech event merger experiments

In Table 3.1, we have seen speech events at three levels. Also in Figure 3.3, different CRF configurations were given, each one combining events of either one or more levels. In this section, a detection-based experimental setup that uses speech events from different levels is tested and evaluated. A multitude of variations of this setup are possible, due to the many combinations of speech events. Not all of them are successful, so after the comprehensive investigation done for this thesis, the most successful ones are going to be presented.

First, we are going to include low-level speech signal events that are represented by MFCC features. Not only MFCCs but also theirs first and second order time derivatives are included, to catch dynamic properties of the spectral events.

Also the MFCC squares are important, because we want to include second order statistic information of the MFCCs. It is important for the two feature families, the phonological posteriors and the MFCCs, to have a similar dynamic range of values. The final feature vector is created when the two feature families are merged. Then this feature vector of combined low and medium level events is fed to the CRF, to complete the recognition task. To limit the dynamic range of MFCCs, mean and variance normalization is used. The MFCC squares are transformed using a 5-th order polynomial transformation function, that limits the range between 0 and 1. The approximate mean value of squared MFCCs was estimated around 300 (not every MFCC has the same mean, but the same transform was used for all). The standard deviation was also around 300. The max value was around 3000, so values approaching max are mapped non-linearly to 1. In Figure 6.1, the fitted transformation function curve is shown, together with the original and new MFCC squares range. We also compute the first and second derivatives of the MFCC squares. The size of the speech event vector grows to 122. In the result tables following, we refer to this setup as **Exp2-Var1**.

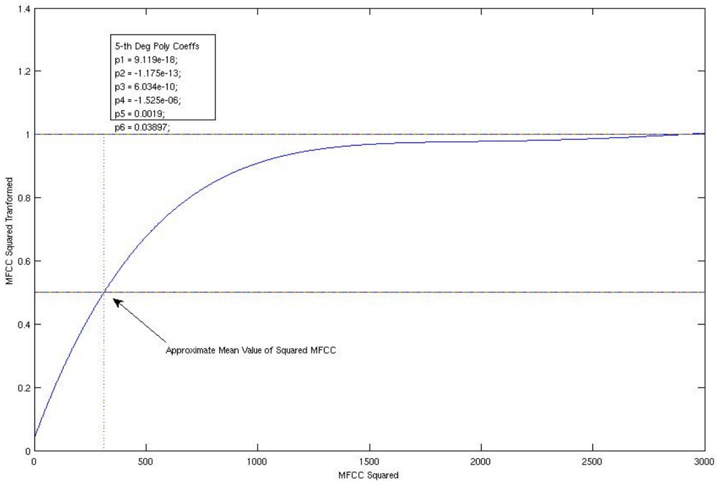


Figure 6.1: MFCC Squares transformation function

Going one step further, we add some more low level events to the already merged phonological and MFCC-based. The new setup also includes the following events: Degree of Voicing, Spectral Roll-off, Spectral Centroid and Spectral Flux. Also first and second order derivatives for the new features are used. The size of the speech event vector grows to 134. We refer to this setup as **Exp2-Var2**.

Another addition is to include the first and second time derivatives of the phonological events. This extra information provides an indication of the dynamic properties of the phonological events. The size of the speech event vector grows to 222. We refer to this setup as **Exp2-Var3**.

Next we want to compare the performance of low-level with mid-level events in contrast to mid-level with high-level events. So we combine the mid-level phonological speech events with the high-level phone presence events. The 61 TIMIT phonemes are used as high level events. The final event vector has size of 105. We refer to this setup as **Exp2-Var4**. We make the comparison with the low-mid level event setup, we de-correlate and reduce the size of vector of the "Phono$\Delta\Delta$ + MFCC + Spec" setup from 222 to 105. The Karhune-Loeve Transform (KLT ) is used to find the principal components of the vector and then 105 features were selected. We refer to this setup as **Exp2-Var5**.

Finally a setup that combines all levels of events into one experiment is constructed. Having all three levels of events, is what can someone do to have as much information as it is available to increase the performance of the phoneme recognition task. The size of the input vector is blown out to 405. We refer to this setup as **Exp2-Var6**.

| Codename | Short Description | Size | Params | Iters |
|---|---|---|---|---|
| Exp2-Var1 | Phono+MFCC | 122 | 8208 | 29 |
| Exp2-Var2 | Phono+MFCC+Spec | 134 | 8784 | 34 |
| Exp2-Var3 | Phono$\Delta\Delta$+MFCC+Spec | 222 | 13008 | 43 |
| Exp2-Var4 | Phono+Phone | 105 | 7392 | 25 |
| Exp2-Var5 | Phono$\Delta\Delta$+MFCC+Spec+KLT | 105 | 7392 | 24 |
| Exp2-Var6 | Phono$\Delta\Delta$+Phone$\Delta\Delta$+MFCC+Spec | 405 | 21792 | 43 |

Table 6.3: Second experimental setup variations

In Table 6.3, a complete summary of all the previously described experimental variations is presented. For each one the short name that will be used next, and a description is given. Second the number of speech events that are contained in the input vector is given. The parameters that are trained during the training step are given. Last in this table is the number of iterations necessary for convergence during training.

| Codename | C.V. set | | Core Test set | | Ext Test set | |
|---|---|---|---|---|---|---|
| | Corr % | Acc % | Corr % | Acc % | Corr % | Acc % |
| Baseline | 72.82 | 69.85 | 70.82 | 67.95 | 72.08 | 68.89 |
| Exp2-Var1 | 75.74 | 71.39 | 73.58 | 69.42 | 74.58 | 70.05 |
| Exp2-Var2 | 75.85 | 71.49 | 73.79 | 69.79 | 74.72 | 70.26 |
| Exp2-Var3 | 76.62 | 72.17 | 74.55 | 70.37 | 75.45 | 70.91 |
| Exp2-Var4 | 76.12 | 72.67 | 74.92 | 69.94 | 75.05 | 70.95 |
| Exp2-Var5 | 75.77 | 71.82 | 74.07 | 70.08 | 74.85 | 70.75 |
| Exp2-Var6 | 78.00 | 73.13 | 75.87 | 71.71 | 77.12 | 72.23 |

Table 6.4: Second experimental setup phoneme recognition results

Phoneme recognition task results are shown in Table 6.4. The increase in accuracy we see when MFCC-based features are implemented to the Posterior CRF system, is quite good than one would expect (somewhat smaller than 2% in absolute value percentage). This may happen for a number of reasons:

- Although some of the phoneme discrimination ability of the MFCC features is already inherent to the system, because the audio features used to derive the phonetic posteriors are PLPs. But it seems that the introduction of any extra information that comes directly from the spectral characteristics are nicely integrated into the system.

- The increase we see is due to the different scale that is used in the MFCC analysis . We use 25msec frames every 10 msec in contrast with the posteriors that are computed from 105 msec frames every 10 msec. In Section 6.3, more detailed experiments on multi-scale analysis will be presented.

The inclusion of the other low level speech events seems to only increase the performance marginally. The quality of these speech events in phoneme recognition task, seem to be poor. The information they hold for phoneme discrimination is very low.

A moderately satisfactory increase in performance comes from the first and second derivatives of the phonological events. The dynamic information they hold, provides an important clue to the recognition setup. Unfortunately the size of the input event vector increases dramatically and so does the number of parameters and the number of iterations needed in training.

The KLT and dimensionality reduction from 222 to 105 features decreased a bit the recognition accuracy. Most probably, this happened because some informative features were removed as a trade-off to decrease dimensionality and complexity of the setup.

The mid- and high-level event merged setup, with the same number of input event vector, seems to have similar performance compared to the low- to mid-level event setup.

Finally, when combining all available speech events, from all levels, we have the best performance. The number of features goes up and so does the number of parameters that need to be trained. The CRF is a really successful integration tool, that efficiently combines the highly correlated and redundant speech events. It takes the most out of every available bit of information included in the event vector and provides the best performance, only slightly increasing the complexity of the system.

In Table 6.5, there is statistical significance testing results for pairs of experiments. The significance testing is done using two methods: (a) the paired t-test and the (b) two-way anova without replication. The measurement variable is the Percent Phoneme Recognition Accuracy and Correct and the nominal variables

| Experimental Pair | One-Tailed Paired T-Test | | Two-way Anova w/out Replication | |
|---|---|---|---|---|
| | Core | Ext. | Core | Ext. |
| Baseline & Var1 | Y | Y | Y | Y |
| Baseline & Var2 | Y | Y | Y | Y |
| Baseline & Var3 | Y | Y | Y | Y |
| Baseline & Var4 | Y | Y | Y | Y |
| Baseline & Var5 | Y | Y | Y | Y |
| Baseline & Var6 | Y | Y | Y | Y |
| Var1 & Var2 | N | N | N | N |
| Var1 & Var3 | N | Y | N | Y |
| Var1 & Var4 | N | Y | N | Y |
| Var1 & Var5 | N | Y | N | Y |
| Var1 & Var6 | Y | Y | Y | Y |
| Var2 & Var3 | N | Y | N | Y |
| Var2 & Var4 | N | Y | N | Y |
| Var2 & Var5 | N | N | N | N |
| Var2 & Var6 | Y | Y | Y | Y |
| Var3 & Var4 | N | N | N | N |
| Var3 & Var5 | N | N | N | N |
| Var3 & Var6 | Y | Y | Y | Y |
| Var4 & Var5 | N | N | N | N |
| Var4 & Var6 | Y | Y | Y | Y |
| Var5 & Var6 | Y | Y | Y | Y |

Table 6.5: Second experimental setup significance testing

are the results for the two experimental variations and the other is the utterances. So with the above status we were able to perform the tests. We tested two hypotheses: the first was tested with the one-sided paired t-test and we tested whether the difference in performance was significantly higher than the previous setup; the second was tested with the two-way anova and we tested whether the performance difference was significant. The result is a binary value showing if the performance difference is significant at 0.05 alpha level.

The statistical significance testing results confirm the following conclusions for the experimental variations:

- All experimental variations are significantly better than the baseline detection-based system, based only on phonological speech events.

- Variation 1 and Variation 2 are not different, meaning that MFCC-based features adequately represent the low-level speech events. The addition of the other low level speech events (Voicing, Spectral Flux, Centroid, ...) is redundant.

- Variations 3 to 5 are significantly better and different from Variation 1 only in the Extended Test set. Phonological Deltas and Phone presence events need a bigger test set to prove they are important.

- Variation 2 has proved to give statistically the same results to Variation 1, so we expect about the same significance test results compared to the other sets. The only difference is that Variation 2 proved to have statistically insignificant difference from Variation 5 in all sets, Core as well as Ext.

- Variation 3 has statistically insignificant difference from 4 and 5. This proves that when applying KLT transformation and reduction from 222 to 105 features, the setup remains essentially the same. Also proves the remark made earlier that low- to mid-level speech event combination gives similar results to mid- to high-level combination. Also in the same sense, Variations 4 and 5 give insignificant different results.

- Variation 6 is left for the end. This experimental variation gives significantly better performance from all previous variations in all measured sets. It gives the best performance achieved with the combination of different level speech events. Although the increase in complexity has been already reported.

## 6.3 Multiple time-scale processing experiments

In Chapter 4, a modeling framework for the speech rate variability was introduced. The idea of multiple time scale analysis was given. Then a functional system was proposed, in order to support the analysis in the detection-based system. Different experiments were conducted, in order to evaluate the proposed solutions.

First multiple FR / WS size speech features were combined and used in a speech recognition task as described in Section 4.4.1. The all-inclusive method was achieved by including different temporal analysis features in a constant frame-rate box. The resulting feature vector was calculated at constant FR, but the features inside the box were computed at different FRs. Using this method, we appended the standard single FR / WS method with a number of signal parameters from other resolutions. The resulting feature vector is highly correlated, so a de-correlation step was deemed necessary.

For HMMs we used the HTK Toolbox and trained 48 Context-Independent (CI) 3-state 16-mixture monophone models on the training set and performed the multiple rate recognition task on the other sets. For comparison we performed the same recognition task on CRF using 48 CI 1-state monophone models. Then a reduction to 39 phonemes on both frameworks was done for comparison.

The features we are using are MFCC with delta and acceleration (MFCC_D_A) combined from different FR/WS pairs, shown as MFCC-MFR in the following tables. We also report results on MFCC_D_A computed at 10 msec as baseline, shown as MFCC-10.

| Method | Feature group | Core Set | | Ext Set | |
|---|---|---|---|---|---|
| | | Acc % | Rec % | Acc % | Rec % |
| HMM | MFCC-10 | 49.08 | 52.77 | 48.98 | 53.15 |
| HMM | MFCC-MFR | 48.35 | 53.55 | 49.03 | 54.50 |
| CRF | MFCC-10 | 47.33 | 51.87 | 47.22 | 51.78 |
| CRF | MFCC-MFR | **50.94** | **58.11** | **51.00** | **58.49** |

Table 6.6: Multi-Rate *Phoneme Recognition* Results: All-inclusive Combination Method

From Table 6.6, we can see that HMMs have better performance when using plain single-rate MFCC_D_A parameters. When using multi-scale MFCC_D_A features, HMMs cannot integrate efficiently this extra information. In contrast, the detection-based system improves performance by combining efficiently these highly correlated parameters.

Next we proceed using the CRF framework and combine 44 phonological event posteriors with single-rate MFCC_D_A and also with multirate MFCC_D_A. We also include results using exclusively phonological event posteriors as a baseline.

| Feature group | Core Set | | Ext Set | |
|---|---|---|---|---|
| | Acc % | Rec % | Acc % | Rec % |
| Posteriors | 66.72 | 68.68 | 68.16 | 70.32 |
| Posteriors+MFCC-10 | 68.25 | 71.12 | **69.86** | 72.86 |
| Posteriors+MFCC-MFR | **68.74** | **72.83** | **69.86** | **74.20** |

Table 6.7: Multi-Rate *Phoneme Recognition* Results: Using Phonological Posteriors - CRF Framework - All-inclusive method

The results in Table 6.7, show the detection-based system's ability to integrate features of different quality and time-scale. When Posteriors are merged with single-rate MFCCs, an improvement is clear. Adding multi-rate MFCCs improves somewhat the results. The improvement in recognition is significant.

The next important group of experiments was the evaluation of the optimal scale selection method, presented in Section 4.4.2. A certain procedure was followed during the construction of the experimental setup. First we computed the spectral change metric given in Equation 4.1. The quantity used to compute the spectral region differences was the Mel-Scale Spectral Magnitude, computed by 20 channel filterbank analysis. We run a first pass recognition task using multiple FR/WS pairs defined in Section 4.3, on the cross validation set. We segmented each utterance to 30 msec segments and labeled each segment using the frame classification results of the first pass. Every segment that was classified correctly under one FR/WS pair, it was labeled with its corresponding ROS label (ROS=10 for FR/WS=10/25msec, etc.). Then an ML training was done using the spectral change metric as input and the ROS label as output. The parameters of Eq. (4.1) and Eq. (4.2) were trained using ML estimation. We kept this mapping as a baseline.

Next we implemented an MCE/GPD embedded iterative training algorithm and done a re-training of the mapping functions. In the first pass, we re-train the parameters of Eq. (4.1). Lower frequencies are mapped with larger weights and higher frequencies tend to have smaller weights. In the second pass we re-train the parameters of Eq. (4.2). To simplify the training procedure we kept the parameter $a$ to value 5 and parameter $c$ to value 5 in order to have a uniform dynamic range of frame rates, i.e, we re-train parameters $b$ and $d$. We use a maximum of 30 iterations.

The mapping function that emerged, as the result of this iterative procedure, was used to select the frame-rate on a frame-based classification task and also the normal recognition task. Frame classification and utterance recognition results on ML and MCE trained FRSUs were then compared.

Next, the results are presented for the detection-based optimal scale selection method. We used the classification results from the cross-validation set to train the FRSU with the MCE method as described earlier. We performed a variable rate experiment on the Core and Ext sets using the ML trained FRSU as baseline and also the MCE trained FRSU, in order to select the optimal FR/WS for each 30msec segment. In this experiment, we used MFCC_D_A features combined from different rates and windows. We report the results for a *frame-based classification* task in Table 6.8. We also have included frame-level results from the baseline single-rate MFCC system and the multi-rate all-inclusive method, using MFCC and Posteriors+MFCC. Also *phoneme recognition* task results are presented in Table 6.9.

| System setup | Features | Core % | Ext % |
|---|---|---|---|
| Const-Rate | MFCC-10 | 50.06 | 48.93 |
| All-inclusive Method | MFCC-MFR | 54.54 | 53.85 |
| | Post+MFCC-MFR | 70.07 | 70.70 |
| ML FRSU | MFCC | 49.81 | 49.36 |
| | Post+MFCC | 61.37 | 61.94 |
| MCE FRSU | MFCC | 51.70 | 51.10 |
| | Post+MFCC | **72.40** | **73.11** |

Table 6.8: Variable & Multi-Rate Speech Recognition System: *Frame Classification* Results

From the results in Table 6.8 for the *frame-based classification* task, we see that the MCE method outperforms the ML method. Also the all-inclusive method performs well using multi-rate MFCC features. Finally the MCE method is the best performer when combining Posteriors and MFCCs at different rates. Looking at the *phoneme recognition* results in Table 6.9, we see the ML method is

| System | Features | Core Set | | Ext Set | |
|---|---|---|---|---|---|
| | | Acc % | Rec % | Acc % | Rec % |
| ML | MFCC | 47.88 | 51.47 | 48.76 | 51.35 |
| FRSU | Post+MFCC | 62.50 | 65.61 | 64.57 | 68.15 |
| MCE | MFCC | 46.48 | 51.14 | 46.75 | 51.81 |
| FRSU | Post+MFCC | **67.49** | **70.33** | **68.81** | **72.41** |

Table 6.9: Variable-Rate Speech Recognition System: Phoneme Recognition Results

| Setup | State features | Transition features |
|---|---|---|
| Baseline | 48 Phonological features | No transition features |
| DPhn | 48 Phonological features | 48 Phonological Delta (1st order) |
| BndF | 48 Phonological features | 6 boundary features |
| DPhn + BndF | 48 Phonological features | 48 Phonological Delta (1st order)<br>+ 6 boundary features |
| DPhn + BndF + DMFCC | 48 Phonological features | 48 Phonological Delta (1st order)<br>+ 13 MFCC delta (1st order)<br>+ 6 boundary features |

Table 6.10: Segmental information experimental setup description

performing better that the MCE method when using MFCCs. MCE performs better when using Posteriors and MFCCs. Also comparing Table 6.7 and Table 6.9, we see that the box method performs better than the best variable rate method (MCE) during recognition. This indicates again the CRFs ability to take the most out of highly correlated features and also the variable rate system's weakness of interpreting good frame classification results to equally good phoneme recognition results. After some experimentation, we have found that by reducing the FR /WS options of the variable rate system, in fact we can increase a bit the performance. This happens due to the reduction of the complexity of the system.

## 6.4 Segmental information experiments

In Chapter 5, an introduction to the basic concepts of the segmental speech recognition approach was given. An initial approach into implementing segmental recognition ideas was proposed in Section 5.4. Transition event detection support was included into the detection-based system. Experimental setups were constructed to evaluate if the proposed approach could give solid results. The transition features described in Section 5.2 were incrementally included into a baseline setup.

In Table 6.10, the different experimental variations are presented. The baseline system includes only a combination of phonological and phone event features

| Setup | C.V. Set | | Core Test Set | | | | | Ext Test Set |
|---|---|---|---|---|---|---|---|---|
| | Acc % | Train Iters | Acc % | Corr | Del | Subs | Ins | Acc % |
| Baseline | 71.4 | 25 | 69.11 | 4597 | 941 | 976 | 95 | 70.25 |
| DPhn | 72.46 | 11 | 70.42 | 4705 | 772 | 1037 | 118 | 71.46 |
| BndF | 72.02 | 19 | 69.53 | 4652 | 849 | 1013 | 123 | 70.84 |
| DPhn + BndF | 73.02 | 2 | 70.77 | 4773 | 692 | 1049 | 163 | 71.76 |
| DPhn + BndF + DMFCC | **73.72** | 3 | **71.51** | 4825 | 647 | 1042 | 167 | **72.32** |

Table 6.11: Segmental information experimental setup results

as state features. 44 Phonological event and 61 Phone presence posteriors were combined and transformed using KLT. Then a reduction was done to 48 combined phonological-phoneme events. No transition features were used in the baseline setup. First we include the 48 Phonological-Phone deltas as transition region indicators. The next variation uses 6 explicit boundary features presented in Table 5.1. Then a variation which uses both deltas and boundary features is constructed. Finally, to the previous transition features, MFCC deltas are merged. Each variation is given a codename which is used in the result tables that follow.

We report two different performance indicators. The first is the phoneme recognition performance of the setup. This is an indicator of how well our experimental boundary feature setup has done in recognizing phonetic units. The second is how well our setup has done in detecting the transition boundaries of phonetic units.

### Recognition results

Our first set of performance indicators are the overall recognition results of different setups. Results for the three sets (Cross Validation, Core Test, Extended Test) are shown in Table 6.11. By using the phonological deltas (DPhn) we got a marginally significant improvement in accuracy. In contrast, when we used the six boundary features alone (BndF), the improvement was not significant. Then when we used both phonological deltas and boundary features (DPhn + BndF) we got a better accuracy from the previous two experiments, as expected. Finally when we used all available transition features - phonological deltas, boundary features and MFCC deltas (DPhn + BndF + DMFCC) - we got the best accuracy. Note that the improvement is due to the significant reduction in deletions (by over 20%). Also by adding more transition features, the training process converges with only a couple of iterations.

### Boundary detection results

In addition to recognition performance, we can see how well the detection-based system directly detects segment boundaries. We report the overall boundary detection performance, i.e., the detection ratio for transitions between two phonetic units in terms of precision and recall for the extended test set. These results in Table 6.12 offer an overview of the detector performance. Two tolerance levels in the detection of transition boundaries are reported: 10 msec (strict) and 20 msec (normal). One can see that when using phonological deltas, a slight increase in recall is achieved with a matching decrease in precision. When using boundary features, we get an increase in recall without any loss in precision. When using both phonological deltas and boundary features we get a complementary effect, recall increases significantly with a small decrease in precision. Finally the addition of MFCC deltas provides a negligible gain in recall.

| Tolerance: | 10 msec | | 20 msec | |
|---|---|---|---|---|
| | Precision | Recall | Precision | Recall |
| Baseline | 0.89 | 0.78 | 0.955 | 0.855 |
| DPhn | 0.875 | 0.795 | 0.95 | 0.88 |
| BndF | 0.89 | 0.795 | 0.955 | 0.87 |
| DPhn + BndF | 0.88 | 0.81 | 0.945 | 0.89 |
| DPhn + BndF + DMFCC | 0.88 | **0.815** | 0.945 | **0.895** |

Table 6.12: Overall boundary detection performance

The detailed performance for transitions between broad phonetic classes (BPC) are reported in Table 6.12 for the extended test set. The phonetic units are grouped into 5 classes, namely: vowels and semi-vowels (VOW), fricatives (FRIC), nasal-flaps (NAS), stops (STOP), silence (SIL). Detection results (precision/recall) for each experimental setup and transitions between these BPC are reported for the strict 10 msec window. The first value in each cell of Table 6.13 is the precision/recall ratio of the transition boundary of the left phonetic class to the right as presented in the table (while the second value is for the right to left transition). Overall, by adding transition events into the detection-based system, boundary detection improves significantly especially among the SIL, STOP and FRIC phonetic classes. It seems that these phonetic classes transitions get the highest complementary effect from the different groups of transition features, so they finally increase their recall without losing precision.

## 6.5 Summary and Discussion

In this chapter, the detection-based approach was tested thoroughly by a number of experiments. The first experimental setup was the implementation of the

| Setup | NAS ↔ STOP | | VOW ↔ FRIC | | VOW↔ STOP | | FRIC ↔ SIL | | STOP ↔ SIL | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | Precision | Recall | Precision | Recall | Precision | Recall | Precision | Recall |
| Baseline | 0.70/0.69 | 0.64/0.86 | 0.92/0.89 | 0.94/0.93 | 0.82/0.92 | 0.95/0.97 | 0.79/0.75 | 0.80/0.75 | 0.66/0.73 | 0.52/0.49 |
| DPhn | 0.71/0.74 | 0.65/0.87 | 0.92/0.89 | 0.93/0.93 | 0.83/0.91 | 0.95/0.96 | 0.78/0.75 | 0.80/0.77 | 0.64/0.70 | 0.56/0.60 |
| BndF | 0.71/0.76 | 0.65/0.86 | 0.92/0.89 | 0.94/0.94 | 0.83/0.91 | 0.96/0.96 | 0.80/0.73 | 0.81/0.78 | 0.69/0.76 | 0.57/0.65 |
| DPhn + BndF | 0.72/0.71 | 0.70/0.96 | 0.92/0.87 | 0.94/0.95 | 0.84/0.91 | 0.96/0.96 | 0.79/0.72 | 0.81/0.78 | 0.65/0.72 | 0.58/0.68 |
| DPhn + BndF + DMFCC | 0.74/0.75 | 0.71/0.88 | 0.92/0.89 | 0.94/0.95 | 0.84/0.91 | 0.96/0.97 | 0.80/0.72 | 0.83/0.78 | 0.68/0.73 | 0.59/0.70 |

Table 6.13: Broad phonetic class boundary detection performance

detection-based system baseline. As a baseline case we assumed the detection of phonological speech events. The phonological events appear to be in the middle level of the speech hierarchy, with the lower level being the spectral events and the higher level the phoneme presence events. Subsequently, different combinations of speech events were tested as an extension to the baseline case. The conclusions drawn were mainly the following: (a) adding low-level spectral events by using MFCC representation was beneficial, (b) all other low-level speech events gave negligible improvement, (c) adding dynamic properties of phonological events by using their deltas was beneficial, (d) low- and mid-level event combination gave similar performance to mid- and high-level events, (e) combining events from all levels gave the best performance, but with an increase in system complexity and training time.

After the event combination experiments, the solutions proposed in Chapter 4 to the speech rate variability were tested. The modeling approach to speech rate variability was the usage of multiple time scales in speech processing. Multiscale analysis support was inserted to the detection-based system throughout all components, from the initial speech signal analysis to the final merging and verification steps. Different methods were tested for the merging on multiple states, an all-inclusive integration and an optimal scale selection method. The first method gave very good results, proving the CRFs capability as integrator of correlated speech events. The second method was more sophisticated and promising, although it suffered complexity issues. After some tweaking on the number of time-scale parameters and utilizing discriminative training methods, the optimal scale selection gave satisfactory results.

The final group of experiments was based on concepts from the segmental processing of speech. Information that is normally used in the segmentation task of a segmental recognition system, were incorporated to the detection-based system by the usage of transition function support of CRFs. Deltas of phonological and phone speech events were used as transition region indicators between phonetic units, increasing the phoneme recognition performance. Also low-level speech events indicating possible transition regions were incuded with limited recognition improvement, but giving the best boundary detection results. The best combination of transition events in terms of phoneme recognition accuracy was the phonological and MFCC deltas, although in terms of boundary detection was the phonological deltas and the dedicated low-level transition events. The broad phonetic classes benefiting the most by the inclusion of transition clues were the silence, stop and fricative classes.

Chapter 7

# Conclusions and Future Work

In this chapter, the most imprortant conclusions drawn during the work on this master thesis are enumerated. They are grouped in conclusions on the overall function of the detection-based system, on the multi-rate functionality of the system and finally on the boundary detection functionality-added system setup.

Finally on the last section of the Chapter, we propose some concepts and ideas on the future direction of the work presented in this thesis.

## 7.1 Conclusions

### On the detection-based system

1. CRF modeling uses a significantly smaller number of trainable parameters compared to HMM to achieve comparable performance.

2. Adding spectral information to the detection-based system in the form of low-level speech events, increases the phone recognition performance by 2% in accuracy.

3. By including first and second order derivatives of speech events, the performance increased, but with a dramatic increase to the speech event vector and the overall system complexity.

4. Merging speech events from all three available levels (low, mid and high) gives optima performance but with an increase in system complexity.

5. Nevertheless, CRF is a successful integration toolbox, keeping the training and testing processes viable, even for large vectors of highly corellated speech events.

### On the multi-rate system

1. Highly correlated speech parameters, from different time-scales were combined in a feature vector. HMMs showed a decline in performance when using the multi-rate vector. In contrast, CRFs showed improved performance.

2. From a true variable speech rate analysis system, we draw the following conclusions:

   (i) When using MCE training on the optimal per-frame time-scale selector, we have better frame classification and phoneme recognition results.

   (ii) Limiting the available time-scales of the system (2-3 on the current implementation), reduces complexity and improves accuracy.

### On the boundary detection aided system

1. A feature set of phonological and cepstrum feature deltas, was a useful transition indicator set that increased the overall recognition performance.

2. Phonological and other spectral and energy domain features, were more important in the boundary detection task. Cepstral feature deltas added insignificant improvement.

3. The broad phonetic classes that return the most correctly detected boundaries are the stop, silence and fricative classes. The final improvement in recognition accuracy is mostly achieved by reducing the deletions.

## 7.2 Proposed future work

The main abstract components of the system can be implemented in the future, by improved versions of the current actual implementations presented in this thesis. Moreover, novel and completely different implementation can substitute the currently proposed, with the only restriction be keeping the main idea of the abstract detection-based system. New speech events can be inserted and old events substituted or removed, based on the evaluation of the recognition accuracy results. New ideas and approaches can be included on the main components of the system, as technology in speech recognition advances. We have already given the example in this thesis by including the concept of multiple time scale processing and boundary detection information.

More specifically, on the concept of multiple time-scales, some improvements are proposed. One is to find a solution in order to interpret good frame classification results into phoneme recognition results on the true variable rate system.

In practice one should work towards minimizing the error rate of the recognition task, by controlling the insertion of the variable rate system and the translation of variable rate frames to states and phonemes. Also, a variable rate system based on phonological posterior features computed at different temporal resolutions should be implemented for improved results.

Moving one step further in including phoneme boundary information, one can use exclusive transition models. Transition models are modeling the dynamics of sound around detected boundaries. A window of analysis centered at the detected boundary can be used. The boundaries can be at phoneme transition regions but also inside phonemes. Because different dynamics exist at transition regions, a trajectory model is necessary to model the fore-mentioned dynamical content of speech. This can be achieved by mapping to a N-degree polynomial and taking the first M-coefficients. The residual also must be taken into account as a normally distributed random variable (mean, variance). The previous analysis must be done for every input feature of the modeling scheme. Finally we will have 2 types of models, the between phoneme transition models and the inside phoneme models. The transition models have been proved to work better than segment models. Also a combination of both seems to achieve the optimal result.

# Bibliography

[1] Ilana Bromberg, Qiang Fu, Jun Hou, Jinyu Li, Chengyuan Ma, Brett Matthews, Antonio Moreno-Daniel, Jeremy Morris, Sabato Marco Siniscalchi, Yu Tsao, and Yu Wang. Detection-based asr in automatic speech attribute transcription project. In *Proceedings of International Conference on Spoken Language Processing*, 2007. [cited at p. 10, 12, 20, 25]

[2] Ozgur Cetin, Arthur Kantor, Simon King, Chris Bartels, Mathew Magimai-Doss, Joe Frankel, and Karen Livescu. An articulatory feature-based tandem aproach and factored observation modeling. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2007. [cited at p. 10, 12]

[3] Ozgur Cetin and Mari Ostendorf. Multi-rate and variable-rate modeling of speech at phone and syllable time scales. In *In Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2005. [cited at p. 11]

[4] Jane W. Chang. *Near-Miss Modeling: A Segment-Based Approach to Speech Recognition.* PhD thesis, Massachusetts Institute of Technology, 1998. [cited at p. 12, 45, 46]

[5] Wu Chou and Biing Hwang Juang. *Pattern Recognition in Speech and Language Processing.* CRC Press, 2003. [cited at p. 19]

[6] Heidi Christensen, Borge Lindberg, and Ove Andersen. Introducing phonetically motivated, heterogeneous information into automatic speech recognition. In *W. J. Barry and W. A. van Dommelen, The Integration of Phonetic Knowledge in Speech Technology*, chapter 5, pages 67–86. Springer, 2005. [cited at p. 12]

[7] Kathleen E. Cummings and Mark A. Clements. Glottal models for digital speech processing: A historical survey and new results. *Digital Signal Processing*, 5:21–42, 1995. [cited at p. 12]

[8] Spiros Dimopoulos, Eric Fosler-Lussier Chin-Hui Lee, and Alexandros Potamianos. Transition features for crf-based speech recognition and boundary detection. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2009. [cited at p. 52]

[9] Spiros Dimopoulos, Alexandros Potamianos, Eric-Fosler Lussier, and Chin-Hui Lee. Multiple time resolution of speech signal using mce training with application to

speech recognittion. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2009. [cited at p. 38, 39]

[10] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern classification.* Wiley, 2 edition, 2002. [cited at p. 20]

[11] Stephane Dupont and Herve Bourlard. Using multiple time scales in a multi-stream speech recognition system. In *Proceedings of International Conference on Spoken Language Processing*, 1997. [cited at p. 11]

[12] Sorin Dusan and Larry R. Rabiner. On integrating insights from human speech perception into automatic speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 2005. [cited at p. 12]

[13] Young et al. *The HTK Book.* Microsoft Corporation, Cambridge, UK, 3.2.1 edition, 2002. [cited at p. 16, 18, 56]

[14] Eric Fosler-Lussier and Jeremy Morris. Crandem systems: Conditional random fields acoustic models for hidden markov models. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2008. [cited at p. 28, 52]

[15] Venkata R. Gadde, Kemal Sonmez, and Horacio Franco. Multirate asr models for phone-class dependent n-best list rescoring. In *In Proceedings of Automatic Speech Recognition and Understanding Workshop*, 2005. [cited at p. 11, 32]

[16] John S. Garofolo, Lori F. Lamel, William M. Fisher, Jonathan G. Fiscus, David S. Pallett, Nancy L. Dahlgrena, and Victor Zue. Timit acoustic-phonetic continuous speech corpus. Technical report, Linguistic Data Consortium, 1993. [cited at p. 21, 84]

[17] J. Glass, J. Chang, and M. McCandless. A probabilistic framework for feature-based speech recognition. In *Proceedings of the International Conference on Spoken Language Processing*, 1996. [cited at p. 46]

[18] Asela Gunawardana, Milind Mahajan, Alex Acero, and John Platt. Hidden conditional random fields for phone classification. In *Proceedings of International Conference on Speech Communication and Technology*, 2005. [cited at p. 12, 23, 24]

[19] Astrid Hagen and Herve Bourlard. Using multiple time scales in the framework of multi-stream speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 2000. [cited at p. 11, 32]

[20] Hynek Hermansky. Perceptual linear predictive (plp) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752, April 1990. [cited at p. 17]

[21] Hynek Hermansky, Daniel P.W. Ellis, and Sangita Sharma. Tandem connectionist feature extraction for conventional hmm systems. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2000. [cited at p. 22]

[22] Jun Hou, Lawrence Rabiner, and Sorin Dusan. Automatic speech attribute transcription (asat) - the front end processor. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2006. [cited at p. 12]

[23] Elliot Moore II, Mark Clements, John Peifer, and Lydia Weisser. Investigating the role of glottal features in classifying clinical depression. In *Proceedings of the 25th Annual International Conference of the IEEE EMBS*, 2003. [cited at p. 12]

[24] D. Johnson. *ICSI Quicknet Software Package*. International Computer Science Institute, http://www.icsi.berkeley.edu/Speech/qn.html. [cited at p. 56]

[25] Biing-Hwang Juang, Wu Chou, and Chin-Hui Lee. Minimum classification error rate methods for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 5(3):257–265, May 1997. [cited at p. 12, 39, 41, 42]

[26] Biing-Hwang Juang and Shigeru Katagiri. Discriminative learning for minimum error classification. *IEEE Transactions on Signal Processing*, 40:3043–3054, December 1992. [cited at p. 39]

[27] Shigeru Katagiri, Biing-Hwang Juang, and Chin-Hui Lee. Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method. *Proceedings of the IEEE*, 86(11):2345–2373, November 1998. [cited at p. 39]

[28] Yeon-Jun Kim and Alistair Conkie. Automatic segmentation combining an hmm-based approach and spectral boundary correction. In *Proc. 7th International Conference on Spoken Language Processing*, Denver, Colorado, USA, 2002. [cited at p. 47]

[29] K. Kirchhoff. *Robust speech recognition using articulatory information*. PhD thesis, University of Bielefeld, 1999. [cited at p. 48]

[30] Chin-Hui Lee. From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 2004. [cited at p. 9, 10, 12]

[31] Chin-Hui Lee, Mark A. Clements, Sorin Dusan, and Eric Fosler-Lussier. An overview on automatic speech attribute transcription (asat). In *Proceedings of International Conference on Spoken Language Processing*, 2007. [cited at p. 9]

[32] K. Lee and H. Hon. Speaker-independent phone recognition using hidden markov models. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(11):1641–1648, November 1989. [cited at p. 56]

[33] Matthew Lee. Feature extraction toolbox: A matlab tool set for speech analysis. Technical report, Georgia Institute of Technology, 2008. [cited at p. 47]

[34] Steven C. Lee. Probabilistic segmentation for segment-based speech recognition. Master's thesis, Massachusetts Institute of Technology, 1998. [cited at p. 12, 49]

[35] Jinyu Li and Chin-Hui Lee. On designing and evaluating speech event detectors. In *Proceedings of International Conference on Spoken Language Processing*, 2005. [cited at p. 10, 20, 21]

[36] R. Lippmann. Speech recognition by human and machines. *Speech Communication*, 22:1–14, 1997. [cited at p. 9]

[37] Karen Livescu, Ari Berzman, Nash Borges, Lisa Yung, Ozgur Cetin, Joe Frankel, Simon King, Mathew Maginai-Doss, Xuemin Chi, and Lisa Lavoie. Manual transcription of conversational speech at the articulatory feature level. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2007. [cited at p. 11]

[38] Karen Livescu, Ozgur Cetin, Mark Hasegawa-Johnson, Simon King, Chris Bartels, Nash Borges, Arthur Kantor, Partha Lal, Lisa Yung, Ari Bezman, Stephen Dawson-Haggerty, Bronwyn Woods, Joe Frankel, Mathew Magimai-Doss, and Kate Saenko. Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 jhu summer workshop. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2007. [cited at p. 19, 20]

[39] Chengyuan Ma, Yu Tsao, and Chin-Hui Lee. A study on detection based automatic speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 2006. [cited at p. 10]

[40] J. Morris and E. Fosler-Lussier. Discriminative phonetic recognition with conditional random fields. In *HLT-NAACL Workshop on Computationally Hard Problems and Joint Inference*, 2006. [cited at p. 24, 25, 57]

[41] J. Morris and E. Fosler-Lussier. Further experiments with detector-based conditional random fields in phonetic recognition. In *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages IV–441–IV–444, Honolulu, Hawaii, USA, April 2007. [cited at p. 48, 52]

[42] Jeremy Morris and Eric Fosler-Lussier. Combining phonetic attributes using conditional random fields. In *Proceedings of International Conference on Spoken Language Processing*, 2006. [cited at p. 9, 12, 23, 25, 26]

[43] Jeremy Morris and Eric Fosler-Lussier. Conditional random fields for integrating local discriminative classifiers. *IEEE Transactions on Audio, Speech and Language Processing*, 16(3), March 2008. [cited at p. 26]

[44] K. Kamal Omar, Mark Hasegawa-Johnson, and Stephen Levinson. Gaussian mixture models of phonetic boundaries for speech recognition. In *In Proceedings of Automatic Speech Recognition and Understanding Workshop*, 2001. [cited at p. 12]

[45] M.K. Omar, M. Hasegawa-Johnson, and S. Levinson. Gaussian mixture models of phonetic boundaries for speech recognition. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 33–36, 2001. [cited at p. 52]

[46] Brian M. Ore and Raymond E. Slyh. Score fusion for articulatory feature detection. In *Proceedings of International Conference on Spoken Language Processing*, 2007. [cited at p. 11, 19, 20]

[47] Mari Ostendorf, Vassilios V. Digalakis, and Owen A. Kimball. From hmm's to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Acoustics Processing*, 4(5):360–378, September 1996. [cited at p. 12, 45, 46, 51, 52]

[48] Gerasimos Potamianos, Juergen Luettin, and Chalapathy Neti. Hierarchical discriminant features for audio-visual lvcsr. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2001. [cited at p. 38]

[49] Lawrence Rabiner and Biing-Hwang Juang. An introduction to hidden markov models. *Acoustics, Speech and Signal Processing Magazine*, 3(1):4–16, Jan 1986. [cited at p. 18]

[50] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Signal Processing. PTR Prentice Hall, Englewood Cliffs NJ, 1993. [cited at p. 9, 19, 20]

[51] Monica Rajamanohar and Eric Fosler-Lussier. An evaluation of hierarchical articulatory feature detectors. In *Proceedings of Automatic Speech Recognition and Understanding Workshop*, 2005. [cited at p. 11]

[52] Chengalvarayan Rathinavelu and Li Deng. Use of generalized dynamic feature parameters for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 5(3):232–242, May 1997. [cited at p. 52]

[53] Sourabh Ravindran. A physiologically inspired method for acoustics classification. *EURASIP Journal on Applied Signal Processing*, 9:1374–1381, 2005. [cited at p. 12]

[54] Richard Rose and Parya Momayyez. Integration of multiple feature sets for reducing ambiguity in asr. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2007. [cited at p. 19]

[55] Eric D. Sandness. Discriminative training of acoustic models in a segment-based speech recognition. Master's thesis, Massachusetts Institute of Technology, 2000. [cited at p. 12]

[56] Hossein Sedarat, Rasool Khadem, and Horacio Franco. Simplified neural network architectures for a hybrid speech recognition system with small vocabulary size. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 1998. [cited at p. 22]

[57] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *Proceedings of HLT-NAACL*, 2003. [cited at p. 26]

[58] Shihab Shamma. On the role of space and time in auditory processing. *TRENDS in Cognitive Sciences*, 5(8):340–348, August 2001. [cited at p. 12]

[59] Jorge Silva, Vivek Rangarajan, Viktor Rozgic, and Shrikanth Narayanan. Information theoretic analysis of direct articulatory measurements for phonetic discrimination. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2007. [cited at p. 19]

[60] Malcolm Slaney. Auditory toolbox, version 2. Technical report, Intercal Reseach Corporation, 1998. [cited at p. 15]

[61] Y. Wang. Integrating phone boundary and phonetic boundary information into ASR sytems. Master's thesis, Dept. of Computer Science and Engineering, The Ohio State University, 2007. [cited at p. 52]

[62] H. You, Q. Zhu, and A. Alwan. Entropy-based variable frame rate analysis of speech signals and its application to asr. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2004. [cited at p. 11, 32]

[63] Jing Zheng, Horacio Franco, and Andreas Stolcke. Rate of speech modeling for large vocabulary conversational speech recognition. In *In Proceedings of Automatic Speech Recognition: Challenges for the new Millenium*, Paris, France, 2000. ISCA Tutorial and Research Workshop (ITRW). [cited at p. 11, 32]

[64] Qifeng Zhu and Abeer Alwan. On the use of variable frame rate analysis in speech recognition. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2000. [cited at p. 11, 32]

[65] Qifeng Zhu, Barry Chen, Nelson Morgan, and Andreas Stolcke. On using mlp features in lvcsr. In *Proceedings of International Conference on Spoken Language Processing*, 2004. [cited at p. 20]

[66] Qifeng Zhu, Barry Chen, Nelson Morgan, and Andreas Stolcke. Tandem connectionist feature extraction for conversational speech recognition. In *Proceedings of Machine Learning for Multimodal Interaction Workshop*, 2004. [cited at p. 11]

# Appendices

Appendix A

# Dataset and Labeling

## A.1 TIMIT Database

TIMIT database is a joint effort among the Massachusetts Institute of Technology, Stanford Research Institute, and Texas Instruments under sponsorship from the Defense Advanced Research Projects Agency - Information Science and Technology Office (DARPA-ISTO). It has been designed to provide speech data for the acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems. The recording of speech was done at Texas Instruments, transcribed at MIT and prepared for production by the National Institute of Standards and Technology (NIST).

The dataset contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect regions of the United States. Each speaker is associated with a dialect region based on where he lived during childhood. Each region contains recognized geographical areas as presented in Language Files from Ohio State University Linguistics Department. Note that western region does not have clearly specified boundaries and there is one region for people that moved around a lot during childhood. In Table A.1, the regions are presented and how many speakers are associated with each region.

In Table A.2, the speech material of the database is presented. The dialect sentences (the SA sentences) expose the dialectal variants of the speakers and were read by all speakers. The phonetically-compact sentences provide coverage of pairs of phones. Phonetic contexts thought to be either difficult or of particular interest have extra occurrences. Each speaker read 5 of these sentences (the SX sentences) and each text was spoken by 7 different speakers. The phonetically-diverse sentences (the SI sentences) originate from existing sources, the Brown Corpus and the Playwrights Dialog. Their purpose is do add diversity to the content. Each speaker read 3 of these sentences, with each sentence being read

| Dialect | Region | Speakers | % in Dataset |
|---|---|---|---|
| dr1 | New England | 49 | 8% |
| dr2 | Northern | 102 | 16% |
| dr3 | North Midland | 102 | 16% |
| dr4 | South Midland | 100 | 16% |
| dr5 | Southern | 98 | 16% |
| dr6 | New York City | 46 | 7% |
| dr7 | Western | 100 | 16% |
| dr8 | Moved around | 33 | 5% |
| Total | | 630 | 100% |

Table A.1: TIMIT dialect regions and speakers

| Sentence Type | Sentences | Speakers | Total | Sentences/Speaker |
|---|---|---|---|---|
| Dialect (SA) | 2 | 630 | 1260 | 2 |
| Compact (SX) | 450 | 7 | 3150 | 5 |
| Diverse (SI) | 1890 | 1 | 1890 | 3 |
| Total | 2342 | | 6300 | 10 |

Table A.2: TIMIT sentence types and speakers

only by a single speaker.

The training set is selected and differentiated from the other sets based on the following criteria [16]:

1. Roughly 20 to 30% of the corpus should be used for testing purposes, leaving the remaining 70 to 80% for training.

2. No speaker should appear in both the training and testing portions.

3. All the dialect regions should be represented in both subsets, with at least 1 male and 1 female speaker from each dialect.

4. The amount of overlap of text material in the two subsets should be minimized; if possible no texts should be identical.

5. All the phonemes should be covered in the test material, preferably each phoneme should occur multiple times in different contexts.

The Core Test set contains 24 speakers, 2 male and 1 female from each dialect region. Each speaker read a different set of SX sentences. Thus it contains 192 sentences, 5 SX and 3 SI for each speaker.

The Extended Test set includes the sentences from all speakers that read any of the SX texts included in the core test set. In addition, no sentence text appears

| Class | Attributes |
|---|---|
| **Sonority** | Obstruent (OBS), Silence (SIL), Sonorant (SON), Syllabic (SYL), Vowel (VOW) |
| **Voicing** | NA, Voiced (VCD), Voiceless (VLS) |
| **Manner** | Approximant (APR), Flap (FLP), Fricative (FRI), NA, Nasal (NAS), NasalFlap (NF), Stop-Closure (STCL), Stop (STP) |
| **Place** | Alveolar (ALV), Dental (, Glottal, Labial, Lateral, NA, Palatal, Rhotic, Velar |
| **Height** | High, Low-High, Low, Mid-High, Mid, NA |
| **Backness** | Back, Back-Front, Central, Front, NA |
| **Roundness** | NA, NonRound, NonRound-Round, Round-NonRound, Round |
| **Tenseness** | Lax, NA, Tense |

Table A.3: Phonological events (IPA attributes)

in both the training and test sets. Thus it contains a total of 168 speakers and 1344 utterances, accounting for about 27% of the total speech material.

The Cross-Validation set contains a selection of 400 sentences from the Extended Test set.

## A.2 Phonetic Units to Phonological Events Mapping

The 61 TIMIT phonemes are associated with specific phonological events. The 8 phonological classes contain in total 44 phonological events as presented in Table 2.1. In this Section, the detailed mapping of each of the 61 phonemes ot its corresponding phonological events from each class is presented. The phonological events are given a codename that acts as a short-name for each event. In Table A.3, we can see the different phonological classes and events together with their codenames. Next the actual mapping is presented in Table A.4. Each phoneme is associated at most with one event from each phonological class. Some event classes are not applicable for certain phonemes and this is indicated with the codename ”NA”.

## A.3 Phoneme label reduction rules

During the experimental setups in this thesis, the TIMIT phoneme set was modified according to well established practices in speech recognition. The 61 phoneme set was reduced to either 48 or 39 phonemes, by using substitution and merging of existing phonemes and pairs of phonemes. The set consisting of 48 phonemes were used during the decoding step of the phoneme recognition task. The smaller set of 39 phonemes was used during the evaluation of the results. The rules to create each reduced set from the original 61 phoneme set are presented in this section.

| Phone | Sonority | Voicing | Manner | Place | Height | Backness | Roundness | Tenseness |
|---|---|---|---|---|---|---|---|---|
| aa | VOW | VCD | NA | NA | LOW | BAK | NRND | TEN |
| ae | VOW | VCD | NA | NA | LOW | FRT | NRND | TEN |
| ah | VOW | VCD | NA | NA | MID | CEN | NRND | TEN |
| ao | VOW | VCD | NA | NA | LOW | BAK | RND | TEN |
| aw | VOW | VCD | NA | NA | LOHI | BAK | NRRD | TEN |
| ax | VOW | VCD | NA | NA | MID | CEN | NRND | LAX |
| ax-h | VOW | VLS | NA | NA | MID | CEN | NA | LAX |
| axr | SYL | VCD | APR | RHO | NA | BAK | RND | LAX |
| ay | VOW | VCD | NA | NA | LOHI | BKFR | NRND | TEN |
| b | OBS | VCD | STP | LAB | NA | NA | NA | NA |
| bcl | OBS | VCD | STCL | LAB | NA | NA | NA | NA |
| ch | OBS | VLS | STP | PAL | NA | NA | NA | NA |
| d | OBS | VCD | STP | ALV | NA | NA | NA | NA |
| dcl | OBS | VCD | STCL | ALV | NA | NA | NA | NA |
| dh | OBS | VCD | FRI | DEN | NA | NA | NA | NA |
| dx | SON | VCD | FLP | ALV | NA | NA | NA | NA |
| eh | VOW | VCD | NA | NA | MID | FRT | NRND | LAX |
| el | SYL | VCD | APR | LAT | NA | BAK | NRND | NA |
| em | SYL | VCD | NAS | LAB | NA | NA | NA | NA |
| en | SYL | VCD | NAS | ALV | NA | NA | NA | NA |
| eng | SYL | VCD | NAS | VEL | NA | NA | NA | NA |
| epi | SIL | NA | NA | NA | NA | NA | NA | NA |
| er | SYL | VCD | APR | RHO | NA | BAK | RND | TEN |
| ey | VOW | VCD | NA | NA | MID | FRT | NRND | TEN |
| f | OBS | VLS | FRI | LAB | NA | NA | NA | NA |
| g | OBS | VCD | STP | VEL | NA | NA | NA | NA |
| gcl | OBS | VCD | STCL | VEL | NA | NA | NA | NA |
| h# | SIL | NA | NA | NA | NA | NA | NA | NA |
| hh | OBS | VLS | FRI | GLT | NA | NA | NA | NA |
| hv | OBS | VCD | FRI | GLT | NA | NA | NA | NA |
| ih | VOW | VCD | NA | NA | HI | FRT | NRND | LAX |
| ix | VOW | VCD | NA | NA | HI | CEN | NRND | LAX |
| iy | VOW | VCD | NA | NA | HI | FRT | NRND | TEN |
| jh | OBS | VCD | STP | PAL | NA | NA | NA | NA |
| k | OBS | VLS | STP | VEL | NA | NA | NA | NA |
| kcl | OBS | VLS | STCL | VEL | NA | NA | NA | NA |
| l | SON | VCD | APR | LAT | NA | NA | NA | NA |
| m | SON | VCD | NAS | LAB | NA | NA | NA | NA |
| n | SON | VCD | NAS | ALV | NA | NA | NA | NA |
| ng | SON | VCD | NAS | VEL | NA | NA | NA | NA |
| nx | SON | VCD | NF | ALV | NA | NA | NA | NA |
| ow | VOW | VCD | NA | NA | MID | BAK | RND | TEN |
| oy | VOW | VCD | NA | NA | MDHI | BKFR | RDNR | TEN |
| p | OBS | VLS | STP | LAB | NA | NA | NA | NA |
| pau | SIL | NA | NA | NA | NA | NA | NA | NA |

Continued on next page

**Table A.4 – continued from previous page**

| Phone | Sonority | Voicing | Manner | Place | Height | Backness | Roundness | Tenseness |
|---|---|---|---|---|---|---|---|---|
| pcl | OBS | VLS | STCL | LAB | NA | NA | NA | NA |
| q | OBS | VLS | STP | GLT | NA | NA | NA | NA |
| r | SON | VCD | APR | RHO | NA | NA | NA | NA |
| s | OBS | VLS | FRI | ALV | NA | NA | NA | NA |
| sh | OBS | VLS | FRI | PAL | NA | NA | NA | NA |
| t | OBS | VLS | STP | ALV | NA | NA | NA | NA |
| tcl | OBS | VLS | STCL | ALV | NA | NA | NA | NA |
| th | OBS | VLS | FRI | DEN | NA | NA | NA | NA |
| uh | VOW | VCD | NA | NA | HI | BAK | RND | LAX |
| uw | VOW | VCD | NA | NA | HI | BAK | RND | TEN |
| ux | VOW | VCD | NA | NA | HI | CEN | RND | LAX |
| v | OBS | VCD | FRI | LAB | NA | NA | NA | NA |
| w | SON | VCD | APR | LAB | NA | NA | NA | NA |
| y | SON | VCD | APR | PAL | NA | NA | NA | NA |
| z | OBS | VCD | FRI | ALV | NA | NA | NA | NA |
| zh | OBS | VCD | FRI | PAL | NA | NA | NA | NA |

Table A.4: Phoneme to Phonological events mapping

Starting from the 48 phoneme set we have the following rules:

| | | |
|---|---|---|
| **q** | ⟶ | **NULL** |
| **pcl** | ⟶ | **cl** |
| **tcl** | ⟶ | **cl** |
| **kcl** | ⟶ | **cl** |
| **qcl** | ⟶ | **cl** |
| **bcl** | ⟶ | **vcl** |
| **dcl** | ⟶ | **vcl** |
| **gcl** | ⟶ | **vcl** |
| **h#** | ⟶ | **sil** |
| **#h** | ⟶ | **sil** |
| **pau** | ⟶ | **sil** |

The first phoneme in fact is deleted - substitute ”q” with NULL. The unvoiced closures are substituted by the label for unvoiced closure. Voiced closures are substituted by the label for voiced closure. Finally all silence labels are substitutes by a common label.

The smaller 39 phoneme set is created by the following rules.

| | | |
|---|---|---|
| **q** | ⟶ | **NULL** |
| **b bcl** | ⟶ | **b** |
| **d dh** | ⟶ | **d** |
| **t tcl** | ⟶ | **t** |
| **p pcl** | ⟶ | **p** |
| **k kcl** | ⟶ | **k** |
| **g gcl** | ⟶ | **g** |
| **pcl** | ⟶ | **cl** |
| **tcl** | ⟶ | **cl** |
| **kcl** | ⟶ | **cl** |
| **qcl** | ⟶ | **cl** |
| **bcl** | ⟶ | **vcl** |
| **dcl** | ⟶ | **vcl** |
| **gcl** | ⟶ | **vcl** |
| **h#** | ⟶ | **sil** |
| **#h** | ⟶ | **sil** |
| **pau** | ⟶ | **sil** |
| **el** | ⟶ | **l** |
| **en** | ⟶ | **n** |
| **zh** | ⟶ | **sh** |
| **ao** | ⟶ | **aa** |
| **ix** | ⟶ | **ih** |
| **ax** | ⟶ | **ah** |
| **ax − h** | ⟶ | **ah** |
| **axr** | ⟶ | **er** |
| **ux** | ⟶ | **uw** |
| **nx** | ⟶ | **n** |
| **hv** | ⟶ | **hh** |
| **eng** | ⟶ | **ng** |
| **em** | ⟶ | **m** |

The rules are applied in the order they appear. TIMIT breaks stop consonants to stop closure and stop release. When a stop is met, the two region labels (e.g. pcl and p) are substituted by a common label, the stop release label. After these rules are applied, the stop closure to either voiced or unvoiced closure rules are applied. This happens when a stop closure label is met, without belonging to a full stop consonant closure release pair. Then some phonemes are simplified to a common label. Finally the nasals are simplified.

# Appendix B

# Signal processing routines

## B.1 Low-level speech event computation

### Degree of Voicing

The degree of voicing is a score between 0 and 1. To determine the degree of voicing of a speech frame, a basic voiced / unvoiced routine is used:

1. subtract the mean
2. center-clipping
3. perform scaled autocorrelation
4. searching for the max autocorrelation in the range 50-500 Hz

### Short-time zero crossing count

The zero crossing count (ZCC ) gives a measure of the noisiness of the signal by providing a measure of the weighted average of the spectral energy distribution of the waveform. However, the ZCC has the advantage of not requiring an FFT computation. The following formula is used for calculation:

$$ZCC = \sum_{n=1}^{N} (|sgn(x(n)) - sgn(x(n-1)|) \tag{B.1}$$

### Spectral Roll-off Point

The spectral roll-off point is defined as the frequency below which 95% of the spectral power is concentrated. The formula used for computation is:

$$\sum_{k=1}^{Rss} P(k) = 0.95 \sum_{k=1}^{K} P(k) \tag{B.2}$$

where P(k) is the power spectrum of the frame at a frequency bin k. Spectral roll-off point is consistently higher for unvoiced speech than for voiced.

### Spectral Centroid

The Spectral Centroid (Css) is the frame-to-frame difference of the center of mass of power spectrum

$$C_{ss}(i) = \frac{\sum^{K} kP_i(k)}{\sum^{K} P_i(k)} \tag{B.3}$$

where $P_i(k)$ is the power spectrum for frame $i$ and frequency $k$, and $K$ is the total number of frequency bins.

### Spectral Flux

The Spectral Flux is the difference between the amplitudes of successive magnitude spectra:

$$F_{ss}^{0}(i) = \sum_{k=0}^{K} [M_i(k) - M_{i-1}(k)]^2. \tag{B.4}$$

where $M_i(k)$ and $M_{i-1}(k)$ are the magnitudes of the spectra for frames $i$ and $i-1$. Fss measures the amount of spectral change between successive frames.