This is to certify that the

thesis entitled

FEASIBILITY STUDY OF VOICE ACCESS TO COMPUTERS
FOR PEOPLE WITH LIMITED SPEECH

presented by

LAMBERT MATHIAS

has been accepted towards fulfillment
of the requirements for

MS degree in ELECTRICAL ENGINEERING

John R. Deller Jr.
Major professor

Date 8/15/02

FEASIBILITY STUDY OF VOICE ACCESS TO COMPUTERS FOR PEOPLE WITH LIMITED SPEECH

By

Lambert Mathias

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering

2002

ABSTRACT

FEASIBILITY STUDY OF VOICE ACCESS TO COMPUTERS FOR PEOPLE WITH LIMITED SPEECH

By

Lambert Mathias

Dysarthria is a general term for a speech disorder in which speech is slow, weak, imprecise or uncoordinated. Commercially available automatic speech recognition (ASR) systems cannot reliably recognize dysarthric speech due to the inherent variability in such utterances. People with dysarthria generally lack articulatory precision. Simple phonemes like vowels are physically the easiest sounds to produce, since they do not require dynamic movement of the vocal system. This research is primarily a feasibility study investigating the reliability of vowel-based phoneme recognition of dysarthric speech. The goal is to evaluate if ASR algorithms could be used to reliably differentiate among the different vowel sounds produced by dysarthric speakers. The intended purpose is to provide personal computer based access methods for people with dysarthric speech. In this work, the hidden Markov model (HMM) is the basic technological approach adopted in developing the speech recognition algorithms, and all the experimental results quantifying the feasibility of these algorithms are presented.

ACKNOWLEDGEMENTS

I am indebted to my research advisor Professor John R. Deller Jr. for his excellent guidance and support throughout the course of my graduate studies. I would like to thank him for his valuable insights, and technical expertise, which were the driving force behind this research. I would also like to express my gratitude to my committee members, Professor Percy A. Pierre and Dr. Michael Seadle for their insigthful comments. This work was supported by the National Institutes of Health under Small Business Innovation Research Grant No. 1-R43-HD36164-01A1. The Principal Contractor in the SBIR award is InvoTek, Inc. of Alma, Arkansas. A special thanks is due to InvoTek's President Tom Jakobs for his generous support, encouragement, technical advice, and for superb management of the project. The MSU team also appreciates the synergistic interactions with, and clinical expertise of, Professor David Beukelman, Ms. Susan Fager, Ms. Cara Ullman, and other staff members, and all of the participants in the study, at the Madonna Rehabilitation Hospital in Lincoln, Nebraska, and at the Department of Special Education of the University of Nebraska.

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES ix

1 INTRODUCTION 1

1.1 Background 1
1.2 The Voice Access System Project 2
1.3 Research Objectives and Goals 3

2 THEORETICAL BACKGROUND 5

2.1 Overview 5
2.2 The Hidden Markov Model 6
2.3 HMM Training Algorithm 10
2.3.1 The General Training Problem 10
2.3.2 The Baum-Welch Training Algorithm 11
2.4 HMM Recognition Algorithm 14
2.4.1 The General Classification Problem 14
2.4.2 The Viterbi Recognition Algorithm 16
2.5 Speech Analysis and Feature Extraction 18
2.5.1 Short-term Processing of Speech 18
2.5.2 Mel-scale Filter-Bank Processing 20
2.5.3 Mel-frequency Cepstrum Coefficients (MFCC) 23
2.5.4 Log Energy, Delta and Acceleration Coefficients 23
2.6 Language Modeling 24
2.7 Phonemes and Phonetic Transcription 26

3 IMPLEMENTATION DETAILS...... 29

3.1 Hardware and Software Tools 29
3.2 Speech Databases 29
3.2.1 TIMIT Speech Corpus 29
3.2.2 The Dysarthric Speech Corpus 30
3.3 Creating the Observation Strings 33
3.3.1 Phoneme Extraction from the TIMIT Speech Corpus 33
3.3.2 Phoneme Extraction from the Dysarthric Database 34
3.3.3 Features Comprising the Observation Strings 34
3.4 Implementation of the Phoneme Recognizer 35
3.4.1 HMM Topology 37

3.4.2 HMM Training and Testing 38
3.5 Bigram Language Model Implementation 40
4 EXPERIMENTAL EVALUATION........ 45
4.1 Overview 45
4.2 Experiments with Normal Speech 45
4.3 Experiments with Dysarthric Speech 47
4.3.1 Classification Experiment 48
4.3.2 Conclusion 51
4.4 Experiments with Language Modeling 52
4.4.1 Speaker-Independent Bigram Language Model 53
4.4.2 Speaker-Dependent Bigram Language Model 58
4.4.3 Conclusion 67
4.5 HMMs Trained on Dysarthric Speech 67
4.5.1 Classification Experiment 68
4.5.2 Conclusion 71
5 CONCLUSIONS AND FUTURE RESEARCH 72
5.1 Conclusions 72
5.2 Future work 74
BIBLIOGRAPHY 76

LIST OF TABLES

Table 2.1. Phonetic transcriptions of vowels. 28

Table 3.1. Isolated utterances in the dysarthric speech database. 32

Table 3.2. Accuracy of the vowel phoneme HMM on TIMIT training data. 39

Table 4.1. Recognition results with normal speech. 47

Table 4.2. Distribution of utterances in the dysarthric speech database. 48

Table 4.3. Confusion matrix for Speaker 4 (shaded cells indicate reliable vowel sound).49

Table 4.4. Confusion matrix for Speaker 1 (shaded cells indicate reliable vowel sound).50

Table 4.5. Confusion matrix for Speaker 2 (shaded cells indicate reliable vowel sound).50

Table 4.6. Confusion matrix for Speaker 3 (shaded cells indicate reliable vowel sound).51

Table 4.7. Speaker-independent Bigram LM. 53

Table 4.8. Confusion matrix with speaker-independent LM for Speaker 1 54

Table 4.9. Confusion matrix without speaker-independent LM for Speaker 1. 55

Table 4.10. Confusion matrix with speaker-independent LM for Speaker 2 55

Table 4.11. Confusion matrix without speaker-independent LM for speaker 2. 55

Table 4.12. Confusion matrix with speaker-independent LM for Speaker 3 56

Table 4.13. Confusion matrix without speaker-independent LM for Speaker 3. 56

Table 4.14. Confusion matrix with speaker-independent LM for Speaker 4 56

Table 4.15. Confusion matrix without speaker-independent LM for Speaker 4. 57

Table 4.16. Error rates for the recognition task with the speaker-independent LM. 57

Table 4.17. Bigram LM for Speaker 1. 59

Table 4.18. Confusion matrix with speaker-dependent LM for Speaker1 (shaded cells indicate reliable vowel sound). 59

Table 4.19. Confusion matrix without speaker-dependent LM for Speaker 1(shaded cells indicate reliable vowel sound). 60

Table 4.20. Error rate for Speaker 1. 60

Table 4.21. Bigram speaker-dependent LM for Speaker 2. 61

Table 4.22. Confusion matrix with speaker-dependent LM for Speaker 2 (shaded cells indicate reliable vowel sound). 61

Table 4.23. Confusion matrix without speaker-dependent LM for Speaker 2 (shaded cells indicate reliable vowel sound). 62

Table 4.24. Error rate for Speaker 2. 62

Table 4.25. Bigram speaker-dependent LM for Speaker 3. 63

Table 4.26. Confusion matrix with speaker-dependent LM for Speaker 3 (shaded cells indicate reliable vowel sound). 63

Table 4.27. Confusion matrix without speaker-dependent LM for Speaker 3 (shaded cells indicate reliable vowel sound). 64

Table 4.28. Error rate for Speaker 3. 64

Table 4.29. Bigram speaker-dependent LM for Speaker 4. 65

Table 4.30. Confusion matrix with speaker-dependent LM for Speaker 4 (shaded cells indicate reliable vowel sound). 65

Table 4.31. Confusion matrix without speaker-dependent LM for Speaker 4 (shaded cells indicate reliable vowel sound). 66

Table 4.32. Error rate for Speaker 4. 66

Table 4.33. Partitioning the dysarthric speech database into training and test sets. 68

Table 4.34. Confusion matrix for Speaker 1 (shaded cells indicate reliable vowel sound). 69

Table 4.35. Confusion matrix for Speaker 2 (shaded cells indicate reliable vowel sound). 69

Table 4.36. Confusion matrix for Speaker 3 (shaded cells indicate reliable vowel sound). 70

Table 4.37. Confusion matrix for Speaker 4 (shaded cells indicate reliable vowel sound). 70

Table 5.1. Row-column access for Speaker 1 using reliable vowel phonemes from (a) recognition task without LM (b) recognition task with LM. 73

LIST OF FIGURES

Figure 2.1. Three-state left-to-right HMM. 9

Figure 2.2. Computation of $P[O \mid \lambda_k]$ using the forward recursion of the F-B algorithm. 16

Figure 2.3. The Viterbi Algorithm. 17

Figure 2.4. The mel scale. 21

Figure 3.1. Example of MFCC feature extraction: (a) The speech waveform for the phoneme utterance 'AE'. (b) The 39-dimensional MFCC parameters. 35

Figure 3.2. The phoneme classifier. 36

Figure 3.3. Example of a bigram LM using seven vowel phonemes. 41

Figure 3.4. LM-based Viterbi search grid. 42

Figure 3.5. LM-based Viterbi search algorithm. 43

Figure 4.1. The *a priori* distribution of the seven vowels in the TIMIT test set. 46

1 Introduction

1.1 Background

Dysarthria is a general term for a speech disorder in which speech is slow, weak, imprecise or uncoordinated. This disorder is commonly associated with other general neuromotor disabilities (Parkinson's disease, cerebral palsy, etc). People with dysarthria may have difficulty in making themselves understood, or in reliably controlling environmental and communication aids. Many individuals with dysarthric speech, who use augmentative and alternative communication (AAC) devices have normal or exceptional intellects, reading and language skills, and would strongly prefer to use their residual speech, however limited [21]. AAC devices using speech technologies have the potential of not only serving vocational and educational needs but can also help satisfy such individuals' social communication needs.

Current commercially available automatic speech recognition (ASR) products (e.g., Dragon Dictate and IBM Via Voice) are designed for individuals whose speech is not impaired. Commercial systems may be able to recognize the speech of individuals with mild impairments, or individuals who have received sufficient training to alter their articulatory patterns to achieve improved machine recognition rates [25] [26]. However, the use of off-the-shelf commercial recognizers for people with dysarthria has not been particularly successful, with recognition rates for severely dysarthric people varying anywhere between 18-85% [27] [28]. Severe dysarthria is still a challenge for most commercial recognizers largely due to the extraordinary variability in dysarthric speech, and also because commercial recognizer systems are optimized for the mass market. The variability in dysarthric speech differs not only across individuals, but also for a particular individual depending upon the amount of stress, the time of day and other personal and environmental conditions. The inconsistency of dysarthric speech makes recognition of dysarthric speech inherently a different problem than that of normal speech.

A different perspective using ASR for people with severe dysarthria is the use of a small set of utterances that can be reliably recognized. What is needed is a speech recognition system that is optimized for people who are capable of producing distinct vocalizations, even though these vocalizations may not be meaningful in normal speech. This approach can help individuals with dysarthria to use communication aids more effectively and improve their performance of job-related tasks. This research is primarily a feasibility study investigating the reliability of vowel-based phoneme recognition system for dysarthric speech. The phoneme-level recognizer developed must be capable of reliably differentiating among the different vowel sounds produced by dysarthric speakers.

1.2 The Voice Access System Project

This thesis was written as part of a NIH-sponsored SBIR Phase 1 joint project between Invotek Inc. , and the Speech Processing Laboratory at Michigan State University. The goal of the Voice Access System (VAS) project is to provide persons who have physical disabilities and unintelligible speech with an access method for assistive devices that significantly reduces the physical fatigue experienced during device access. An important feature of this voice access system is that it does not attempt to recognize a particular sound sequence. The only criterion for recognition is that the system be able to consistently discriminate among the sounds used for access. The VAS offers significant advantages over other more physically-demanding access methods.

1.3 Research Objectives and Goals

People with dysarthria generally lack articulatory precision. Simple 'steady-state' phonemes like vowels are physically the easiest sounds to produce, since they do not require dynamic movement of the vocal system [22]. In this research, we investigate the reliability of recognizing vowel utterances of dysarthric speakers.

Hidden Markov Models (HMMs) are known to be quite efficient in speech recognition related tasks [29]. In this study, speaker-independent HMM models of seven representative vowel sounds, trained on normal speech, are used to build a phoneme classifier. The test utterances consist of multiple utterances of the seven vowels spoken by dysarthric individuals The subjects for this research have been provided by the Madonna Rehabilitation Hospital, Nebraska and the Department of Special Education of the University of Nebraska. The dysarthric test utterances are passed through the phoneme classifier and the classification results are used to compute a confusion matrix.

The confusion matrix gives information about the number of test utterances that are classified as belonging to each of the HMMs representing the different vowels. The confusion matrix is used to evaluate which vowel sounds are most reliably recognized for a particular speaker. A similar recognition experiment is performed using HMMs trained on utterances from the dysarthric speech database. Furthermore, a bigram language model is added to the phoneme classifier to evaluate its effect on vowel recognition accuracies. Both, the speaker-dependent and speaker-independent bigram LM is considered in this research.

Chapter 2, introduces the concept of HMMs, the algorithms commonly used for speech recognition and training, and the extraction of observation strings from raw speech. Chapter 3, discusses the actual implementation of the vowel-based phoneme recognition system, the feature extraction process, the speech corpora used for training and testing, the Baum-Welch training algorithm, the Viterbi recognition algorithm, and the bigram language model implementation. Chapter 4, discusses the results of the phoneme classification experiments. The focus is on which vowels can be most reliably recognized for a particular dysarthric speaker with minimum amount of confusability, and whether adding language information to the recognition task can help improve recognition rates. The final chapter, Chapter 5, summarizes the research conclusions, and outlines the course of further research.

2 Theoretical Background

2.1 Overview

Speech recognition is a difficult task, given the variability associated with speech. A good recognition system must account for all the dynamics and uncertainties in speech in order to achieve reasonable accuracy. Stochastic methods provide adequate models to characterize much of the variability in speech. Furthermore, the question whether a given utterance belongs to a certain class becomes that of hypothesis testing, a statistical decision theory problem. Hidden Markov modeling is a parametric technique that has been successfully applied to speech recognition with considerable success [7] [8]. The HMM uses Markov chains to model the changing statistical characteristics that exist in the actual observations of speech signals. The HMM also has inherent time normalization properties. In terms of implementation, the HMM lends itself easily to computation on sequential machines. HMMs are iteratively trained using one of two iterative algorithms and variations: Viterbi decoding and Baum-Welch re-estimation [9] [10] [11].

However there is some front-end processing involved on the speech data to map it to a feature space that completely characterizes the dynamics of the speech waveform. The main purpose of the front-end processing is to derive feature vectors such that different vectors belonging to a given class of utterance are similar to each other, while feature vectors belonging to different utterances are maximally different from one another. The feature extraction is carried out over small segments of speech called "frames" over which the speech signal can be reasonably assumed to be stationary. The feature extraction process serves to isolate the effect of the environment noise and the speaker identity on the speech utterance, thereby enhancing the speaker independence of the system and making it more robust to environmental changes. The procedure also reduces the amount of data to be managed by the speech recognition and training systems. The feature vectors thus completely represent the temporal and spectral behavior of a short segment of the acoustical speech input. The ultimate goal of the front-end is to estimate parameters that effectively discriminate among the different phonetic units, while reducing the computational demand on the classifier. The mel-frequency cepstrum is the most commonly used feature space in characterizing the speech signal. The different aspects of speech recognition and training are discussed in the following sections. For detailed information, the reader is referred, for example to the text by Deller *et al.* [2] and the paper by Rabiner [3].

2.2 The Hidden Markov Model

Signal modeling based on HMMs is a technique that extends conventional stationary spectral analysis principles to the analysis of time-varying signals [4]. HMMs use a Markov state process to model the changing statistical characteristics that are probabilistically manifested through actual observations. The state sequence is hidden, and is observed through another set of observable stochastic processes. The observable output probabilities associated with each of the hidden states are characterized by either discrete probability distributions or continuous probability density functions. In this thesis, the latter approach is used. This class of HMMs is called continuous hidden Markov models (CHMM). The advantage of CHMMs is that the observations are continuous signals or vectors, and therefore do not suffer from degradation due to quantization errors as in the discrete case. The model structure usually adopted for speech recognition is a left-to-right or Bakis structure [12]. In the Bakis model, states are aligned so that only "left-to-right" transitions are allowed. Such a model is appropriate to characterize speech signals whose dynamics progress sequentially along a timeline. Based on the above discussion we can now formally define an HMM.

A HMM is characterized by the following sets of quantities :

1. N_{state}, the number of states in the model. We denote the individual states as $S = \{1, 2, 3, \ldots, N_{\text{state}}\}$, and the state at any time t [1] as s_t.

2. The transition probability matrix, $A = \{a_{ij}\}$ where

$$a_{ij} = P[s_{t+1} = i \mid s_t = j], \qquad 1 \le i, j \le N_{\text{state}}, \text{ for any } t. \tag{2.2.1}$$

[1] The time t here represents the re-indexing of the original sample sequence of speech, so that the frames can now be indexed by sequential integers. For detailed information the reader is referred to Chapter 4 in [2].

The state entered at time $t+1$ depends only on the previous state at time t. The state sequence is therefore characterized by a stationary (or homogenous), first-order, Markov chain.

3. The initial state distribution $\pi = \{\pi_i, \quad 1 \le i \le N_{\text{state}}\}$ where

$$\pi_i = P[s_1 = i], \qquad\qquad 1 \le i \le N_{\text{state}}. \tag{2.2.2}$$

The initial state distribution and the transition probability matrix completely specify the probability of residing in any state at any time.

4. Let the output observation vector at any time t be denoted by o_t where $o_t \in \mathbb{R}^p$. In HMM literature, the feature vectors extracted from the speech utterance are referred to as the output observation vectors, since they represent the information that is "observed" from the incoming speech utterance. p denotes the dimension of the feature vectors that have been extracted from the raw speech signal. The output observation probability distribution in state j at any time t is denoted by, $b_j(o_t)$ where

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm} \mathcal{n}(o_t; \mu_{jm}, C_{jm}). \tag{2.2.3}$$

$b_j(o_t)$ represents a multivariate Gaussian mixture density function of M mixtures, and $\mathcal{n}$ is a single Gaussian probability density function (pdf) given by

$$\mathcal{n}(o_t;\mu_{jm},C_{jm}) = \frac{1}{\sqrt{(2\pi)^p \mid C_{jm} \mid}} \exp\left(-\frac{1}{2}(o_t - \mu_{jm})C_{jm}{}^{-1}(o_t - \mu_{jm})' \right). (2.2.4)$$

with μ_{jm} and C_{jm} being the mean vector and the covariance matrix of the m^{th} mixture. c_{jm} is the mixture coefficient of the m^{th} mixture in state j. The mixture coefficients must be nonnegative and satisfy the constraint

$$\sum_{m=1}^{M} c_{jm} = 1, \qquad 1 \le j \le N_{\text{state}}. \qquad (2.2.5)$$

From the above discussion, a HMM can be represented in a compact form as

$$\lambda = \left\{ N_{\text{state}}, \pi_1, A, \{b_j(o_t), \; 1 \le j \le N_{\text{state}}\} \right\}. \qquad (2.2.6)$$

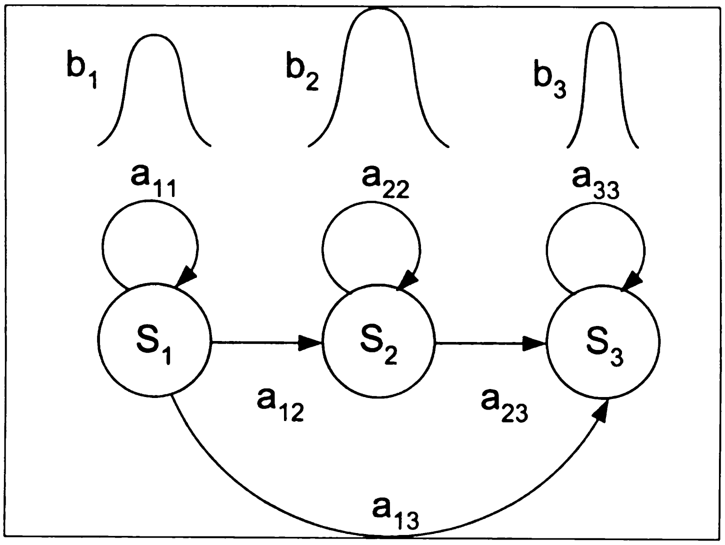


Figure 2.1. Three-state left-to-right HMM.

The output observation probabilities of the HMMs are assumed to be conditionally independent, i.e., the output probability depends only on the state regardless of when and how the state is entered. Formally, this means that $b_j(o_t)$ is independent of time t as implied by (2.2.3) and (2.2.4). The first-order Markov chain and the conditional output-independence assumptions reduce the number of free parameters, and make learning and decoding algorithms efficient.

Having formally described the HMM, we now examine two central issues on the training and use of the HMM. These are the following :

1. Given a series of training observations for a given utterance how do we estimate the optimum state transition matrix, A, and the observation pdfs, $b_j(o)$ for each state j? This represents the HMM *training* problem.

2. Given a trained HMM, how do we find the likelihood that it produced an incoming speech observation sequence? This constitutes the *recognition* problem.

2.3 HMM Training Algorithm

2.3.1 The General Training Problem

The training problem involves choosing the right HMM parameters for a given training set (the data that are used to train the HMM) using an optimization criterion. The maximum likelihood (ML) criterion is used here. Formally, the training procedure involves finding

$$\lambda^* = \arg\max_{\lambda} P[O \mid \lambda]. \tag{2.3.1}$$

where λ represents the set of HMM parameters (2.2.6), and λ^* the optimal set. $O = \{o_1, o_2, ..., o_t, ..., o_T\}$ is the given observation sequence and $P[O \mid \lambda]$ is the likelihood score of that sequence given the model λ. T here denotes the total number of observations. There is no analytical way to solve for the model λ that maximizes λ^*. However, we can choose model parameters such that the likelihood achieves a local maximum using an iterative procedure, like the Baum-Welch algorithm.

2.3.2 The Baum-Welch Training Algorithm

Given, an observation sequence O, we need to determine the parameters of a HMM that satisfy the ML criterion. We start out with an initial model λ_0 of form (2.2.6) with arbitrary parameters. The iterative procedure used to find the ML model λ^* is called the Baum-Welch algorithm, also known as the forward-backward (F-B) reestimation procedure [13]. To formally develop this algorithm, we define the forward probability $\alpha_t(i)$ that is the joint probability of the partial observation sequence from time 1 to time t and the state j that is reached at time t from all possible states i at time $t-1$. This can be calculated iteratively as follows :

$$\alpha_t(j) = [\sum_{i=1}^{N_{\text{state}}} \alpha_{t-1}(i)a_{ij}]b_j(o_t), \quad \text{for any } t. \tag{2.3.2}$$

Similarly, we define the backward probability as the joint probability of the partial observation sequence from time $t+1$ to the final observation at time T, given state i at time t. This is calculated iteratively as follows

$$\beta_t(i) = \sum_{j=1}^{N_{\text{state}}} a_{ij}b_j(o_{t+1})\beta_{t+1}(j), \quad \text{for any } t. \tag{2.3.3}$$

The resultant probability $P[O \mid \lambda]$ can now be calculated as

$$P[O \mid \lambda] = \sum_{i=1}^{N_{\text{state}}} \alpha_t(i)\beta_t(i), \quad \text{for any } t. \tag{2.3.4}$$

For an HMM with M mixtures, the means, covariance matrices, mixture weights and transition probabilities are re-estimated as follows :

$$\bar{\mu}_{jm} = \frac{\sum_{t=1}^{T} \sigma_t(jm)o_t}{\sum_{t=1}^{T} \sigma_t(jm)} \tag{2.3.5}$$

$$\bar{C}_{jm} = \frac{\sum_{t=1}^{T} \sigma_t(jm)(o_t - \bar{\mu}_{jm})(o_t - \bar{\mu}_{jm})'}{\sum_{t=1}^{T} \sigma_t(jm)} \tag{2.3.6}$$

$$\bar{c}_{jm} = \frac{\sum_{t=1}^{T} \sigma_t(jm)}{\sum_{t=1}^{T} \delta_t(j)} \tag{2.3.7}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)} \tag{2.3.8}$$

where $\sigma_t(jm)$ denotes the probability of the observation sequence occupying the m^{th} mixture component of state j at time t, and $\delta_t(j)$ denotes the probability of the observation sequence occupying state j at time t. They are related as follows

$$\delta_t(j) = \sum_{m=1}^{M} \sigma_t(jm) = \sum_{m=1}^{M} \frac{1}{P} \sum_{i=1}^{N_{\text{state}}} \alpha_{t-1}(i) a_{ij} c_{jm} b_{jm}(o_t) \beta_t(j), \qquad \text{for any } t \tag{2.3.9}$$

The re-estimated parameters comprise a new HMM model denoted by say, $\bar{\lambda}$. We can now calculate $P[O \mid \bar{\lambda}]$ as in (2.3.4). If we define some threshold value ε, we can run the above algorithm iteratively till we achieve convergence, i.e.

$$P[O \mid \bar{\lambda}] - P[O \mid \lambda] \leq \varepsilon \tag{2.3.10}$$

The HMM model always improves under the reestimation procedure unless its parameters already represent a local maximum. So, this algorithm does not necessarily give us the optimal model λ^*. Its is common practice, to run the Baum-Welch algorithm several times with different initial parameters and to take as the trained model the one that gives the maximum likelihood score.

The Viterbi algorithm can also be used for training of HMMs. However, in the Viterbi approach, the likelihood computation for estimating the HMM parameters is based only on the most probable sequence of states through the model. The Baum-Welch algorithm on the other hand is found to be more effective, precise and standard because it takes into account all the possible state sequences through the model, while estimating the HMM parameters [5]. The Baum-Welch algorithm has been implemented to train HMMs in this research.

2.4 HMM Recognition Algorithm

2.4.1 The General Classification Problem

Phoneme classification involves passing the given observation sequence through a set of, say, L_λ, given HMMs, where λ_k denotes the k^{th} HMM and $1 \le k \le L_\lambda$. In this research, a single HMM model is used to represent each of seven vowels. Since, we are using seven vowels in the phoneme classifier, $1 \le k \le 7$ in this case and we have a set of seven HMMs $\{\lambda_1, \lambda_2, ..., \lambda_7\}$ each HMM representing a vowel phoneme. The HMM with the maximum *a posteriori probability* represents the vowel type to which the given utterance belongs. Baye's well-known maximum *a posteriori* (MAP) decision rule for optimal classification is given by

$$k^* = \arg\max_k P[\lambda_k \mid O]. \tag{2.4.1}$$

Here $P[\lambda_k \mid O]$ is the *a posteriori* probability of the HMM λ_k given the observation sequence O. There is no easy way to estimate $P[\lambda_k \mid O]$ directly, but using conditional probability, we can express $P[\lambda_k \mid O]$ in terms of probabilities that can be estimated. Using conditional probability we can rewrite (2.4.1) as

$$k^* = \arg\max_k \frac{P[O \mid \lambda_k]P[\lambda_k]}{P[O]}. \tag{2.4.2}$$

The observation sequence O does not depend upon k and so we can leave it out of the classification rule. Let us also assume that all phoneme HMMs are equally likely. Then we can also drop $P[\lambda_k]$ out of the classification rule. (2.4.2) now reduces to

$$k^* = \arg\max_k P[O \mid \lambda_k]. \tag{2.4.3}$$

This is sometimes called the ***maximum likelihood*** (ML) classification rule and states that we can choose an HMM λ_k in the phoneme classification process as the winning HMM if it makes the observed data most likely. The probability $P[O \mid \lambda_k]$ can be calculated using the forward recursion of the F-B algorithm as shown in Figure 2.2.

Figure 2.2. Computation of $P[O \mid \lambda_k]$ using the forward recursion of the F-B algorithm.

Initialization: For all states $i = 1,2,...,N_{\text{state}}$

$$\alpha_t(i) = \pi_i b_i(o_1)$$

Recursion: For $t = 2,3,...,T$ and $j = 1,2,...,N_{\text{state}}$

$$\alpha_t(j) = \left[\sum_{i=1}^{N_{\text{state}}} \alpha_{t-1}(i) a_{ij} \right] b_j(o_t)$$

Termination:

$$P[O \mid \lambda_k] = \sum_{i=1}^{N_{\text{state}}} \alpha_T(i)$$

2.4.2 The Viterbi Recognition Algorithm

In order to decode an incoming observation sequence for correct classification we need to solve (2.4.3). Computing the likelihood $P[O \mid \lambda_k]$ using the forward recursion procedure shown in Figure 2.2 involves summing up the probabilities over all the possible paths (i.e. state sequences), which can be computationally intensive. Instead, we can compute the likelihood only for the best possible state sequence using the Viterbi algorithm. Using the Viterbi algorithm, given an incoming unknown observation sequence $O = \{o_1, o_2, ..., o_t, ..., o_T\}$ we can find the best possible state sequence $I^* = \{s_1^*, s_2^*, ..., s_t^*, ..., s_T^*\}$ that maximizes the probability $P[O \mid \lambda_k]$. The Viterbi algorithm is summarized in Figure 2.3.

The Viterbi algorithm is computationally efficient and is very easy to implement. In addition, it is possible to obtain the best state sequence along with the likelihood score, for a given observation sequence. Because of its advantages, the Viterbi algorithm is the most widely used in speech recognition systems.

Figure 2.3. The Viterbi Algorithm.

Initialization: For all states $i = 1,2,\ldots,N_{\text{state}}$,

$$\phi_1(i) = \pi_i b_i(o_1);$$
$$\psi_1(i) = 0;$$

Recursion: From time $t = 2,3,\ldots T$, for all states $j = 1,2,\ldots N_{\text{state}}$,

$$\phi_t(j) = \underset{1 \le i \le N_{\text{state}}}{Max} [\phi_{t-1}(i) a_{ij}] b_j(o_t)$$
$$\psi_t(j) = \underset{1 \le i \le N_{\text{state}}}{\arg\max} [\phi_{t-1}(i) a_{ij}]$$

Termination:

$$\text{The best score, } P^* = \underset{1 \le j \le N_{\text{state}}}{Max} [\phi_T(j)]$$
$$s_T^* = \underset{1 \le j \le N_{\text{state}}}{\arg\max} [\phi_T(j)]$$

Backtracking: From time $t = T-1, \ldots\ldots, 2, 1$

$$s_t^* = \psi_{t+1}(s_{t+1}^*)$$

$$\text{The best state sequence } I^* = \{s_1^*, s_2^*, \ldots, s_T^*\}$$

2.5 Speech Analysis and Feature Extraction

HMMs do not use raw data directly as an input. The speech signal is first sampled, digitized, and then transformed, into a multi-dimensional feature space using either time-domain or frequency-domain approaches. The feature extraction process transforms the one-dimensional speech signal into a multi-dimensional stream of feature vectors at a reduced sampling rate (frame rate) thereby resulting in the compression of data. Although the speech signal is non-stationary, it contains small portions of stationary spectral characteristics within a given utterance, giving rise to the term quasi-stationary. Hence, the feature analysis procedure must be applied over a window of speech short enough to be considered *stationary*, and at the same time long enough to make a good estimate of the speech signal parameters. The mel-cepstrum is the most popular feature space employed in many speech recognition systems [14]. Mel-frequency cepstrum coefficients (MFCC) feature extraction involves computing the short-term discrete Fourier transform (stDFT) of the given speech signal, then passing it through the mel-scale filter banks to compute the log total energy in each critical band and finally taking the inverse discrete Fourier transform (IDFT) of the mel scale coefficients. The feature extraction process is explained in the following sections.

2.5.1 Short-term Processing of Speech

The first step in feature extraction is the short-term processing of speech. This involves breaking the speech signal into a series of short segments known as the analysis frame.

Let $s(n)$ be a discrete time speech signal and $w(n)$ be the finite window with which we multiply the speech signal in order to get the speech frame $f(n;r)$ given by

$$f(n;r) = s(n)w(r-n). \qquad (2.5.1)$$

This new frame of speech is a sequence on n, which happens to be zero outside the short term $n \in [r - L_s + 1, r]$. Here L_s is the total length of the speech frame and r is the end position of the speech frame. Typically a Hamming window with impulse response of the form

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{L_s - 1}\right) \qquad n = 0, \ldots L_s - 1. \qquad (2.5.2)$$

is used for the short-term analysis of the speech signal. The length L_s of the window is typically less than the length of the speech utterance. Overlapping frames are used to smooth the frame-to-frame transition. The short-term Fourier transform of the speech signal is then obtained by using the stDFT,

$$S(d;r) = \sum_{n=r-N'+1}^{r} f(n;r)e^{-jd\frac{2\pi n}{N'}} \qquad d = 0, \ldots N' - 1. \qquad (2.5.3)$$

N' is the number of points used to compute the stDFT. The number of points used to compute the DFT is generally a power of two, so that it is easier and more efficient to implement. In the case that the frame length is not a power of two, zeros are padded at the end of the frame sequence to increase the resolution of the stDFT by increasing the number of points over which the stDFT is computed. Hence, N' is usually equal to the length of the speech frame after zero padding. The magnitude of the above stDFT denoted by $|S(d;r)|$ gives the magnitude spectrum of the speech frame for which the stDFT has been computed..

2.5.2 Mel-scale Filter-Bank Processing

The human ear resolves frequencies non-linearly across the audio spectrum and empirical evidence [14] suggests that designing a front-end to operate in a similar non-linear manner to that of the human auditory system improves recognition rates [6]. The mel-scale filterbank approach is the most straightforward way to obtain the desired non-linear frequency transformation. The mapping from the linear scale to mel scale is given by the approximation

$$F_{mel} = 2595\log_{10}(1+\frac{F_{Hz}}{700}). \tag{2.5.4}$$

where, F_{mel} is the perceived frequency and F_{Hz} denotes the real frequency [31]. A *mel* is a unit of measure of perceived pitch or frequency of a tone. Figure 2.4 shows the warping of the linear frequency scale by the mel scale.

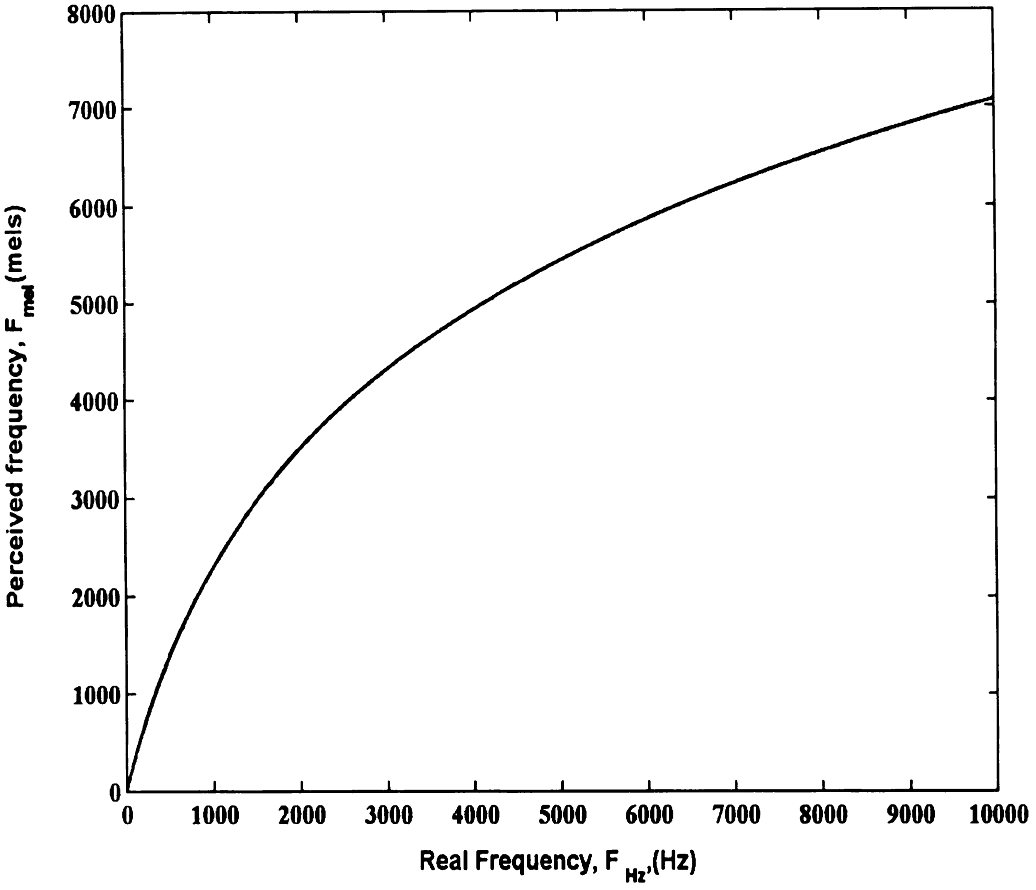


Figure 2.4. The mel scale.

It has been found that the perception of a particular frequency by the auditory system, is influenced by the energy in a critical band of frequencies around that particular frequency [17]. Further, the bandwidth of a critical band varies with frequency, beginning at about $100Hz$ for frequencies below $1kHz$, and then increasing logarithmically above $1kHz$. The *log total energy* in critical bands centered around the mel frequencies are computed by correlating the log magnitude spectrum corresponding to a critical band filter and calculating the weighted sum of the log magnitude spectrum for that particular critical band filter. We use the notation $Y(i)$ to denote the log total energy in the i^{th} critical band with center frequency F_{ic} and lower and upper cutoff frequency F_{il} and F_{iu} respectively, where

$$\begin{aligned} Y(i) &= \sum_{q=0}^{N'/2} \log|S(q;r)|\, H_i(q\frac{2\pi}{N'}) \\ &= \sum_{q=d_{il}}^{d_{iu}} \log|S(q;r)|\, H_i(q\frac{2\pi}{N'}) \end{aligned} \qquad (2.5.5)$$

where, N' is the number of points used to compute the stDFT. The integers i index the center frequencies of the critical band filters, each of which is assumed to be centered on one of the frequencies resolved by the stDFT. $H_i(q\frac{2\pi}{N'})$ is the magnitude spectrum of the i^{th} critical band filter. If we know the sampling frequency F_s of the speech signal, the relation between the cutoff frequencies F_{il} and F_{iu}, and their corresponding sequence indices is given by

$$F_{il} = d_{il}\frac{F_s}{N'} \quad \text{and} \quad F_{iu} = d_{iu}\frac{F_s}{N'} \qquad (2.5.6)$$

The resultant sequence is given by

$$\tilde{Y}(q) = \begin{cases} Y(i), & q = d_{ic} \\ 0 & \text{other } q \in [0, N'-1] \end{cases}, \text{ where } F_{ic} = d_{ic}\frac{F_s}{N'} \qquad (2.5.7)$$

2.5.3 Mel-frequency Cepstrum Coefficients (MFCC)

The final step in the MFCC feature extraction process ids taking an IDFT of the mel-scaled filter-bank coefficients. The MFCC at frame position r is given by

$$c_s(n;r) = \frac{2}{N'} \left[\sum_{\substack{d_{ic} \\ i=1,2,\ldots,N_{cb}}} \tilde{Y}(d_{ic}) \cos(d_{ic} \frac{2\pi n}{N'}) \right] \tag{2.5.8}$$

N_{cb} is the total number of critical band filters used on the Nyquist range, hence there are only N_{cb} terms in the sum of (2.5.8). Here we note that the IDFT reduces to a discrete cosine transform (DCT). This simplifies the stochastic characterization of the features thereby reducing computational costs.

2.5.4 Log Energy, Delta and Acceleration Coefficients

To augment the spectral parameters derived from the MFCC analysis, a *log energy coefficient* is added to the MFCC parameters which is given by

$$c_s(0;m) = \sum_{i=1,\ldots N_{cb}} \tilde{Y}(d_{ic}) \tag{2.5.9}$$

A further improvement in performance can be obtained by adding *differenced* or *delta* cepstrum coefficients to the MFCC parameters thereby accounting for the dynamics of the speech signal. The delta coefficient at frame r is defined as

$$\Delta c_s(n;r) \overset{def}{=} c_s(n;r+\eta Q) - c_s(n;r-\eta Q) \qquad (2.5.10)$$

for all n. Here Q represents the number of samples by which the window is shifted for each frame and η is chosen to smooth the delta cepstrum. The acceleration coefficients (also known as the *delta-delta* coefficients) are obtained by applying the above equation to the delta coefficients.

2.6 Language Modeling

When statistical relationships among utterances are known, a language model (LM) makes it possible to reduce the search space for the given recognition task, or alternatively assign higher probabilities to some utterances than others, thereby reducing recognition errors. Stochastic LMs apply a probabilistic and statistical framework to the language modeling problem. The most widely used stochastic language model in speech recognition tasks is the N-gram model. An N-gram grammar is a representation of an Nth-order Markov LM in which the probability of occurrence of an utterance is conditioned upon the prior occurrence of N-1 other utterances. The utterances can either be whole words or simple phonemes. In this research, we use utterances of vowel phonemes. In the N-gram approach, the language information is formulated as a probability distribution of the different utterances in the vocabulary. In this thesis, the utterances represent the individual phoneme utterances that are used in the training and testing of HMMs. Let us formally define an N-gram stochastic language model. Let $W = w_1, w_2, \ldots\ldots, w_{L_w}$ be a string of known utterances of length L_w in the vocabulary and $P(W)$ be the *a priori* probability of the given sequence W . Then $P(W)$ can be factored as

$$P(W) = P(w_1, w_2, \ldots., w_{L_w}) = \prod_{i=1}^{L_w} P(w_i \mid w_1, \ldots, w_{i-1}) \qquad (2.6.1)$$

However estimating the joint probability above is a computationally intensive task. A practical solution is to use an N-gram language model with N=2 (known as a bigram LM). In a bigram LM the probability of occurrence of a given utterance is conditioned only on the occurrence of the preceding utterance. Bigram models help reduce computation and also provide a simple unified framework to embed both language and phonetic information in a single HMM. In this thesis, we concentrate on bigram language models. (2.6.1) can now be re-written as

$$P(W) = \prod_{i=2}^{L_w} P(w_i \mid w_{i-1}). \qquad (2.6.2)$$

Let the observed string or sequence of utterances be $O = \{o_1, o_2, \ldots, o_{L_o}\}$. We can find the most likely utterance string W^* using the MAP classification rule

$$W^* = \arg\max_{W} \{P(O \mid W)P(W)\}. \qquad (2.6.3)$$

To evaluate (2.6.3), we replace the known word string $W = w_1, w_2, \ldots\ldots, w_{L_w}$ with HMM models representing each of the known utterances in the word string W, i.e., we construct a network of HMMs $\{\lambda_k, \quad 1 \le k \le L_\lambda\}$ representing each utterance in W. Here, we assume that each utterance in the vocabulary has only one HMM associated with it. The probability $P(O \mid W)$ now is equivalent to estimating the probability $P(O \mid \lambda_k)$, the likelihood score of the HMM, which can be evaluated using the forward recursion of the forward-backward algorithm as discussed in Section 2.4.1. The bigram probability $P(W)$, which is nothing but the probability of transition from one phoneme can be obtained from the LM defined in (2.6.2). The detailed algorithm for evaluating the bigram probability scores is described later in Section 3.5.

2.7 Phonemes and Phonetic Transcription

"The basic theoretical unit for describing how speech conveys linguistic information is called a *phoneme*. For American English, there are about 42 phonemes consisting of vowels, semivowels, diphthongs and consonants. Each phoneme is a result of a unique set of *articulatory gestures* (such as the type and location of the sound excitation as well as the position or movement of the vocal tract articulators). Due to many different factors including, for example, accents, gender, and, most importantly, coarticulatory effects, a given phoneme will have a variety of acoustic manifestations in the course of flowing speech. Thus, from an acoustic point of view, the phoneme represents a class of sounds that convey the same meaning. The phonemes of a language, therefore, comprise a minimal theoretic set of units sufficient to convey all the meaning in the language. The process of translating speech into a string of symbols representing the phoneme is called *phonemic transcription* and if it includes diacritical marks indicating allophonic variation, the process is called *phonetic transcription*" [2]. The three widely used phonetic transcriptions are the *International Phonetic Alphabet* (IPA), the *Single Symbol Version*, and the *Upper Case version of the ARPAbet.* In this thesis, we will use the upper case *ARPAbet*, which is used for the phonetic transcriptions in the TIMIT database developed by Texas Instruments and Massachusetts Institute of Technology. The mapping of the IPA symbols and upper case ARPAbet for the vowels in American English are shown in Table 2.1.

Table 2.1. Phonetic transcriptions of vowels.

IPA Symbol	Upper case ARPAbet	Example word
i	IY	beet
I	IH	bit
E	EH	bet
e	EY	bait
æ	AE	bat
ɑ	AA	bott
ɑU	AW	bout
ɑI	AY	bite
Λ	AH	but
ɔ	AO	bought
ɔI	OY	boy
o	OW	boat
U	UH	book
u	UW	boot
ə	AX	about
ɨ	IX	debit

3 Implementation Details

3.1 Hardware and Software Tools

The phoneme recognition system and all the following experiments were carried out on Pentium-III class personal computers running the Windows NT 4.0 operating system. The HMM routines from the Bayes Net Toolbox developed by Kevin Murphy at University of California, Berkeley [15] was used for training and implementing the HMM based speech recognition system. In addition, the VoiceBox toolbox developed by Mikes Brooks, Imperial College, London [16] was employed to extract MFCC parameters from the speech signal. MATLAB-6.1 was used as the development environment as it is an interactive, matrix-oriented programming language with built-in support for data analysis and visualization.

3.2 Speech Databases

3.2.1 TIMIT Speech Corpus

"The TIMIT database is a corpus of read speech developed by Texas Instruments and Massachusetts Institute of Technology. The main purpose for designing his corpus was to provide speech data for acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems. TIMIT contains speech from 630 speakers representing eight major dialect divisions of American English, each speaking 10 phonetically-rich sentences. The TIMIT corpus also includes time-aligned orthographic, phonetic, and word transcriptions, as well as speech waveform data for each spoken sentence. The text material in the TIMIT prompts, consists of two dialect "shibboleth" sentences, 450 phonetically-compact sentences, and 1890 phonetically-diverse sentences. The dialect sentences (designated SA type in the database) were meant to expose dialectal variants of the speakers, and were read by all 630 speakers. The phonetically compact sentences (designated SX type in the database) were meant to be comprehensive as well as compact. The phonetically diverse sentences (designated SI type in the database) were selected to add diversity in sentence types and phonetic contexts. The corpus is also subdivided into training set (70-80% of the corpus) and test set (20-30% of the corpus)" [1]. The speakers are both male and female. The speech data were sampled at $16kHz$ and the digitized wavfile was stored in the National Institute for Standards and Technology (NIST) SPeech HEader REsource (SPHERE) format using 16 bits/sample.

3.2.2 The Dysarthric Speech Corpus

This corpus consists of non-labeled speech data collected by personnel in the Communication Center of Excellence, at Madonna Rehabilitation Hospital, Nebraska, following an Institutional Review Board (IRB)-approved protocol for the protection of human subjects. The principal investigator for the clinical study is Professor David Beukelman of the University of Nebraska, Department of Special Education, and is also a Researcher associated with the Madonna Rehabilitation Hospital. The corpus is comprised of isolated utterances of nine vowel sounds, four semivowel sounds and two nasal sounds. Each of the utterances was repeated at least 10 times to provide sufficient number of speech samples for testing and evaluation purposes. Four speakers with varying amounts of dysarthria provided the isolated sound utterances. The description of the four speakers is given below:

- Speaker 1 is a 38-year-old male with a diagnosis of Traumatic Brain Injury (TBI). His intelligibility is severely/profoundly impaired. Speech characteristics include slow rate, inability to produce consonant sounds other than nasals, vowel distortions, but some control over pitch and intonation.

- Speaker 2 is a 26-year-old female with a diagnosis dysarthria secondary to athetoid cerebral palsy. Her intelligibility is severely impaired. Speech characteristics include imprecise consonants, slow rate, distorted vowels, and some control over prosody.

- Speaker 3 is a 39-year-old male with a diagnosis of TBI. His intelligibility is moderately impaired. His speech characteristics include impaired control over respiration, strained/strangled voice quality, imprecise consonant production and decreased word boundaries.

- Speaker 4 is a 49-year-old female with a diagnosis of dysarthria secondary to mixed cerebral palsy. Her intelligibility is severely impaired. Speech characteristics include imprecise consonants, repetition of phonemes, irregular articulatory breakdown, and distorted vowels

The waveforms were digitally recorded at a 44.1kHz sampling rate, stereo and all the utterances were stored in a single wave file (.wav format) [18] for each speaker. Table 3.1 tabulates the isolated utterances that were used to build the dysarthric speech corpus.

Table 3.1. Isolated utterances in the dysarthric speech database.

Sounds	**Uppercase ARPAbet Symbols**	**Example**
Vowels	OW	open
	AA	ma
	IY	eat
	AH	up
	AY	eye
	AE	cat
	AO	awful
	UW	oops
	EY	ate
Semivowel	L	fall
	R	earn
	Y	young
	W	way
Nasals	M	hum
	N	no

3.3 Creating the Observation Strings

As discussed in Section 2.5, the speech signal must be converted to a suitable feature space. This thesis concentrates on developing vowel phoneme-level HMMs and evaluating their performance on the dysarthric speech corpus. For this purpose, seven vowel sounds (UW, OW, AY, AE, AO, IY and EY) were chosen for training and evaluation purposes. The HMM models for the two remaining vowel sounds (AA and AH) could not be properly trained as there was considerable acoustic variation associated with these phonemes within the TIMIT database. Hence, the phonemes AA and AH were not incorporated into the phoneme classifier. All results and experiments have been carried out using these seven vowel sounds.

3.3.1 Phoneme Extraction from the TIMIT Speech Corpus

The speech utterances in the TIMIT database are complete sentences and each wave file is associated with a phonetic transcription of the sentence spoken. This transcription was used to extract the seven chosen vowel phonemes from all the speakers across seven dialects, for the training set. Before extracting the phonemes the NIST SPHERE wave files were converted to Windows PCM wave files using the NIST-provided software *sphconvert.exe* [19]. Only the SI and SX type sentences were considered during the phoneme extraction process, as the SA type sentences tend to introduce a bias in the recognition process [20]. The resultant extracted phonemes were stored in a binary format.

3.3.2 Phoneme Extraction from the Dysarthric Database

The dysarthric speech database obtained from the University of Nebraska contained all the utterances for a given speaker within a single large wave file. The wave file for each of the speakers was broken into smaller segments containing only a single phoneme utterance and were stored as individual wave files using the commercial software package “Cool Edit” [30]. Before segmenting, the speech files were down sampled to 16kHz and converted from stereo (2 channels) to mono (1 channel) using Cool Edit.

3.3.3 Features Comprising the Observation Strings

The feature space is a 39-dimensional feature vector comprising 12 mel-cepstrum coefficients, a log energy coefficient, 13 delta cepstrum coefficients and 13 delta-delta cepstrum coefficients. The 0^{th} order coefficient of the MFCC is not included as it is closely related to the log energy measure. The features are computed over a speech signal frame of 160 points after a Hamming window has been applied. The window is advanced by 64 points for each frame. For a $16kHz$ speech signal, this implies that a $4ms$ window is applied for every $10ms$ of speech. The 39-dimensional MFCC parameters obtained both for normal as well as the dysarthric speech, constitutes the observation string that is applied as input to an HMM. Figure 3.1 shows the MFCC for the vowel phoneme ‘AE’. The feature vector consists of 12 mel-cepstrum coefficients, 1 log energy coefficient, 13 delta coefficients and 13 delta-delta coefficients. The MFCC features have been extracted for the entire speech waveform for all the frame positions.

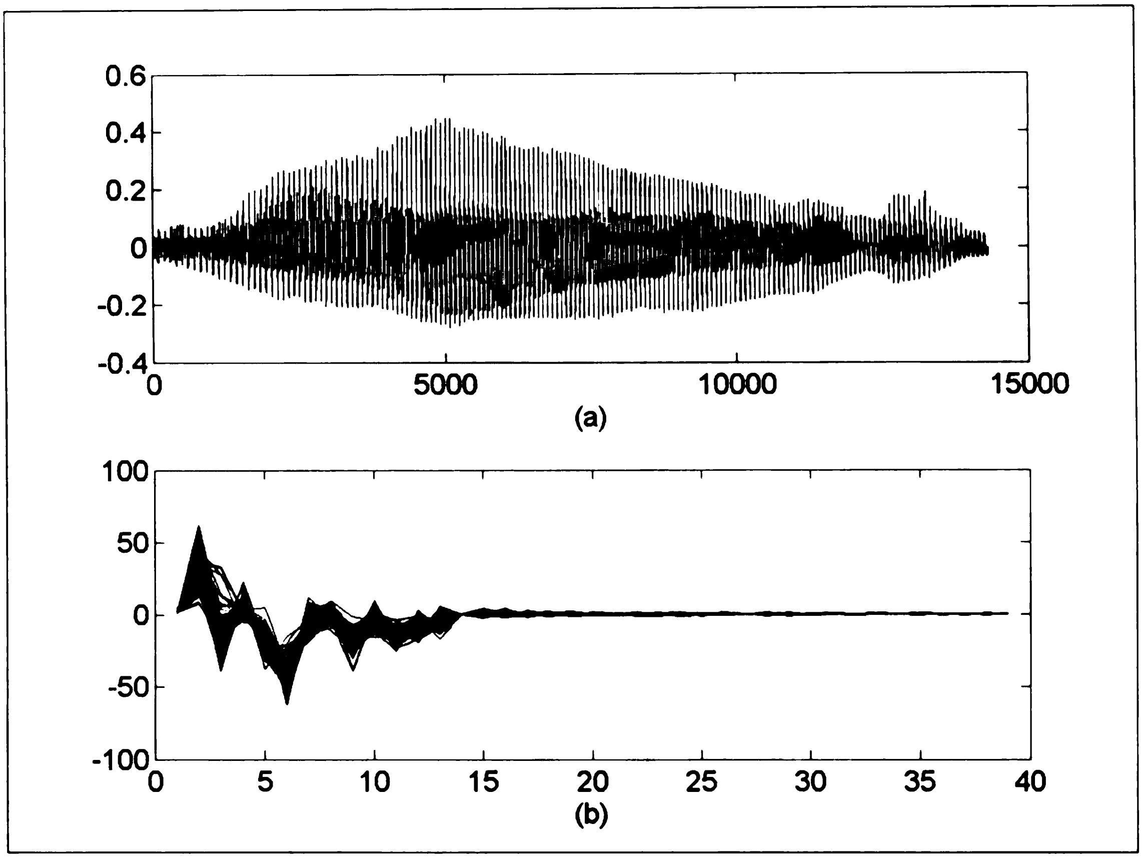


Figure 3.1. Example of MFCC feature extraction: (a) The speech waveform for the phoneme utterance 'AE'. (b) The 39-dimensional MFCC parameters.

3.4 Implementation of the Phoneme Recognizer

The phoneme recognizer consists of seven vowel-based HMMs trained on normal speech from the TIMIT database. A given test speech utterance is passed through the seven vowel HMMs and the likelihood scores are computed. The vowel associated with the model giving the maximum score is chosen as the recognized vowel. Thus each given speech utterance is classified as one of the seven different vowels. Figure 3.2 shows the structure of the phoneme classifier that was implemented.

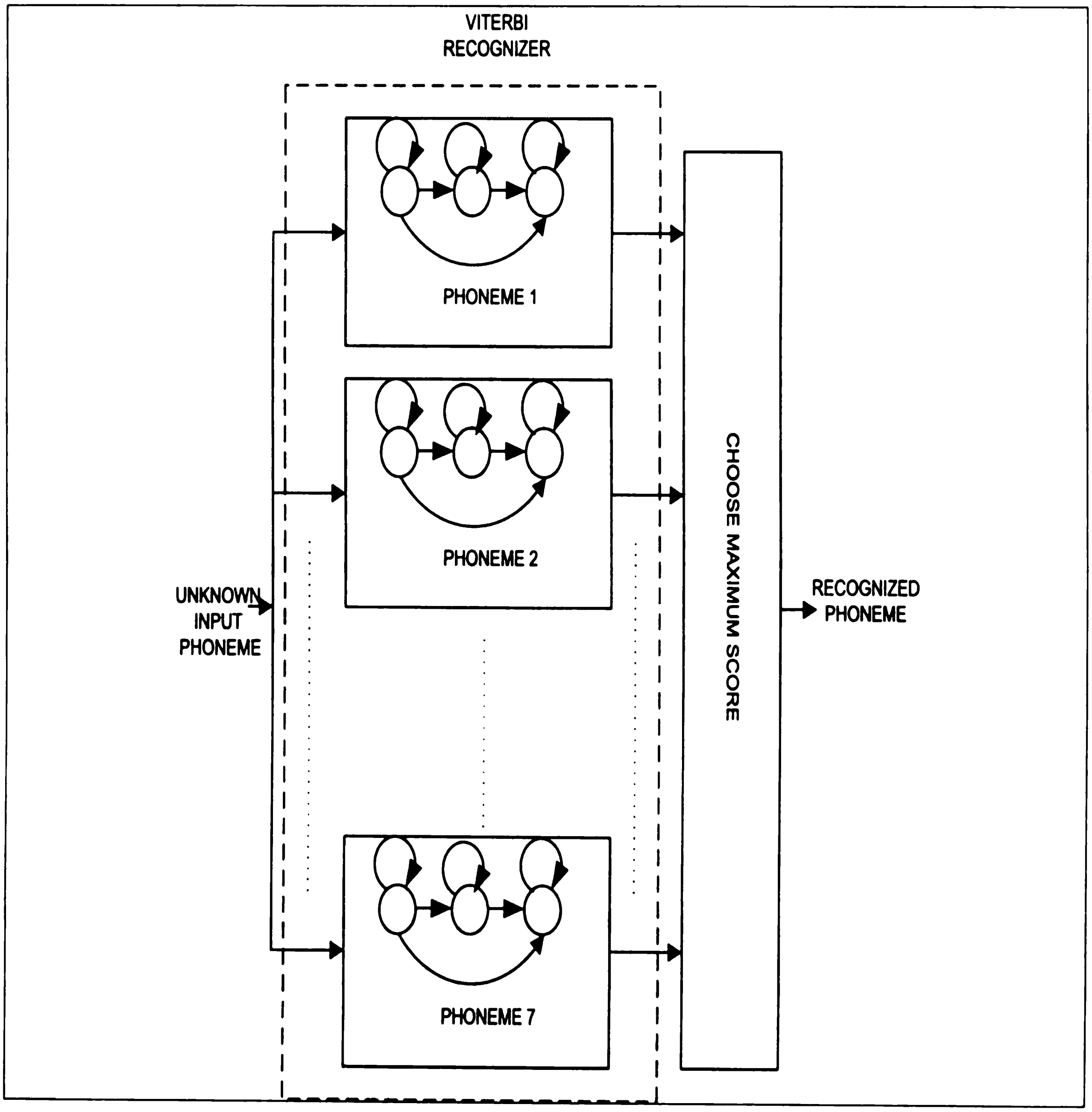


Figure 3.2. The phoneme classifier.

3.4.1 HMM Topology

Phoneme utterances are typically represented by a three-state Bakis structure [12]. The first state and the last state nominally represent the transition into and out of the phoneme, respectively, and the middle state represents the steady-state portion of the utterance. The state transition probabilities are governed by the following equations

$$a_{ij} = 0, \qquad j < i. \tag{3.4.1}$$

$$a_{Ni} = \begin{cases} 0, & i < N \\ 1, & i = N. \end{cases} \tag{3.4.2}$$

i.e., no transitions are allowed to state whose indices are lower than that of the current state. Furthermore, the initial state probabilities are

$$\pi_i = \begin{cases} 1, & i = 1 \\ 0, & i \neq 1. \end{cases} \tag{3.4.3}$$

since the state sequences must begin in state 1 and end in state N (in this case N=3). The state transition probability matrix for the phoneme HMM has the following upper diagonal form

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4.4}$$

Since we are using CHMMs, the output observation pdf is modeled by 10-20 Gaussian mixture models(GMM) depending upon the best fit for a HMM.

3.4.2 HMM Training and Testing

The Baum-Welch re-estimation procedure is used for training the HMMs as discussed in the previous chapter. The algorithm is run iteratively for each phoneme HMM for a loop count of 15 and the convergence threshold value is fixed at 7×10^{-4}. Each of the models is trained over the entire training set over all the dialects in the TIMIT training set. The speakers in the training set have been limited to male speakers as the dysarthric speech corpus consists only of male speakers. The Baum-Welch algorithm requires proper initialization in order to achieve correct training of the phoneme HMMs. The HMM state transition probabilities are initialized as given in (3.4.4) and the initial state probabilities are initialized as given in (3.4.3). The initial self-loop probabilities are assigned a value of 0.8. The initial transition matrix has the following values

$$A=\begin{bmatrix}0.8 & 0.1 & 0.1\\ 0 & 0.8 & .2\\ 0 & 0 & 1\end{bmatrix}$$

and the initial state probability matrix takes the form

$$\pi_i=\begin{cases}1, & i=1\\ 0, & i=2,3.\end{cases}$$

The means and covariance matrices are initialized to the global mean and global variance of the training data respectively.

The testing is done by using the training data as an input to the phoneme recognizer. The accuracy of a specific phoneme model (say belonging to class k) is measured as follows

$$\%\text{Accuracy of phoneme } k = \frac{\#\,\text{correctly recognized occurences of phoneme } k}{\#\text{total occurences of phoneme } k} \quad (3.4.5)$$

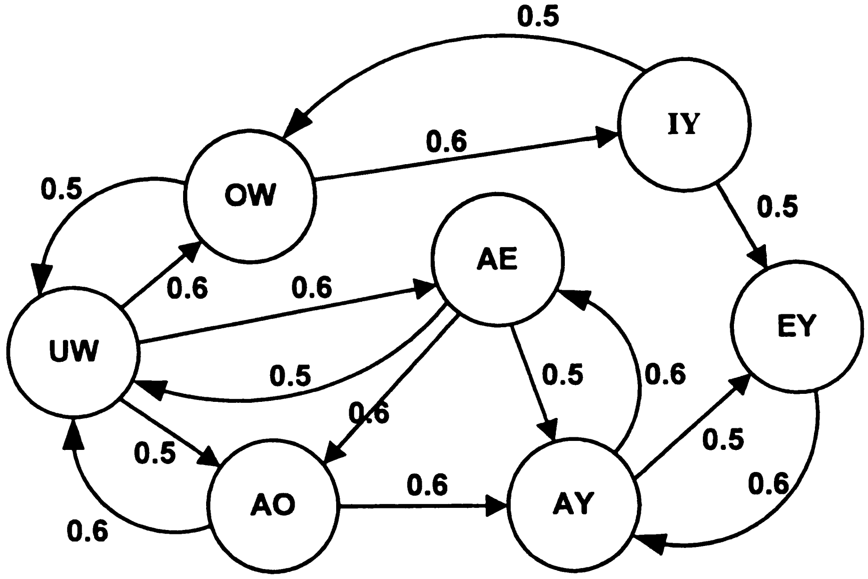
A correctly trained HMM will have a recognition accuracy near 100%, when the data upon which it is trained is used, as input. In practice, however, there is a small error in recognition. The recognition error is due to outliers in the training set that are not properly modeled during the training procedure. Such imperfections are acceptable given the variability in speech and the variation in dialects of the different speakers in the TIMIT training set. Table 3.2 shows the accuracy of the trained HMMs, when tested with their respective TIMIT training data.

Table 3.2. Accuracy of the vowel phoneme HMM on TIMIT training data.

Vowel HMMs	**Training accuracy**
UW	94.23%
OW	96.65%
IY	98.44%
AE	96.02%
AO	95.54%
EY	91.01%
AY	97.12%

3.5 Bigram Language Model Implementation

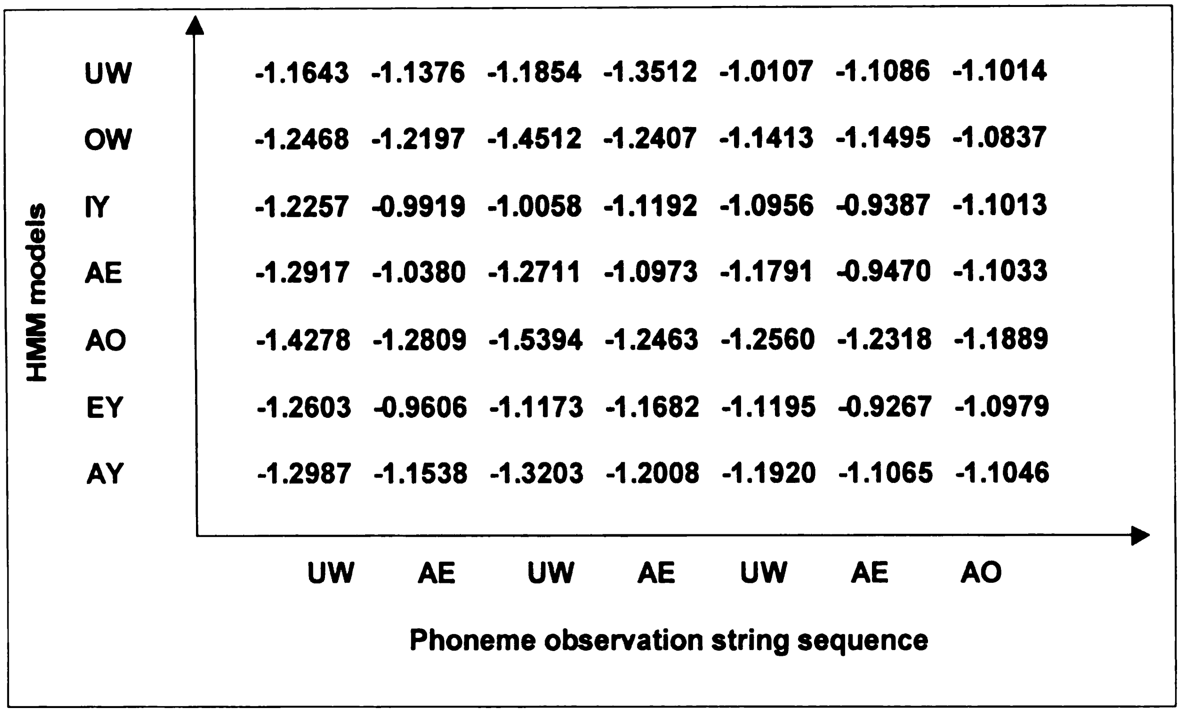
The phoneme recognizer discussed in Section 3.4 is the baseline acoustic recognizer. This baseline recognizer assumes that all the vowels in the test set occur with equal probability. From Section.2.6 we know that the performance of the recognizer can be improved if we constrain the search path in the vocabulary by assigning different probabilities to the different utterances. In this thesis, the dysarthric speech utterances are phoneme utterances of the different vowel sounds. So, in order to test the hypothesis that bigram models do improve recognition as compared to simple acoustic recognition (phoneme classifier without bigram LM) we formulate a phoneme-to-phoneme bigram probability matrix, which is our LM. In the context of this research, if $W = \{w_1, \ldots\ldots, w_{L_w}\}$ represents the string of phoneme utterances of dysarthric speech, the bigram probability $P[w_i \mid w_{i-1}]$ is defined as the probability given utterance w_{i-1}, utterance w_i follows it. Thus the LM gives the phoneme-to-phoneme transition probabilities. Figure 3.3 shows a bigram LM using seven vowel phonemes. The bigram LM, gives us the phoneme-to-phoneme transition probability and can be conveniently represented in the matrix form as shown in Figure 3.3. The bigram LM in this research, is a contrived one as the database consists of only isolated vowel sounds. The purpose of using such a contrived LM is to demonstrate the increase in recognition accuracies of the vowel phonemes as compared to a recognizer without any LM.



		Vowel Phonemes						
		UW	OW	IY	AE	AO	EY	AY
Vowel Phonemes	UW	0	.6	0	.6	.5	0	0
	OW	.5	0	.6	0	0	0	0
	IY	0	.5	0	0	0	.5	0
	AE	.5	0	0	0	.6	0	.5
	AO	.6	0	0	0	0	0	.6
	EY	0	0	0	0	0	0	.6
	AY	0	0	0	.6	0	.5	0

Figure 3.3. Example of a bigram LM using seven vowel phonemes.

Once we have deduced the bigram probability matrix, from the training data, we generate a finite length sequence of phoneme utterances based on this bigram probability distribution, which represents the observation string. This sequence of phonemes is then passed through the acoustic phoneme recognizer and the resultant likelihood scores of each of the utterances are accumulated to construct a Viterbi search grid. Figure 3.4 shows the construction of the search grid for the LM defined in Figure 3.3. The ordinate in Figure 3.4 represents the HMM models used in the recognizer and the abscissa represents the observation string sequence generated from the bigram LM. The numbers in the figure represent the actual likelihood scores (acoustic scores) obtained at the output of each vowel HMM after passing each utterance in the observation sequence through the recognizer.



HMM models	UW	AE	UW	AE	UW	AE	AO
UW	-1.1643	-1.1376	-1.1854	-1.3512	-1.0107	-1.1086	-1.1014
OW	-1.2468	-1.2197	-1.4512	-1.2407	-1.1413	-1.1495	-1.0837
IY	-1.2257	-0.9919	-1.0058	-1.1192	-1.0956	-0.9387	-1.1013
AE	-1.2917	-1.0380	-1.2711	-1.0973	-1.1791	-0.9470	-1.1033
AO	-1.4278	-1.2809	-1.5394	-1.2463	-1.2560	-1.2318	-1.1889
EY	-1.2603	-0.9606	-1.1173	-1.1682	-1.1195	-0.9267	-1.0979
AY	-1.2987	-1.1538	-1.3203	-1.2008	-1.1920	-1.1065	-1.1046

Phoneme observation string sequence

Figure 3.4. LM-based Viterbi search grid.

Then a Viterbi search is applied to this search grid, which combines both the acoustic scores and the logarithm of the bigram language model probabilities to determine the best path, through the search grid. The Viterbi search algorithm used in the LM is a modification to the Viterbi algorithm given in Figure 2.3. In the case of the LM-based Viterbi search the observations are a string of phoneme utterances instead of feature vectors, and instead of searching through each state of a single HMM, in this case, the search is among different HMM models representing each vowel phoneme. The best path thus obtained represents the recognized string, based on the given bigram information. The algorithm used for the LM-based Viterbi search is shown in Figure 3.5.

Figure 3.5. LM-based Viterbi search algorithm.

Let L_λ be the number of phoneme HMM classes in the phoneme recognizer and L_o be the length of the string of utterances $W' = \{w_1', w_2', \ldots, w_{L_o}'\}$ generated using the bigram LM. The likelihood score associated with utterance i given HMM λ_j is given by $\phi_i(j)$. The recognized utterance string is given by $W^* = \{w_1^*, w_2^*, \ldots, w_{L_o}^*\}$.

Initialization:

$$\text{For,}\quad 1 \le j \le L_\lambda$$
$$\phi_1(j) = \log P[w_1' \mid \lambda_j]$$
$$\text{For,}\quad 1 \le j \le L_\lambda \text{ and } 1 \le i \le L_o$$

Recursion:

$$\textit{For,}\quad 1 \le i \le N_o,\ 1 \le j \le L_\lambda \ \textit{and}\ 1 \le k \le L_\lambda$$
$$\phi_i(j) = \max\left\{\phi_{i-1}(j) + \log P[w_i' \mid \lambda_j] + \log P[w_i \mid w_k]\right\}$$
$$\psi_i(j) = \arg\max_k \left\{\phi_{i-1}(j) + \log P[w_i' \mid \lambda_j] + \log P[w_i \mid w_k]\right\}$$

Figure 3.5. LM-based Viterbi search algorithm (cont'd).

Termination:

$$w^*_{L_o} = \arg\max_k \left\{ \phi_{L_o}(j) \right\}, \qquad 1 \le j \le L_\lambda$$

Backtracking:

$$w^*_n = \psi_{n+1}(n+1), \qquad \text{for } n = L_o - 1 \text{ to } 1$$

Recognized phoneme sequence

$W^* = \{ w^*_1, w^*_2, \ldots, w^*_{L_o} \}$

4 Experimental Evaluation

4.1 Overview

The main goal of this research is to evaluate the feasibility of a phoneme-based vowel recognizer for dysarthric speech. The question is, are there certain vowel sounds that can reliably be distinguished from one another, for a given dysarthric speaker? In order to use any of the vowel sound as control triggers for an AAC device, it is necessary that they be reliably discriminated. This chapter evaluates the feasibility of such a phoneme-based vowel recognizer.

4.2 Experiments with Normal Speech

Recognition experiments were performed with normal speech to test the trained models and to establish a baseline result. The test set for the recognition experiments was obtained from the TIMIT database. The recognition accuracy of a particular phoneme, as defined in Section 3.4.2 is given by

$$A_k = \frac{\#\ \text{correctly recognized occurences of phoneme}\ k}{\#\ \text{total occurences of phoneme}\ k}$$

The overall recognition accuracy of the recognizer is calculated by taking the weighted sum of the different phoneme recognition accuracies. The weighting coefficient here is the *a priori* probability of occurrence of the different phonemes. If A denotes the overall accuracy of the recognizer then

$$A = \sum_{k=1}^{7} P[\text{phoneme}\, k]A_k. \tag{4.1.1}$$

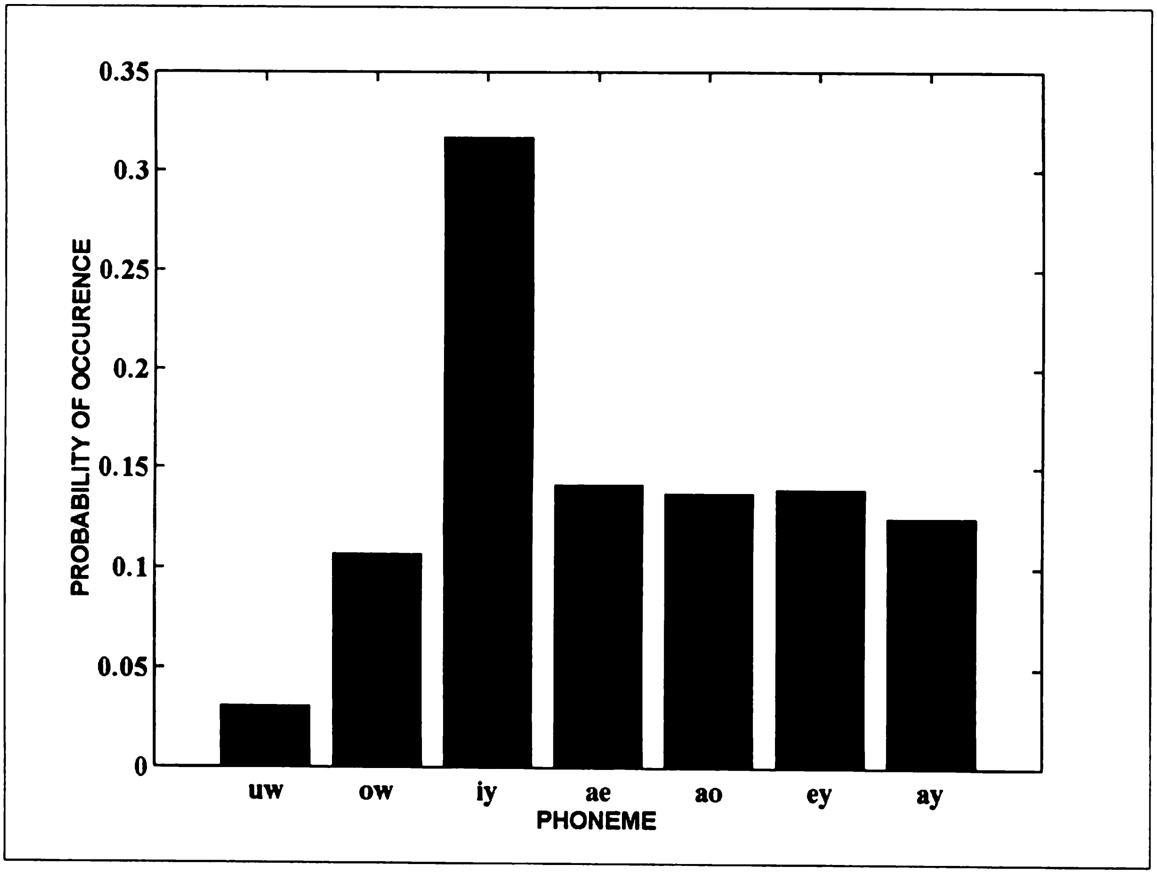


Figure 4.1. The *a priori* distribution of the seven vowels in the TIMIT test set.

The results of the recognition experiment on the TIMIT test set are given in Table 4.1.

Table 4.1. Recognition results with normal speech.

k	Vowel	P[phoneme k]	A_k	Total Number of test utterances in TIMIT	%Accuracy
1	UW	0.0308	73/106	106	68.86%
2	OW	0.1074	306/369	369	82.92%
3	IY	0.3169	961/1089	1089	88.24%
4	AE	0.1420	419/488	488	85.86%
5	AO	0.1380	385/474	474	81.22%
6	EY	0.1397	410/480	480	85.41%
7	AY	0.1251	364/430	430	84.65%

The recognition accuracy of the phoneme-based vowel recognizer can now be calculated from the data in Table 4.1 and (4.1.1). The overall accuracy of the recognizer was calculated to be 84.92%. This is quite consistent with the phoneme recognition accuracies normally obtained using TIMIT, which are typically in the range of 70-80% for phonemes. For details, the reader is referred to [32].

4.3 Experiments with Dysarthric Speech

The dysarthric speech database consists of multiple utterances of the seven vowel phonemes (UW, OW, IY, AE, AO, EY, AY) spoken by four different speakers with varying amounts of dysarthria. Reliability of a vowel phoneme here refers to the degree to which the phoneme classifier does not confuse a vowel phoneme with other vowel phonemes. Table 4.2 shows the number of vowel utterances for each of the four dysarthric speakers. Each of the vowels in the dysarthric speech database has been repeated atleast 10 times by each of the speakers. The different number of uterances of each vowel is due to the improper prompts for each of the dysarthric speech wave files, which made it difficult to identify the dysarthric utterance.

Table 4.2. Distribution of utterances in the dysarthric speech database.

Dysarthric Speakers	Number of utterances of vowels						
	UW	OW	IY	AE	AO	EY	AY
Speaker 1	10	12	8	7	7	9	16
Speaker 2	10	12	10	12	10	10	10
Speaker 3	10	10	10	10	10	10	10
Speaker 4	10	11	10	10	10	10	10

4.3.1 Classification Experiment

The dysarthric speech utterances for each of the speakers were passed through the phoneme classifier to produce a confusion matrix. The confusion matrix is a grid, which, for each vowel, indicates the number of times that vowel was classified as each of the candidate vowels in the set.

Table 4.3. Confusion matrix for Speaker 4 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel utterances	UW	0	10	0	0	0	0	0
	OW	1	[illegible]	0	0	0	0	0
	IY	0	0	[illegible]	0	0	3	0
	AE	0	0	0	10	0	0	0
	AO	0	2	0	4	0	0	4
	EY	0	0	2	4	0	4	0
	AY	0	0	0	5	0	0	5

Table 4.3 shows the confusion matrix for Speaker 4. The columns represent the different phoneme vowels used in the recognizer and the rows indicate the different vowel utterances that were passed through the phoneme classifier. The number in each cell represents the number of test utterances of a given vowel phoneme (from the dysarthric speech database) that were classified as belonging to each vowel HMM. This method of representation of the classification results helps identify the vowel sounds that are least likely to be confused with other vowel sounds.

In Table 4.3, it is observed that the three vowel phonemes OW, IY and AE are never confused with each other and hence are the most reliable sounds produced by Speaker 4. Thus for Speaker 4 we can postulate at least three reliable vowel phonemes.

Similarly, confusion matrices for Speaker 1, Speaker 2 and Speaker 3 represent results of running similar classification experiments.

Table 4.4. Confusion matrix for Speaker 1 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel utterances	UW	9	0	1	0	0	0	0
	OW	8	1	1	0	0	2	0
	IY	0	0	8	0	0	0	0
	AE	0	0	1	1	0	5	0
	AO	0	2	0	0	0	5	0
	EY	0	0	9	0	0	0	0
	AY	0	0	8	0	0	8	0

Table 4.5. Confusion matrix for Speaker 2 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	6	3	1	0	0	0	0
	OW	2	8	0	0	1	0	1
	IY	2	0	8	0	0	0	0
	AE	2	2	0	4	0	0	4
	AO	0	0	0	0	3	0	7
	EY	0	0	4	4	0	2	0
	AY	0	2	0	0	4	0	4

Table 4.6. Confusion matrix for Speaker 3 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	0	9	0	1	0	0	0
	OW	0	4	0	0	6	0	0
	IY	0	0	9	0	0	1	0
	AE	0	0	0	10	0	0	0
	AO	0	0	0	4	2	0	4
	EY	0	0	0	2	0	8	0
	AY	0	0	0	9	0	0	1

For Speaker 1, we observe from Table 4.4 that the phonemes UW and IY have high recognition accuracies. UW is confused with IY only once and this suggests that with a little intervention (some articulatory training by clinicians), Speaker 1 can be trained to reliably produce the phoneme UW. Similarly, for Speaker 2 we observe that the phonemes OW and IY are reliable choices, and for Speaker 3, OW, AE and IY are reliable choices.

4.3.2 Conclusion

Using a phoneme classifier trained on normal speech, we were able to recognize at least two reliable vowel sounds for each of the dysarthric speakers. This suggests that it is feasible to build a speech recognition system capable of recognizing vowel sounds with the minimum amount of confusability with other vowels.

4.4 Experiments with Language Modeling

The purpose of the LM is to restrict the search space of the recognition task and thereby help in reducing recognition error rates. In this research, we use bigram LMs to investigate the effect of language modeling on vowel phoneme recognition rates. The goal is to ascertain whether introducing LM in the recognition task, increases the accuracy with which vowels are recognized. Two cases were considered for the bigram LM. First, a speaker-independent bigram LM was computed for all speakers to determine whether it is possible to derive a single LM that can increase recognition accuracies across all dysarthric speakers. In the second experiment, a restrictive speaker-dependent bigram LM was used to test for increase in vowel recognition accuracies. Speaker-independent bigram LM in this context is a single LM that represents the entire dysarthric population used in this research. Similarly, a speaker-dependent bigram LM is one that is specific to each speaker. In order to quantify the bigram LM recognition task results, the LM recognition results were compared with the baseline acoustic recognition results using vowel phoneme error rate. In this research, the vocabulary consists of isolated vowel phoneme utterances, so the error rate is defined as

$$\text{Error rate } E = \frac{\text{\# misrecognitions in the utterance string}}{\text{length of the utterance string}} \tag{4.1.2}$$

The utterance string here refers to a hypothetically observed sequence of phoneme utterances that we wish to recognize.

4.4.1 Speaker-Independent Bigram Language Model

In this experiment, we use a single LM for all four dysarthric speakers. The transition probabilities associated with the bigram LM used in this experiment is shown in Table 4.7. These probabilities are shown in the form of a grid in Table 4.1, where the numbers in each cell denotes the probability of the vowel HMM representing the column (to which the cell belongs) following the vowel HMM representing the row (to which the cell belongs). The details of this representation of the bigram LM are explained in Section 3.5.

Table 4.7. Speaker-independent Bigram LM.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel HMMs	UW	0	0.56	0	0.66	0.45	0	0
	OW	0.45	0	0.56	0	0	0.55	0
	IY	0	0.75	0	0	0.65	0	0
	AE	0.75	0	0	0	0.6	0	0.65
	AO	0.6	0	0.5	0	0	0.43	0.6
	EY	0	0.45	0	0.78	0	0	0.6
	AY	0	0.5	0	0.6	0	0.5	0

The bigram LM shown in Table 4.7 is a contrived one as we have only phoneme utterances in the dysarthric database. The results of the experiments in Section 4.2 were first used to ascertain which vowels could be considered as reliable phonemes. The phoneme-to-phoneme transition probabilities were then assigned such that a poorly recognized vowel phoneme is always followed by a reliably recognized vowel phoneme. In addition, the probability of a vowel phoneme being immediately following itself was made zero. The probability distribution of the vowel HMMs from the bigram LM in Table 4.7 was used to generate finite strings of 30 vowel phoneme utterances for the recognition task. The vowel phonemes used in the LM task are randomly generated, which implies that the number of utterances used in the acoustic recognition task in case of the LM is not necessarily the same as those used in the recognition task of Section 4.2. Hence, the classification results shown in Section 4.2 are not always the same as those shown in the bigram LM recognition task. The results from the acoustic classification task and the bigram LM recognition task are tabulated below.

Table 4.8. Confusion matrix with speaker-independent LM for Speaker 1.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
	UW	6	0	1	0	0	1	0
Vowel Utterances	OW	0	6	0	5	0	0	0
	IY	1	0	4	0	0	2	0
	AE	0	0	0	3	0	0	0
	AO	0	1	0	0	0	0	0
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.9. Confusion matrix without speaker-independent LM for Speaker 1.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	7	0	1	0	0	0	0
	OW	7	1	1	0	0	2	0
	IY	0	0	7	0	0	0	0
	AE	0	0	0	1	0	2	0
	AO	0	1	0	0	0	0	0
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.10. Confusion matrix with speaker-independent LM for Speaker 2.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	9	0	0	0	0	1	0
	OW	0	0	0	7	0	0	0
	IY	5	0	0	0	0	0	0
	AE	0	0	0	5	0	0	0
	AO	0	0	0	2	1	0	0
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.11. Confusion matrix without speaker-independent LM for speaker 2.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	6	3	1	0	0	0	0
	OW	1	4	0	0	1	0	1
	IY	0	0	5	0	0	0	0
	AE	1	1	0	1	0	0	2
	AO	0	0	0	0	0	0	3
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.12. Confusion matrix with speaker-independent LM for Speaker 3.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	6	0	1	0	0	3	0
	OW	0	7	0	1	0	0	0
	IY	0	0	5	0	0	0	0
	AE	0	0	0	6	0	0	0
	AO	0	0	0	0	0	0	1
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.13. Confusion matrix without speaker-independent LM for Speaker 3.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	0	9	0	1	0	0	0
	OW	0	2	0	0	6	0	0
	IY	0	0	5	0	0	0	0
	AE	0	0	0	6	0	0	0
	AO	0	0	0	1	0	0	0
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.14. Confusion matrix with speaker-independent LM for Speaker 4.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	6	0	1	0	0	3	0
	OW	0	7	0	1	0	0	0
	IY	0	0	5	0	0	0	0
	AE	0	0	0	6	0	0	0
	AO	0	0	0	0	0	0	1
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.15. Confusion matrix without speaker-independent LM for Speaker 4.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	0	9	0	1	0	0	0
	OW	0	2	0	0	6	0	0
	IY	0	0	5	0	0	0	0
	AE	0	0	0	6	0	0	0
	AO	0	0	0	1	0	0	0
	EY	0	0	0	0	0	0	0
	AY	0	0	0	0	0	0	0

Table 4.16. Error rates for the recognition task with the speaker-independent LM.

Dysarthric Speakers	Error rate for phoneme recognition with LM	Error rate for phoneme recognition without LM
Speaker1	11/30	14/30
Speaker 2	15/30	14/30
Speaker 3	6/30	17/30
Speaker 4	14/30	13/30

The confusion matrices for the recognition task shown in Table 4.8 and Table 4.15 give the distribution of the classified vowel utterances for each speaker, for both baseline acoustic recognition and recognition with the bigram LM. As a side note, the baseline recognition results in Table 4.9, Table 4.11, Table 4.13 and Table 4.15 consists of vowel utterances randomly chosen from the database depending upon the sequence of phonemes being generated using the speaker-independent bigram LM. This implies that the baseline classification results from the tables mentioned before may be different from those shown in Table 4.3 to Table 4.6. The speaker-independent LM does improve recognition rates for Speaker 1 and Speaker 3 as observed from the error rates in Table 4.16. However, the baseline acoustic recognition rates outperform those of the recognizer with the bigram LM for Speaker 2 and Speaker 4. This suggests that although we can improve recognition rates for dysarthric speech with a bigram LM, it is not possible to deduce a speaker-independent bigram LM to represent the entire dysarthric population.

4.4.2 Speaker-Dependent Bigram Language Model

In this experiment, a speaker-dependent bigram LM was computed and the recognition rates for both the baseline acoustic recognition task and the recognition with speaker – dependent bigram LM were compared. In this case, too, the bigram LM is a contrived one. The phoneme-to-phoneme transition probabilities are defined for each speaker such that there is a high probability transition from a poorly recognized vowel phoneme (for that speaker) to a reliably recognized vowel phoneme (for that speaker). In addition, the LM used is very restrictive so as not to allow too many transitions from a given phoneme. The probability of a vowel phoneme following itself is zero. Using these rules we can generate numerous such LMs for the recognition task. The LMs shown in the Tables below are the ones that perform consistently better than the baseline recognition task more than 90% of the time. The results of the language modeling experiments for each speaker are presented in the tables below.

Table 4.17. Bigram LM for Speaker 1.

		Vowel Phonemes						
		UW	OW	IY	AE	AO	EY	AY
Vowel Phonemes	UW	0	.6	0	.6	.5	0	0
	OW	.5	0	.6	0	0	0	0
	IY	0	.5	0	0	0	.5	0
	AE	.5	0	0	0	.6	0	.5
	AO	.6	0	0	0	0	0	.6
	EY	0	0	0	0	0	0	.6
	AY	0	0	0	.6	0	.5	0

Table 4.18. Confusion matrix with speaker-dependent LM for Speaker1 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	▨	0	1	1	1	0	0
	OW	0	▨	0	0	1	0	0
	IY	0	0	6	0	0	0	0
	AE	0	0	0	▨	0	0	1
	AO	0	0	0	0	**2**	0	0
	EY	0	0	0	0	0	▨	0
	AY	0	0	0	0	0	0	▨

Table 4.19. Confusion matrix without speaker-dependent LM for Speaker 1(shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	■	0	1	0	0	0	0
	OW	6	■	1	0	0	1	0
	IY	0	0	6	0	0	0	0
	AE	0	0	1	■	0	3	0
	AO	0	2	0	0	**0**	0	0
	EY	0	0	7	0	0	**0**	0
	AY	0	0	4	0	0	4	**0**

Table 4.20. Error rate for Speaker 1.

	Error Rate
with bigram LM	5/45
without bigram LM	30/45

We observe from Table 4.18 and Table 4.19 that the bigram model improves recognition rates for the vowels OW, AE, AO, EY and AY. Furthermore, if we observeTable 4.19, the phoneme pairs UW and AE, and OW and AE are never confused with each other for the baseline acoustic recognizer. From, Table 4.18 we also observe that the phoneme set UW, OW, EY and AY are never confused with each other. Thus, there is an increase in the number of reliable vowel phonemes recognized for Speaker 1, as a result of the bigram LM.

Table 4.21. Bigram speaker-dependent LM for Speaker 2.

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel HMMs	UW	0	0	.5	0	0	.5	0
	OW	0	0	0	0	0	0	.5
	IY	.5	0	0	0	.5	0	0
	AE	0	0	0	0	.5	0	0
	AO	0	0	.5	.5	0	0	0
	EY	.5	.5	0	0	0	0	.5
	AY	0	.5	0	0	0	.5	0

Table 4.22. Confusion matrix with speaker-dependent LM for Speaker 2 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	■	0	0	0	0	0	0
	OW	0	■	0	0	0	0	0
	IY	0	0	■	0	0	0	0
	AE	0	0	0	■	0	0	0
	AO	0	0	0	0	■	0	0
	EY	0	0	1	0	0	**6**	0
	AY	0	0	0	0	0	0	■

Table 4.23. Confusion matrix without speaker-dependent LM for Speaker 2 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	**4**	3	1	0	0	0	0
	OW	2	▨	0	0	1	0	1
	IY	1	0	▨	0	0	0	0
	AE	0	0	0	**0**	0	0	1
	AO	0	0	0	0	**1**	0	4
	EY	0	0	3	2	0	**2**	0
	AY	0	2	0	0	4	0	▨

Table 4.24. Error rate for Speaker 2.

	Error Rate
with bigram LM	1/50
without bigram LM	25/50

For Speaker 2, without the LM the vowel phonemes OW and IY, and IY and AY respectively, can be considered to be reliable vowels. With the bigram LM, we obtain significant improvements in the recognition results, as six vowel phonemes UW, OW, IY, AE, AO and AY can be reliably recognized.

Table 4.25. Bigram speaker-dependent LM for Speaker 3.

		Vowel Phonemes						
		UW	OW	IY	AE	AO	EY	IY
Vowel Phonemes	UW	0	0	0	.5	.5	0	0
	OW	.6	0	0	0	0	.5	0
	IY	0	.5	0	0	.6	0	0
	AE	.5	0	0	0	0	0	.5
	AO	.5	0	.5	0	0	0	0
	EY	0	0	0	0	0	0	.5
	IY	0	.5	0	.5	0	0	0

Table 4.26. Confusion matrix with speaker-dependent LM for Speaker 3 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel utterances	UW	**5**	0	4	0	0	0	1
	OW	0	■	0	0	0	0	0
	IY	0	0	■	0	0	0	0
	AE	0	0	0	■	0	0	0
	AO	0	1	0	1	■	0	0
	EY	0	0	0	0	0	■	0
	AY	0	0	0	0	0	0	■

Table 4.27. Confusion matrix without speaker-dependent LM for Speaker 3 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	**0**	9	0	1	0	0	0
	OW	0	▨	0	0	2	0	0
	IY	0	0	▨	0	0	0	0
	AE	0	0	0	▨	0	0	0
	AO	0	0	0	4	**2**	0	4
	EY	0	0	0	1	0	**1**	0
	AY	0	0	0	4	0	0	**1**

Table 4.28. Error rate for Speaker 3.

	Error Rate
with bigram LM	7/40
without bigram LM	25/40

For Speaker 3, without language modeling the vowels OW, IY and AE can be considered as reliable vowels for recognition. With the introduction of the bigram LM, the vowel phonemes OW, IY, AE, EY and AY are never confused with each other.

Table 4.29. Bigram speaker-dependent LM for Speaker 4.

		Vowel Phonemes						
		UW	OW	IY	AE	AO	EY	IY
Vowel Phonemes	UW	0	0	0	.5	.5	0	0
	OW	.6	0	0	0	0	.5	0
	IY	0	.5	0	0	.6	0	0
	AE	.5	0	0	0	0	0	.5
	AO	.5	0	.5	0	0	0	0
	EY	0	0	0	0	0	0	.5
	IY	0	.5	0	.5	0	0	0

Table 4.30. Confusion matrix with speaker-dependent LM for Speaker 4 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	7	0	3	0	0	0	0
	OW	0	[illegible]	0	0	0	0	0
	IY	0	0	[illegible]	0	0	0	0
	AE	0	0	0	4	1	0	0
	AO	0	0	0	0	7	0	0
	EY	0	0	0	0	0	[illegible]	0
	AY	0	0	0	0	0	0	[illegible]

Table 4.31. Confusion matrix without speaker-dependent LM for Speaker 4 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	**0**	10	0	0	0	0	0
	OW	1	6	0	0	0	0	0
	IY	0	0	**3**	0	0	1	0
	AE	0	0	0	5	0	0	0
	AO	0	2	0	1	**0**	0	4
	EY	0	0	0	1	0	**1**	0
	IY	0	0	0	2	0	0	**3**

Table 4.32. Error rate for Speaker 4.

	Error Rate
with bigram LM	4/40
without bigram LM	22/40

For Speaker 4, we can identify the vowels OW and AE as reliable vowel sounds for the recognition task without LM. In the bigram LM case, the vowel phonemes OW, IY,EY and AY are never confused with each other.

4.4.3 Conclusion

The bigram LM increases recognition rates of the phoneme vowels. However, it is not possible to deduce a speaker-dependent LM that represents this dysarthric population, due to the variability in dysarthric speech. Different speakers produce different vowel sounds reliably, depending upon the degree and type of dysarthria. A better approach is to build a LM that is specific to a dysarthric speaker. From the language modeling experiments, we observe that the speaker specific LMs always improve the recognition rates of vowel phonemes. Furthermore, with the help of a proper LM it is possible to obtain a larger number of reliable vowel sounds than in the baseline acoustic recognition case. This suggests that including a bigram LM in the baseline recognition task can give us more reliable sounds that can be used as control triggers for an AAC device.

4.5 HMMs Trained on Dysarthric Speech

In this experiment, the seven vowel HMMs were trained on the dysarthric speech database. The utterances of each speaker were partitioned into a training set and test set.

Table **4.33** shows the number of utterances that were used as training set and test set for the HMM training and recognition tasks respectively.

Table 4.33. Partitioning the dysarthric speech database into training and test sets.

Vowels	Speaker 1		Speaker 2		Speaker 3		Speaker 4	
	Train set	**Test set**	**Train set**	**Test set**	**Train set**	**Test set**	**Train set**	**Test set**
UW	6	4	6	4	6	4	5	5
OW	7	5	8	4	6	4	6	5
IY	5	3	6	4	6	4	5	5
AE	5	2	8	4	6	4	5	5
AO	4	3	6	4	6	4	5	5
EY	5	4	6	4	6	4	5	5
AY	12	4	6	4	6	4	5	5

4.5.1 Classification Experiment

The purpose of this experiment is to investigate whether we can obtain reliable vowel recognition when the phoneme classifier consists of HMMs trained on the utterances of dysarthric speakers themselves. To achieve speaker independence, the vowel HMMs are trained on the training sets across all the four speakers. The test utterances of each of the four speakers are then passed through this phoneme classifier trained on dysarthric speech. The classification results obtained are used to deduce a confusion matrix for each speaker.

Table 4.34. Confusion matrix for Speaker 1 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	2	2	0	0	0	0	0
	OW	0	■	0	1	0	0	0
	IY	0	0	■	0	0	0	2
	AE	0	0	0	0	0	0	2
	AO	0	0	0	1	1	0	1
	EY	0	0	0	0	0	0	4
	AY	0	0	0	2	0	0	2

Table 4.35. Confusion matrix for Speaker 2 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	4	0	0	0	0	0	0
	OW	1	■	0	0	0	0	0
	IY	0	0	■	0	0	0	0
	AE	0	0	0	■	0	0	1
	AO	0	1	0	0	1	0	2
	EY	0	0	0	2	0	2	0
	AY	0	0	0	0	1	0	3

Table 4.36. Confusion matrix for Speaker 3 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	4	0	0	0	0	0	0
	OW	0	[illegible]	0	0	0	0	0
	IY	0	0	[illegible]	0	0	0	0
	AE	0	0	0	1	0	0	3
	AO	0	0	0	0	4	0	0
	EY	0	0	0	0	0	4	0
	AY	0	0	0	1	0	0	3

Table 4.37. Confusion matrix for Speaker 4 (shaded cells indicate reliable vowel sound).

		Vowel HMMs						
		UW	OW	IY	AE	AO	EY	AY
Vowel Utterances	UW	4	1	0	0	0	0	0
	OW	0	5	0	0	0	0	0
	IY	0	1	4	0	0	0	0
	AE	0	1	0	1	1	2	0
	AO	0	1	0	0	4	0	0
	EY	0	1	0	0	0	4	0
	AY	0	1	0	0	1	0	3

For Speaker 1, the vowel phoneme OW is the only one that is well recognized. The phonemes OW, IY and AO are never confused with each other. However, the phoneme sounds IY and AO do not have high recognition rates as a result of which they cannot be considered as reliable vowel sounds. For Speaker 2, the vowel phonemes OW, IY and AE are reliably recognized and are never confused with each other. Similarly for Speaker 3, it is OW, IY and AO, and for Speaker 4 it is IY and AO, which can be considered as reliable vowel phonemes.

4.5.2 Conclusion

In the above experiment, we trained a classifier based on the dysarthric speech utterances across four speakers. We were able to obtain reliable recognition from some vowel phonemes, for all speakers except Speaker 1. There is tremendous variability associated with dysarthric speech, primarily due to the articulatory imprecision associated with the dysarthric speakers. The inconsistency of dysarthric speech makes it difficult to train a reliable recognition system that can reliably recognize phonemes for a large population of dysarthric speakers.

5 Conclusions and Future Research

5.1 Conclusions

The main goal of this research was to evaluate the feasibility of using ASR techniques to obtain reliable recognition of dysarthric vowel utterances, with the long-term goal of incorporating this vowel recognizer into AAC and PC-based devices. It is very difficult to achieve high recognition accuracies for words or utterances spoken by severely dysarthric individuals, mainly due to the inconsistency of dysarthric speech. A different perspective would be to identify the vocalizations, which can be used as reliable sounds for the recognition task. A "reliable sound" in the context of this research is the one that the recognizer can consistently discriminate among the given vocalizations. To test this hypothesis, only vowel phoneme utterances obtained from four dysarthric speakers at the Madonna Rehabilitation Hospital, Nebraska were used for evaluation purposes.

The experimental results obtained from the phoneme-based vowel recognizer (without a LM) trained on normal speech indicate that for each of the speakers, at least two vowel phonemes can be identified as reliable vocalizations. The next task was to investigate whether the addition of language modeling information to the phoneme recognizer could increase reliability of vowel recognition. For this purpose a bigram, LM was incorporated into the recognition task. The results from the bigram LM recognition task imply that including language information does increase vowel recognition accuracy. In addition, we observe that the number of reliable vowel phonemes obtained for each of the speakers is more than those obtained from the recognition task without a LM. However, it is not possible to build a speaker-independent language modeling framework representing a large dysarthric speaker population. It is not possible for the LM to take into account all the variability associated with dysarthric speech. This implies that we can obtain the benefits of language modeling by building a speaker-dependent LM. The main advantage of language modeling is that it enables us to identify more control words that can be used as access controls for any AAC device. For example, for Speaker 1, the baseline recognition system (without LM) identified three vowel sounds (UW, OW and AE) as reliable access sounds. This means we can access a maximum of nine vowel pairs using a row-column access method. However, the bigram LM implementation gives us four reliable vowel sounds (OW, AE, EY and AY). This implies now we can access a maximum of 16 keys using a row-column access method.

		Column Control Vocalizations		
Row Control Vocalizations		UW	OW	AE
	UW	Key 1	Key 2	Key 3
	OW	Key 4	Key 5	Key 6
	AE	Key 7	Key 8	Key 9

(a)

		Column Control Vocalizations			
Row Control Vocalizations		OW	AE	EY	AY
	OW	Key 1	Key 2	Key 3	Key 4
	AE	Key 5	Key 6	Key 7	Key 8
	EY	Key 9	Key 10	Key 11	Key 12
	AY	Key 13	Key 14	Key 15	Key 16

(b)

Table 5.1. Row-column access for Speaker 1 using reliable vowel phonemes from (a) recognition task without LM (b) recognition task with LM.

Table 5.1 shows the construction of a row-column access type keypad using the reliable vowel phonemes obtained from the recognition tasks for Speaker 1.

An additional recognition experiment was carried out using a vowel recognizer trained on dysarthric speech of the four participants in the study. The dysarthric speech database was partitioned into training and testing sets for this experiment. The goal of this experiment was investigate if we could obtain reliable vowel phonemes from this recognition task. The results obtained indicated that although it was possible to obtain reliable vowel phoneme sounds for each of the speakers, the results were not consistent with those obtained from the vowel recognizer trained on normal speech. The training utterances in the dysarthric speech database do not result in a good representation of their representative vowel sounds. As a result, the HMM models used in the phoneme recognizer trained on dysarthric speech is not as well modeled as the HMMs used in the phoneme recognizer trained on dysarthric speech. However, if we are interested only in the feasibility of a recognizer trained on dysarthric speech, then it is possible to obtain reliable vowel phonemes.

5.2 Future work

The scope of this work was limited to testing the feasibility of using ASR techniques for reliable recognition of dysarthric speech. The long-term goal is to use these reliable sounds as control triggers for an array of AAC devices, including the personal computer (PC). All the conclusions and results obtained in this research were from the four dysarthric speakers selected to participate in this study. More reliable statistics to validate the conclusions above can be obtained by performing similar experiments over a larger population of dysarthric individuals. Another improvement would be to use words built around the reliable vowel sounds as access triggers. For example, if 'OW' is a reliable sound, we can use words like 'boat' and 'open' built around this vowel phoneme as control triggers. Of course, this requires more sophisticated algorithms to implement vowel spotting within a given word. Further, we can evaluate the performance of a context dependent phoneme-based vowel recognition system to investigate which words can be most reliably used as control triggers in the AAC device.

BIBLIOGRAPHY

[1] Garofolo, John S., *et al.*, *DARPA-TIMIT Acoustic-Phonetic Continuous Speech Corpus*, Documentation for "NIST Speech Disc 1-1.1", February, 1993.

[2] Deller, J.R., Hansen, J.H.L. and Proakis, J.G., *Discrete-Time Processing of Speech Signals*, New York: IEEE Press, 2000.

[3] Rabiner, L.R. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, pp. 257-286, 1989.

[4] Huang, X.D., Ariki, Y., and Jack, M.A., *Hidden Markov Models for Speech Recognition*, Edinburgh : Edinburgh University Press, 1990.

[5] Wu, Y., Ganapathiraju, A., and Picone, J., "Baum-Welch Re-Estimation of Hidden Markov Model," Institute for Signal and Information Processing, 1999.

[6] Steve Young *et al.*, *The HTK Book (for HTK Version 3.1)*, Cambridge University Engineering Department, 2002.

[7] Baker, J.K., "Stochastic Modeling for Automatic Speech Understanding." In D. R. Reddy, ed., *Speech Recognition*, New York: Academic Press, pp. 521-542, 1975.

[8] Jelinek, F., "Continuous Speech Recognition by Statistical Methods," *Proceedings of the IEEE*, vol. 64, pp. 532-556, April 1976.

[9] Picone, J., "Continuous Speech Recognition Using Hidden Markov Models," *IEEE Acoustics, Speech, and Signal Processing Magazine*, vol. 7, pp. 26-41, July 1990.

[10] Liporace, L.A., "Maximum Likelihood Estimation for Multivariate Observations of Markov Sources," *IEEE Transactions on Information Theory*, vol. 28, pp. 729-734, September 1982.

[11] Juang, B.H., Levinson, S.E. and Sondhi, M.M., "Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Chains," *IEEE Transactions on Information Theory*, vol. 32, pp. 307-309, March 1986.

[12] Bakis, R., "Continuous Speech Word Recognition via Centisecond Acoustic States," *Proceedings of the 91st Annual Meeting of the Acoustical Society of America*, Washington D.C., 1976.

[13] Baum, L.E., "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes," *Inequalities*, vol. 1, pp. 1-8, 1972.

[14] Davis, S.B. and Mermelstein, P., "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, pp. 793-807, 1983.

[15] Murphy, K., Bayes Net Toolbox for Matlab [software]. Retrieved February 2002 from the World Wide Web :
http://www.cs.berkeley.edu/~murphyk/Bayes/bnt.html.

[16] Brookes, M., VOICEBOX: Speech Processing Toolbox for MATLAB [software]. Retrieved February 2002 from the World Wide Web :
http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html.

[17] Schroeder, M.R., "Recognition of Complex Acoustic Signals," *Life Science Research Reports*, vol. 55, pp. 323-328, 1977.

[18] IBM and Microsoft, Waveform Audio File Format, Multimedia Programming Interface and Data Specification v1.0, 1991.

[19] NIST, Sphconvert.zip v2.1 (for Wintel) [software]. Retrieved February 2002 from the World Wide Web:
ftp://ftp.ldc.upenn.edu/pub/ldc/misc sw.

[20] Lee, K.F., "Speaker Independent Phone Recognition Using Hidden Markov Models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, pp. 1641-1648, November 1989.

[21] Treviranus, J., Shein, F., Haataja, S., Parnes, P. and Milner, M., "Speech Recognition to Enhance Computer Access for Children and Young Adults who are Functionally Non-speaking," *Proceedings of RESNA 14th Annual Conference*, pp. 308-310, 1991.

[22] Deller, J.R., Hsu, D. and Ferrier, L., "On the Use of Hidden Markov Modeling for Recognition of Dysarthric Speech," *Computer Methods and Programs in Biomedicine,* vol. 35, no. 2, pp. 125-139, 1991.

[23] Lorei, Marcus, "Phonetic Modeling of Dysarthric Speech." Master's Thesis, Michigan State University, 1995.

[24] Manasse, Nancy, "Speech Recognition." Barkley AAC training Laboratory, University of Nebraska-Lincoln, May 1999.

[25] Carlson, G.S. and Bernstein, J., "Speech Recognition of Impaired Speech," *Proceedings of RESNA 10th Annual Conference*, pp. 103-105, 1987.

[26] Ferrier, L.J., Jarell, N., Carpenter, T. and Shane, H., "A Case Study of a Dysarthric Speaker using the Dragon Dictate Voice Recognition System," *Journal for Computer Users in Speech and Hearing,* vol. 8, no. 1, pp. 33-52, 1992.

[27] Ferrier, L.J., Shane, H.C., Ballard, H.F., Carpenter, T. and Benoit, A., "Dysarthric Speakers' Intelligibility and Speech Characteristics in Relation to Computer Speech Recognition," *Augmentative and Alternative Communications,* vol. 11, pp. 165-173.

[28] Doyle, P., Leeper, H., Kotler, A., Thomas-Stonell, N., OíNeill, C., Dylke, M. and Rolls, K., "Dysarthric Speech: A Comparison of Computerized Speech Recognition and Listener Intelligibility," *Journal of Rehabilitation Research and Development*, vol. 34, no. 3, pp. 309-316, 1997.

[29] Poritz, A.M., "Hidden Markov Models: A Guided Tour," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, New York, vol. 1, pp. 7-13, 1988.

[30] Johnston, David, Cool Edit 2000 [software]. Copyright Syntrillium Software Corporation, 2000.

[31] Fant, C.G.M., "Acoustic Description and Classification of Phonetic Units," *Ericcson Technics*, no. 1, 1959.

[32] Klatau, Aldebaro, "Survey of Results on Phoneme Classification and Recognition Using TIMIT." University of California San Diego, 2000. Retrieved May 2002 from the World Wide Web: http://speech.ucsd.edu/aldebaro/papers/.