NTNU

Norwegian University of
Science and Technology

Computer Assisted Pronunciation Training

Evaluation of non-native vowel length pronunciation

Eivind Versvik

Master of Science in Electronics

Submission date: June 2009

Supervisor: Magne Hallstein Johnsen, IET

Co-supervisor: Ingunn Amdal, IET

Norwegian University of Science and Technology
Department of Electronics and Telecommunications

Problem Description

Mobility over country borders is increasing. People on the move have various mother tongues, but all have the same need for quickly learning the new native language to be able to participate in the society both socially and professionally. Computer assisted language learning can help people to achieve these skills. Computer assisted pronunciation training (CAPT) has been shown to be effective for non-natives to learn and evaluate pronunciation details of a new language. However, no automatic pronunciation evaluation system exists for non-native Norwegian presently.

When designing a CAPT system it is important to select pronunciation exercises that are beneficial for the users. Many pronunciation properties that are crucial in Norwegian may not be important in the user's mother tongue. One such property is the contrast between long and short vowels. This contrast is phonemic in Norwegian, but not important in many other languages.

This master assignment consists of the following tasks:
• Record a small database of both native and non-native speakers pronouncing Norwegian words of minimal pairs with respect to vowel length.
• Design a classifier to detect whether the user (both native and non-native) pronounces a long or a short vowel.
• Derive suitable acoustic-phonetic features for the classifier. A feature set previously derived by Dynamic Time Warping should be expanded and improved. A similar feature set should be generated using Hidden Markov Models and the two sets should be compared.
• Run experiments to compare and evaluate the different classifier set-ups. Compare the results for native and non-native speakers.
• Implement a chosen set-up in an existing Java-based CAPT-demo.

Assignment given: 15. January 2009
Supervisor: Magne Hallstein Johnsen, IET

Abstract

Computer Assisted Pronunciation Training systems have become popular tools to train on second languages. Many second language learners prefer to train on pronunciation in a stress free environment with no other listeners. There exists no such tool for training on pronunciation of the Norwegian language.

Pronunciation exercises in training systems should be directed at important properties in the language which the second language learners are not familiar with. In Norwegian two acoustically similar words can be contrasted by the vowel length, these words are called vowel length words. The vowel length is not important in many other languages. This master thesis has examined how to make the part of a Computer Assisted Pronunciation Training system which can evaluate non-native vowel length pronunciations.

To evaluate vowel length pronunciations a vowel length classifier was developed. The approach was to segment utterances using automatic methods (Dynamic Time Warping and Hidden Markov Models). The segmented utterances were used to extract several classification features. A linear classifier was used to discriminate between short and long vowel length pronunciations. The classifier was trained by the Fisher Linear Discriminant principle.

A database of Norwegian words of minimal pairs with respect to vowel length was recorded. Recordings from native Norwegians were used for training the classifier. Recordings from non-natives (Chinese and Iranians) were used for testing, resulting in an error rate of 6.7%. Further, confidence measures were used to improve the error rate to 3.4% by discarding 8.3% of the utterances. It could be argued that more than half of the discarded utterances were correctly discarded because of errors in the pronunciation. A CAPT demo, which was developed in an former assignment, was improved to use classifiers trained with the described approach.

Preface

This master thesis was carried out at the Department of Electronics and Telecommunication, at the Norwegian University of Science and Technology (NTNU). The master thesis is the completion of my master in Electronics.

I would especially like to thank Ingunn Amdal for guiding and helping me. I would also like to thank Magne H. Johnsen for all the useful advice. Further, I would like to thank Line Adde, who labeled the speech. Finally, I would like to thank all who were recorded during this work.

Contents

1 Introduction **1**
- 1.1 Motivation and background . . . 1
- 1.2 Approach . . . 3
- 1.3 CAPT-demo . . . 4
- 1.4 Report structure . . . 4

2 Theory **6**
- 2.1 Preprocessing - acoustic features . . . 7
- 2.2 Segmentation: Dynamic Time Warping . . . 9
- 2.3 Segmentation: Hidden Markov Models . . . 13
- 2.4 Classification features . . . 18
- 2.5 Automatic classification: Fisher Linear Discriminant . . . 20
- 2.6 Confidence measures . . . 23

3 Recording and preparing a database **27**
- 3.1 Word selection . . . 27
- 3.2 Recording a database . . . 28
- 3.3 Preparing the database . . . 30
- 3.4 Pronunciation evaluation by a judge . . . 31

4 Method for classification of vowel length **35**
- 4.1 Preprocessing: acoustic features . . . 35
- 4.2 Segmentation . . . 36
- 4.3 Classification features and classification . . . 37
- 4.4 Confidence . . . 39

5 Results and discussion **40**
- 5.1 Segmentation results: teacher voice . . . 40
- 5.2 Classification results . . . 42
- 5.3 Inspection of classification errors . . . 45
- 5.4 Evaluation of classification features . . . 50

5.5 Classification results with confidence 53

6 Conclusion **60**

References **62**

Appendices **64**

A Information about the databases **65**
A.1 General information 65
A.2 List of recorded words and number of recorded words 66
A.3 Information about the speakers 67
A.4 Information given to the speaker before recording 68

B HMM and DTW information **73**
B.1 DTW normalization 73
B.2 Hidden Markov Model training data 73
B.3 HTK MFCC parameterization 75

C Results **76**

D CAPT-demo **78**

List of Figures

1.1 Visual feedback with Tell Me More 2
1.2 The 4 steps done to classify vowel length pronunciations . . . 3

2.1 Illustration of the source-filter model [21]. 7
2.2 Steps for calculating MFCCs 8
2.3 Calculation of accumulated distortion 10
2.4 How DTW calculates the minimum overall distortion 10
2.5 Illustration of the best path 11
2.6 Illustrative example of the 6 steps done to segment the users response 12
2.7 Different ways of restricting the number of insertions and deletions in DTW 12
2.8 Example of a Markov chain with 3 states 14
2.9 Example of probability density functions 15
2.10 Example of a 3 state model for a phoneme 15
2.11 Calculation of the likelihood 16
2.12 Illustration of Viterbi decoding 17
2.13 Example of FLD class separation 21
2.14 Calculation of Word Confidence Score 24
2.15 Example of using the LDA unsure window 26

3.1 The recording interface 29
3.2 Preparation of the database 31
3.3 The decision interface 32

4.1 Classification of vowel length pronunciations 35

5.1 Manually segmented teacher voice: Vowel and Consonant duration. 42
5.2 Example of one of the utterance with significant segmentation error 46

5.3 HMM segmentation with short and long HMMs representing the Vowel. (Short Vowel Long Model = Short vowel length, long HMM) 51
5.4 Figure to the left: The Vowel LLR for all non-native speakers. Figure to the right: The Vowel LLR for utterances that were classified wrong. 54
5.5 This figure shows the distribution of the LDA score calculated when classifying the non-native speakers 56

D.1 The graphical user interface 78

List of Tables

3.1 This table shows the conflicting labels 33

5.1 HMM segmentation of the teacher voice compared with manual segmentation 41
5.2 Classification error rate with different segmentation methods. . 43
5.3 Classification results using rotation or Norwegian trained classifier 45
5.4 This table shows the error rate per speaker 47
5.5 This table shows the error rate for non-sense and regular words. 48
5.6 This table shows the 7 word pairs with highest error rate. . . . 49
5.7 This table shows the error rate with segmentation done with only long or short HMMs. 50
5.8 Error rate with different classification features. 52
5.9 Number of discarded test utterances with a given Vowel LLR threshold. 54
5.10 Number of discarded test utterances for a given unsure window. 57
5.11 Error rate with the confidence measures with chosen thresholds 58

A.1 General information about the recording conditions and equipment 65
A.2 What country the speakers were from 66
A.3 List of the selected words 66
A.4 Information about the speakers 67

C.1 Number of discarded test utterances for a given Vowel duration threshold with long HMM segmentation. 76
C.2 Number of discarded test utterances for a given Vowel duration threshold with short HMM segmentation. 77

Abbreviations

A list of all abbreviations used in this thesis is shown below:

CALL = Computer Assisted Language Learning

CAPT = Computer Assisted Pronunciation Training

MFCC = Mel-Frequency Cepstral Coefficient

PDF = Probablity Density Function

HMM = Hidden Markov Model

DTW = Dynamic Time Warping

HTK = Hidden Markov Model Toolkit

LLR = Log Likelihood Ratio

CES = Confidence Evaulation Score

GUI = Graphical User Interface

FLD = Fisher Linear Discriminant

LDA = Linear Discriminant Analysis

LV = Long Vowel

SV = Short Vowel

LL = Log Likelihood

L2 = Second Language

Chapter 1

Introduction

This chapter is the introduction to the report. Section 1.1 presents the motivation and background for the master thesis. Section 1.2 explains the chosen approach in the thesis and section 1.3 presents what was done with a CAPT-demo. At last section 1.4 shows the structure of this report.

1.1 Motivation and background

Computer Assisted Language Learning (CALL) systems have become popular tools to train pronunciation in the second language (L2) because they offer extra learning time and material as well as the possibility to practice in a stress-free environment [12]. Pronunciation training is an important aspect of learning a new language. Wrong pronunciation can be confusing and lead to difficulties when communicating. Thus a CALL system should include pronunciation training. This section describes the motivation and background for including a pronunciation training system in CALL systems.

By integrating Computer Assisted Pronunciation Training (CAPT) into CALL systems, the computer can evaluate the student's speech and react with appropriate feedback on the pronunciation. With feedback the learning process will be more realistic and engaging, and errors in the student's pronunciation that are not noticeable for the student through playback can be corrected. A group of people who were trying to improve their pronunciation using a CALL system with or without CAPT was studied in [11]. The study claims that the CAPT system was effective in correcting the errors addressed in the training, which indicates that a CAPT system is useful in a CALL system.

There are several CALL systems for foreigners trying to learn Norwegian, some of which can be found in [8]. These CALL systems make L2 learners

learn through interaction, playing games and listening to a native speaker. However none of these systems include a CAPT system, which limits their ability to learn the student correct pronunciation. No CAPT system exists for Norwegian; however there are several systems for languages like French, Spanish and English. An example of a program that includes pronunciation training is 'Tell Me More' [9]. 'Tell Me More' tells the user what words are badly pronounced and scores the pronunciation, see figure 1.1. By giving feedback with score the user will get an idea how good the pronunciation is and be able to correct it by repeating the exercise.

Figure 1.1: Example of visual feedback on the pronunciation including score (from Tell Me More). The word 'servirle' is outlined in red because it has been pronounced wrong and the low score indicates that the pronunciation is not good.

There are large pronunciation variations between different languages; therefore each language needs its own specialized CAPT system. A system designed for English will not work for Norwegian, because the languages have different properties in the pronunciation which are important. Properties in a language which L2 learners are not familiar with can be difficult to learn without training and feedback. Thus it is important to include systems which can train L2 learners on these properties.

One such property in Norwegian is the vowel length. Two acoustically similar words can be contrasted by the vowel length in the stressed syllable. Two such words are 'Hanne' (girl's name) and 'hane' (rooster) where 'Hanne' is a short vowel length word and 'hane' is a long vowel length word.

Two words, which are only contrasted by the vowel length, are defined as minimal pairs with respect to vowel length. The vowel length is defined as the perceived opinion by a human about the pronounced vowel length while the vowel duration is how long the vowel lasts in terms of seconds. In this thesis it is often referred to the vowel and consonant with uppercase letter:

the Vowel is defined as the vowel in the stressed syllable of the word, while the Consonant is the consonant following the Vowel. It is the Vowel which contrasts between two minimal pairs with respect to vowel length.

1.2 Approach

This report focuses on how to automatically evaluate non-native vowel length pronunciations. To evaluate non-native vowel length pronunciations a classifier has to detect whether the pronunciations are long or short vowel length. All of the steps included to classify vowel length pronunciations are shown in figure 1.2.

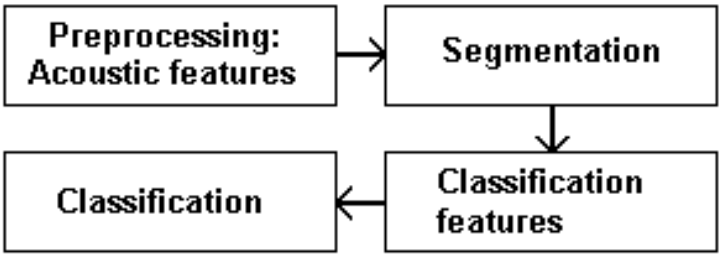


Figure 1.2: The 4 steps done to classify vowel length pronunciations

The vowel length pronunciations are first preprocessed to produce acoustical features. Following the preprocessing step comes a segmentation step which finds all of the phoneme[1] boundaries[2] in the word. Two different methods are tested to do the segmentation: Mel-Frequency Cepstral Coefficients and Hidden Markov Models.

Further, classification features are extracted from the segmented word. A long vowel duration is not necessarily synonymous with a long vowel length and it is difficult to automatically find the Vowel duration with a high precision and reliability. Therefore it was assumed that the Vowel duration is not the only discriminative factor in vowel length pronunciations. [16] argues that the duration of the consonant following the Vowel is an important factor. It is also assumed that a short and long vowel can contain some acoustical differences.

Several classification features are extracted to find out which features are important for the vowel length. All of the features are combined

[1]The smallest phonetic unit in a language that is capable of conveying a distinction in meaning [2].

[2]A phoneme boundary is the place where one phoneme ends and another one begins.

to a classification feature vector and used to classify the vowel length pronunciations. A classifier needs to find out how much each of the classification features should affect the classifications. How much a classification feature affects a classification is called the weight of the feature. The problem of finding the best weighing of the classification features is too difficult for a human when the number of classification features is high. For that reason the classifier is trained using an automatic method called Fisher Linear Discriminant.

To test the proposed classification method a database was recorded of both native and non-native speakers pronouncing Norwegian words of minimal pairs with respect to vowel length. In a realistic situation the L2 learner is a non-native trying to learn Norwegian. Thus to test how good the classifier performs in a realistic situation, non-natives with little Norwegian background are used. The native speakers are used to train the classifier and to check that the classification method is reliable on natives.

The proposed classification method can not be expected to be error free. Therefore confidence measures are used to prune out utterances[3] with high potential of resulting in a classification error. A classification error happens when the classifier decides a vowel length pronunciation is short vowel length while a human judge perceive the pronunciation as long vowel length, and vice versa.

1.3 CAPT-demo

A CAPT demo was implemented in a former assignment which is described in [15]. As a part of this work the demo was improved and updated with some of the methods used in this thesis. The improved demo is described in appendix D and will not be discussed or evaluated in the rest of the report.

1.4 Report structure

The report is structured as follows:

- Chapter 2: Theory

  This chapter explains the theory behind preprocessing, Hidden Markov Models, Dynamic Time Warping, the classification features, Fisher Linear Discriminant and the confidence measures.

[3] An utterance is a complete unit of speech in spoken language. It is generally but not always bounded by silence. [3]

- Chapter 3: Recording and preparing databases

  This chapter presents how the recording and preparing of the native and non-native databases was done. It also presents the labeling of the non-native database.

- Chapter 4: Method

  This chapter presents the specific choices in the classification method and confidence measures. It also presents how the classification method and confidence measures were evaluated.

- Chapter 5: Results and discussion

  This chapter presents the results and discusses them.

- Chapter 6: Conclusion

  This chapter presents a conclusion.

Chapter 2

Theory

The main purpose of this thesis is to automatically classify vowel length pronunciations. This chapter will present the theory, while chapter 4 explains how it is used.

The classification of a vowel length pronunciation can be split into the 4 steps shown in figure 1.2. The steps are explained below:

1) Acoustic features: Extract features from the sound signal which contains all the essential information for the segmentation step. The acoustic feature is explained in section 2.1.

2) Segmentation: The reason for doing segmentation/labeling is to find information about the sound signal that can be used to make classification features. Two different methods are used to segment the sound signal. The first method is called Dynamic Time Warping and is explained in section 2.2. The second method is called Hidden Markov Models and is explained in section 2.3.

3) Classification features: Extract the features that are used to classify the word. What kind of features are used is explained in section 2.4.

4) Classification: Use the classification features to classify words. The theory behind automatic classification (with training data) and an explanation on how automatic classification is used to classify vowel length words is explained in section 2.5.

The classifier is bound to make some classification errors. If it is possible to detect utterances with pronunciations that are likely to result in classification errors the utterances could either be discarded or treated specially. Section 2.6 presents some confidence measures which are used to discard problematic utterances.

2.1 Preprocessing - acoustic features

The sound wave is not used directly when recognizing or segmenting words, but a feature containing the most important information is calculated and used instead. This section will present the feature used for segmenting words.

The sound wave emitted when speaking can be captured by a microphone. It is a complete description of a pronounced word and it is possible to reproduce the word with a loudspeaker. However the sound wave contains a lot of redundancy and it is difficult to detect phonemes or words in a sound wave directly because of huge variations between different speakers. The ideal feature should be an exact representation of the wave form for the duration it covers. If the feature is going to be used for detecting specific events, the feature should have small variations for same kind of events and vice versa.

Mel-frequency Cepstral Coefficients (MFCCs) is the acoustic feature used in this thesis. The representation is going to be used in both Hidden Markov Models and Dynamic Time Warping. MFCCs are based on the source-filter model. The source is assumed to be the vocal chord that excites either pulses or white noise, while the filter model is the vocal tract which modifies the excitations into phonemes. Figure 2.1 illustrates the source-filter model. The filter changes over time to make different sounds. By assuming that the sound signal is stationary over short time periods (around 5-10 ms) the filter coefficients can be calculated. The filter coefficients are used to find MFCCs.

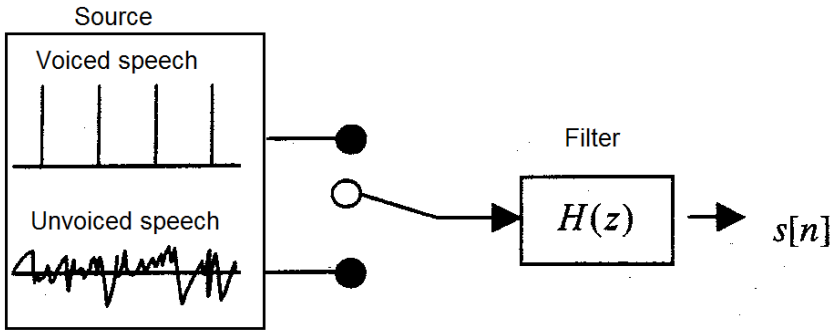


Figure 2.1: Illustration of the source-filter model [21].

The MFCCs are compact features and the coefficients are a good representation for phonemes, because MFCCs for a specific phoneme are in general distinct from other phonemes. It is not an exact representation of the speech, because the sound wave is not stationary, which is assumed when calculating the feature. The value of a MFCC represents the magnitude of energy over a frequency and time range. The time range is often referred to

as the window size. A common choice is to calculate around 12 coefficients that represent different frequencies for each window. The window size used to calculate a MFCC is usually around 15-20 ms, with a step size of 5-10 ms for each new MFCC calculation.

The step size is important because it decides the segmentation resolution of speech. Smaller step size leads to better theoretical resolution, but at some point the resolution will saturate due to unclear boundaries between different phonemes. The window size is a choice between time and frequency resolution. A longer window size leads to better estimate of the frequency in the speech, thus giving a better representation of the phonemes. However too long windows will make the assumption that speech is stationary false. The window size also influences the resolution, because the boundaries between phonemes will become diffuse as the window size increases.

The coefficients can be derived as shown in figure 2.2.

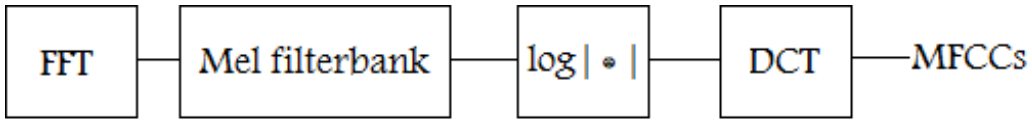


Figure 2.2: Steps for calculating MFCCs

The boxes are explained below. Information acquired from [18].

1) Take the Fourier transform of (a windowed excerpt of) a signal.

2) Map the powers of the spectrum obtained above onto the Mel scale, using triangular overlapping windows.

3) Take the logs of the powers at each of the Mel frequencies.

4) Take the discrete cosine transform of the list of Mel log powers, as if it were a signal.

5) The MFCCs are the amplitudes of the resulting spectrum.

Temporal changes in the spectra play an important role in human perception [21]. Hidden Markov Models assume that a feature vector calculated at a time is independent of the past (see section 2.3). This assumption neglects a lot of information, but some of it can be included through dynamic features. Delta coefficients measure the change in coefficients over time. The 1st-order and 2nd-order delta MFCC can be

computed by respectively (2.1) and (2.2).

$$\Delta c_k = c_{k+2} - c_{k-2} \tag{2.1}$$

$$\Delta\Delta c_k = \Delta c_{k+1} - \Delta c_{k-1} \tag{2.2}$$

where c_k is a MFCC

Delta and delta delta coefficients are often added to the MFCC feature vector. Another common choice is to add the log energy of the sound signal as one more MFCC coefficient. Further details about MFCCs are found in [21].

The result after preprocessing a sound file is a acoustic feature vector with MFCCs which contain all the information from the sound signal that is used for segmentation by Hidden Markov Models and Dynamic Time Warping.

2.2 Segmentation: Dynamic Time Warping

Dynamic Time Warping (DTW) is an algorithm for measuring the similarity between two patterns. The algorithm is used to segment words. DTW was chosen because the algorithm is easy to implement and can be solved computationally quickly. For more details about DTW see [17]. The next paragraphs will explain the basic principle about DTW and show how it is used for segmentation.

The DTW algorithm calculates the minimum overall distortion $D(n, m)$ between two feature vectors, $(x_1, x_2..x_n)$ and $(y_1, y_2..y_m)$, which may vary in length. The distortion between the two feature vectors at a point (i, j) can be calculated using a distortion measure $d(x_i, y_j)$. The accumulated distortion from point $(1, 1)$ to a point (i, j) is defined as $D(i, j)$.

DTW exploits the fact that the accumulated distortion to point (i, j) can be calculated from previously calculated accumulated distortions. How the accumulated distortion to point (i, j) is calculated is illustrated in figure 2.3 and shown in the following equation:

$$D(i, j) = min[D(i-1, j), D(i-1, j-1), D(i, j-1)] + d(x_i, y_j) \tag{2.3}$$

By initializing the accumulated distortion at $D(1, 1)$ to be $d(1, 1)$ and defining $D(z, w)$ where $z, w < 1$ to be infinite, the overall distortion $D(n, m)$ can be calculated as shown in the pseudocode in figure 2.4. The distortion measure $d(x_i, y_j)$ was chosen as the Euclidian distance squared.

When calculating accumulated distortion a choice is made whether to do a match, insertion or deletion, see figure 2.3 for term explanations. If the choices are stored, the path with least distortion, called 'best path', can be

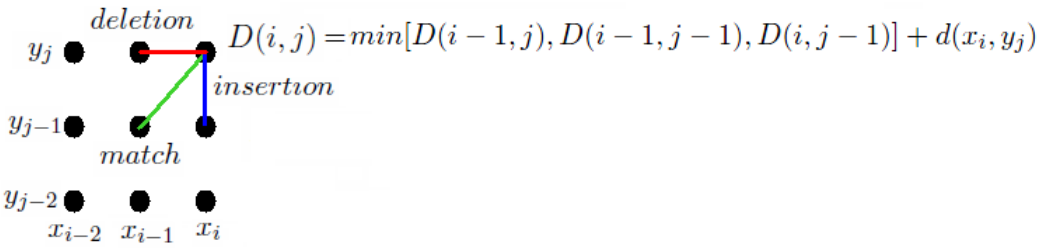


Figure 2.3: A figure showing how calculation of accumulated distortion $D(i,j)$ is performed in Dynamic Time Warping. The algorithm uses previously calculated scores to find the optimal score to each point (i,j).

```
for i = 1 to n
    for j = 1 to m
        D(i,j) = d(xi,yj) + min [D(i-1,j),
                                 D(i,j),
                                 D(i,j-1)]

D(n,m) = minimum overall distortion
```

Figure 2.4: How DTW calculates the minimum overall distortion

found. The path is found by starting at point (n,m) and going back to point (i,j) by checking the choice made at each point; this method is called 'backtracking'. An example of how the best path can look is illustrated in figure 2.5.

The DTW algorithm can be used to segment phonemes in a pronunciation if the phonemes in the word are known. To segment a word with unknown phoneme boundaries a reference pronunciation of the same word is also needed. The unsegmented response is in the next paragraphs referred to as the user response.

DTW can be used to segment words because comparison of feature vector sections with equal phonemes should result in low distortion, while sections of the feature vector with different phonemes should result in high distortion. DTW finds the path with lowest distortion, therefore the best path will contain similar phonemes compared with each other. If it is known which section of the reference feature vector belongs to a phoneme, it is possible to find out which section of the user feature vector belongs to the phoneme. The time of the feature vector calculation is known, thus the phoneme boundaries are given.

How to perform the segmentation is explained below in 6 steps and

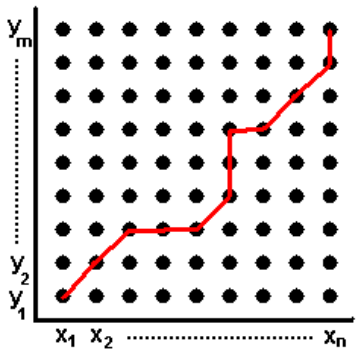


Figure 2.5: Illustration of the best path through two feature vectors $(x_1, x_2..x_n)$ and $(y_1, y_2..y_m)$.

illustrated in figure 2.6.

a) Find out when reference phoneme number i ends.

b) Find the reference feature vector calculated at that time.

c) Find out where the reference feature vector was used in the best path gotten from DTW.

d) Find out what user feature vector was compared with the reference feature vector.

e) Find out at what time the user feature vector was calculated.

f) Assign phoneme number i as being after phoneme number i-1 and ending at the found time, then start from a) with phoneme number i+1.

Modified DTW was introduced to get a restriction in the number of insertions and deletions. The reason for implementing restriction in the algorithm is that the unmodified DTW can compare a small section of the reference word with a large section of user word. This can be fixed by restricting how many insertions and deletions the algorithm can do at a time.

One way of restricting insertions and deletions is to force a match before any insertions and deletions can be performed, as shown in figure 2.7. Experience from earlier work showed that 2-1 DTW was too restrictive[15]. The 3-1 DTW had another problem where the best path did not originate from the start $(0, 0)$. Therefore a combination was used called 'Restricted DTW', which opens up reasonable path options from 2-1 and 3-1 DTW.

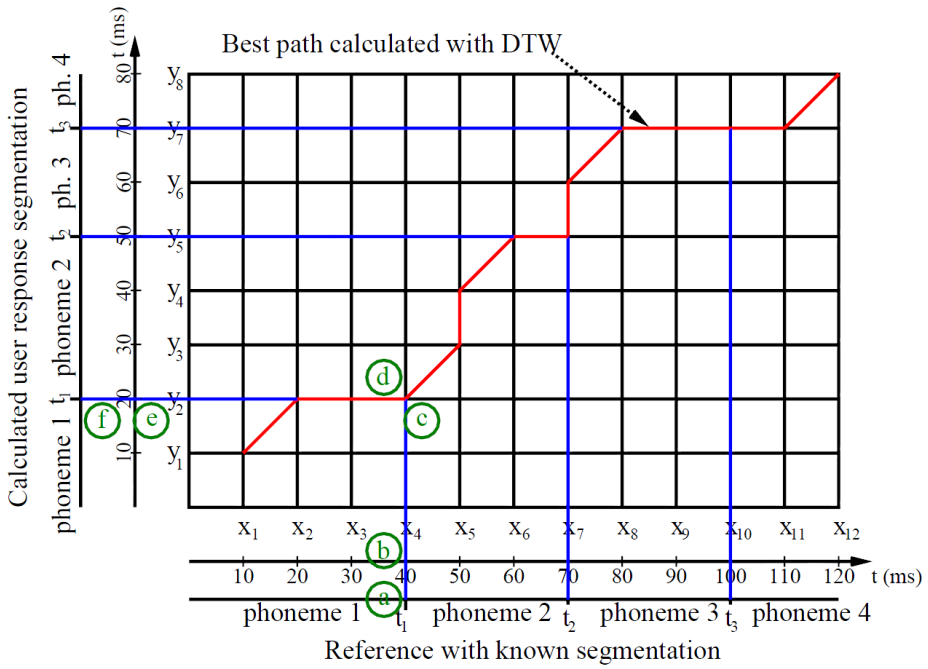


Figure 2.6: Illustrative example of the 6 steps done to segment the users response

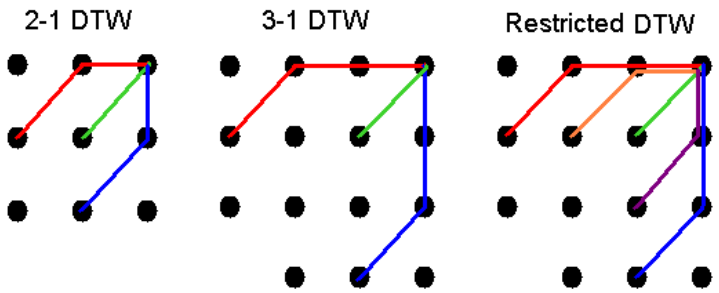


Figure 2.7: Different ways of restricting the number of insertions and deletions in DTW.

2.3 Segmentation: Hidden Markov Models

Hidden Markov Models (HMMs) are used in temporal pattern recognition within speech. HMMs is used to do segmentation of speech. Relevant theory of HMMs will be presented in this section. For further details about HMMs see [21], [6] and [17].

A Markov chain models a class of random processes that incorporates a minimum amount of memory without being completely memoryless [21]. A Markov chain can be described by a finite state process with transition between states specified by a probability function. In HMMs it is assumed that the process is a first-order Markov chain, which implies that the probability of being in a state s at time t is only dependent on the preceding state:

$$P(s_t|s_{t-1}, s_{t-2}...s_1) = P(s_t|s_{t-1}) \tag{2.4}$$

A second assumption is that the probability of an output o_t at a state s_t is only dependent on the current state:

$$P(o_t|o_{t-1}, o_{t-2}...o_1, s_t, s_{t-1}...s_1) = P(o_t|s_t) \tag{2.5}$$

An example of a small Markov chain is shown in figure 2.8. Each state has a set of transition probabilities which specifies the probability of entering a state given the current state. In speech each state is usually associated with events that are to be detected, which in this case are phonemes. If a state is associated with a phoneme it is common to include (at least) one state for every phoneme in a language. It is also important to include states for silence and noise; otherwise it can be confused with phonemes. The transition probability from a state to the same state models the time spent at the phoneme or noise. The transition probability from a state to other states models the probability of going from one phoneme to another phoneme. The matrix which describes all transition probabilities will be referred to as the transition matrix.

The output at a state is the mapping from states (or phonemes) to feature vectors or given the feature vectors it can be a mapping from feature vectors to phonemes. In HMMs it is assumed that the output in each state is 'hidden', i.e. the output at a state can not be directly observed. The output is stochastic according to some kind of probability density function (PDF). Thus there exists no one to one mapping from state to output. Given an output o_t it is possible to calculate the probability that the output belong to a state by finding the probability that the output belongs to the state's

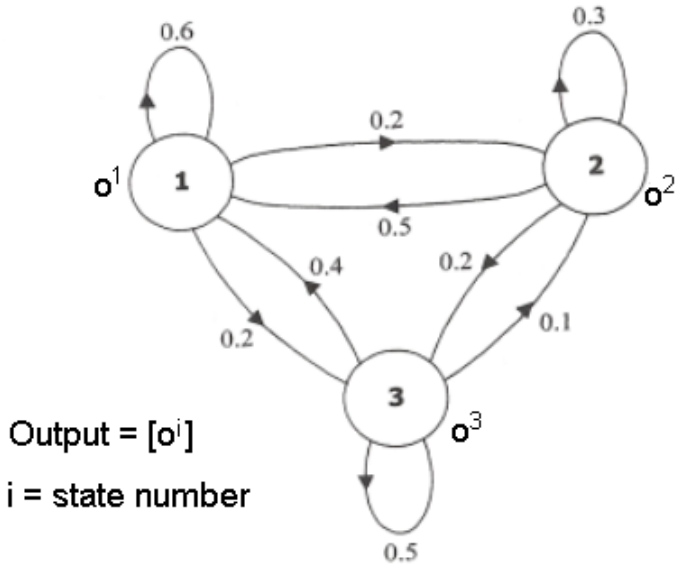


Figure 2.8: Example of a Markov chain with 3 states giving an output o^i. The probability of entering a state, given the last state, is shown as a number along the arrows.

PDF. It is assumed that the feature vectors will approximately be distributed according to the Gaussian PDF. A Gaussian PDF is fully described by its mean and variance. An example of the output at a state is shown in figure 2.9.

The feature vectors used in HMMs are often N-dimensional, which implies that the Gaussian PDFs are multidimensional with N dimensions. For example a MFCC vector with 1 MFCC calculated for each window is 1 dimensional, while a MFCC vector with N MFCCs calculated per window is N dimensional. Thus for a N dimensional MFCC vector there will be a N dimensional Gaussian PDF.

Feature vectors extracted from a phoneme will not fit perfectly to one Gaussian PDF. The mismatch happens mostly because of large pronunciation variations between speakers. To accommodate for variations more than one PDF for each state can be included. The number of PDFs used per state is usually referred to as the number of mixture components per state. An example of a state with 2 dimensional PDFs and 5 mixture components is shown in figure 2.9.

The spectral representations at different times in a phoneme pronunciation can be very dissimilar. The start, middle and end of a phoneme is for example distinct. To better represent the phoneme at different times a 3

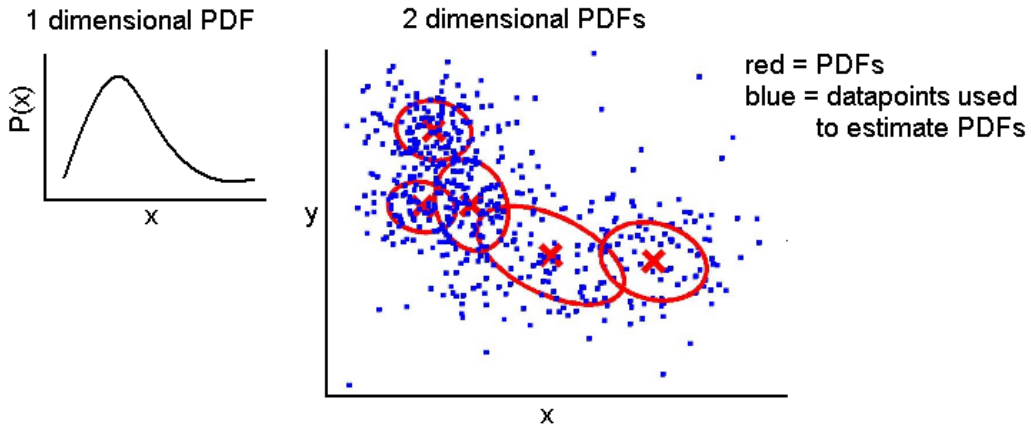


Figure 2.9: To the left is an example of output x spread according to a probability density function at a state. To the right is an example of a state with many output points (x, y) spread according to 2 dimensional Gaussian PDFs with several mixture components.

state model for each phoneme can be used. The start model can only transit to start or middle state, while middle state can transit to middle or end state. The end state can transit to a new start model. The phoneme model is illustrated in figure 2.10.

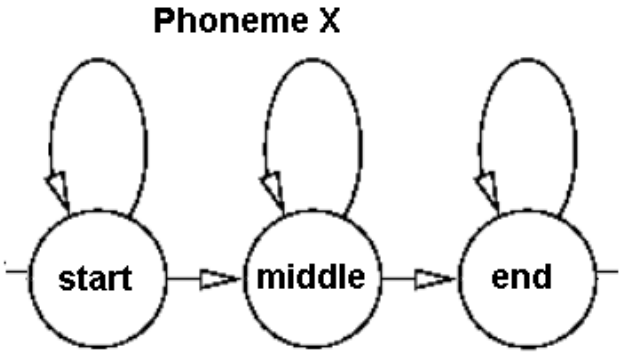


Figure 2.10: Example of a 3 state model for a phoneme. The model has 3 states following each other with transitions only allowed to next or same state.

As presented earlier the states are assumed to give an output according to a probability density function which is fully described by its variance and mean. The variance, mean and transition matrix are then the properties that fully describes a state. The values of these properties are not given and it is impossible for a human to set them manually. Thus the properties

need to be trained using an automatic method. The most common algorithm for training HMMs is called the 'Baum-Welch' algorithm. The algorithm is complex and will not be described here, but it is explained in [21].

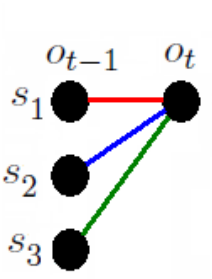
Training of HMMs requires a large database with speech to get good estimates of the state models. The models are highly dependent on relevant speech for the task and environment the HMMs are going to be used in for a good performance. For example Swedish HMMs will not perform well on Norwegian speech and vice versa. It should also be noted that it is a common choice to include long and short HMMs which represent long and short vowel length in Norwegian.

HMMs can be used to segment and label phonemes in an unknown word given a Hidden Markov Model. This is called the 'decoding problem': which state sequence is the most probable given the feature vector $o_1, o_2 ... o_T$. The problem can be solved using the Viterbi algorithm. The Viterbi algorithm finds the most likely state sequence to state number i at time t given the output (feature vector) $o_1, o_2 ... o_T$ and a HMM model by using the following formula:

$$P(s_t^i|o_t) = max_k \Big[P(s_{t-1}^k|o_{t-1}) * P(s_t^i|s_{t-1}^k) \Big] * P(o_t|s_t^i) \tag{2.6}$$

$$k = 1, 2 ... n \text{ where n = number of states} \tag{2.7}$$

The formula basically means that the probability of being at state i at time t can be found by finding the probabilities for being in the states at time t-1 multiplied with the transition probability to state i. The most likely probability is chosen and multiplied with the probability of being at state i given the output o_t. The formula is illustrated in figure 2.11.



$$= max \begin{bmatrix} P(s_1^{t-1}|o_{t-1}) * P(s_1^t|s_1^{t-1}) \\ P(s_2^{t-1}|o_{t-1}) * P(s_1^t|s_2^{t-1}) \\ P(s_3^{t-1}|o_{t-1}) * P(s_1^t|s_3^{t-1}) \end{bmatrix} * P(o_t|s_1^t)$$

Figure 2.11: The figure illustrates how the score at a given state can be calculated by using previously calculated scores. Only 3 states are included in the figure. There are no restrictions on the number of states in general.

By finding the state with highest probability at the end of the feature vector, $max_i[P(s_T^i|o_T)]$, the most likely state sequence can easily be found

using backtracking to find the best path. The backtracking can be done in the same way as for DTW: by looking at the path which gave the highest probability at the last state and going backwards from there. Given the state sequence, the phoneme sequence is known. The Viterbi algorithm is illustrated in figure 2.12.

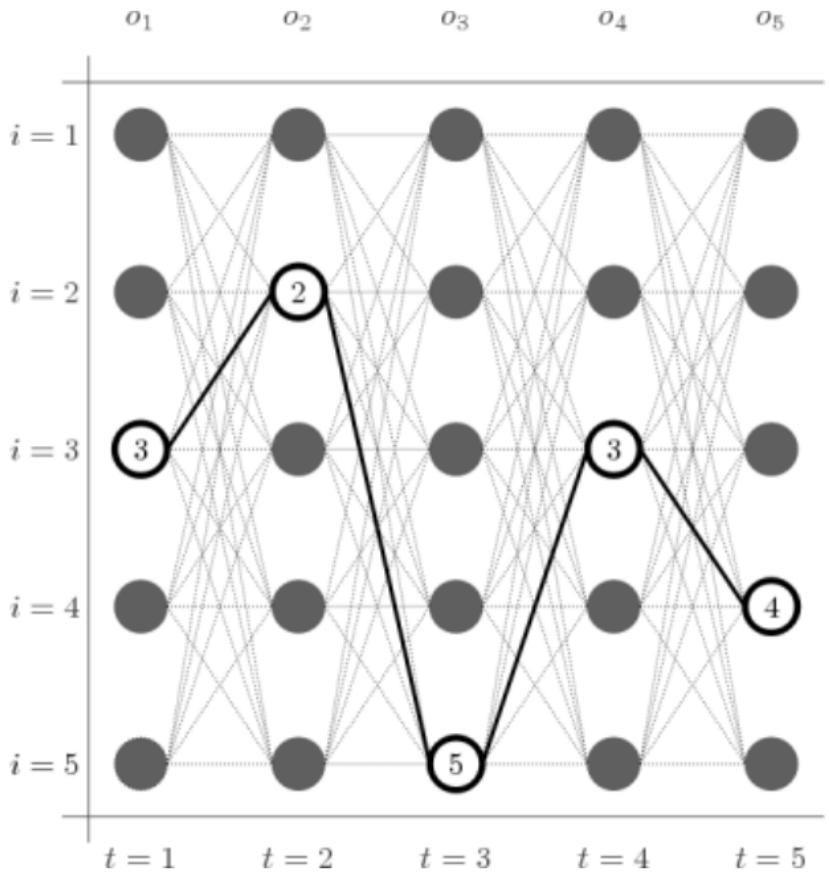


Figure 2.12: Illustration of Viterbi [10]. The states are aligned from top and down, while the given output (feature vector) is aligned from the left to the right. The Viterbi algorithm finds the path with highest likelihood, which is outlined in this figure.

If the phoneme sequence is known before decoding, forced alignment can be performed. Forced alignment implies that the problem of labeling and segmenting a sound file has been reduced to finding only the phoneme boundaries. Finding the phoneme boundaries are done in basically the same way as without forced alignment, but the Viterbi decoding is changed to only use the states according to the phoneme sequence.

One problem with calculating the likelihood with Viterbi in computers is that the likelihood will decrease fast to a very small number. The small number is a problem because a computer cannot represent the low number which occurs after some calculations. A solution to this problem is to take the logarithm (log) of the likelihood. The logarithm is an monotonic increasing function and does not change the end result, but will make the likelihood decrease slower. The logarithm of the likelihood is referred to as the log likelihood (LL).

2.4 Classification features

To classify a vowel length pronunciation a set of features are needed that gives the classifier information about which class the word belongs to. This section will present the classification features and state the reason for why they are included.

It is not known exactly how humans classify vowel length pronunciations or which class of features that are useful with automatic methods. Several features will be tested and it is up to the classifier to decide the weight in classification each feature should have.

A list of all classification features is shown below. The list specifies if a feature is calculated only with HMMs or DTW.

- Vowel and Consonant duration
- Normalized duration
- (HMM) Vowel, Consonant and Total Log Likelihood
- (HMM) Vowel, Consonant and Total Log Likelihood Ratio
- (DTW) Vowel, Consonant and Total Distortion
- Vowel and Consonant Uniform Energy
- (HMM) Vowel and Consonant State Energy

The *Vowel duration* and *Consonant duration* are found by using HMMs or DTW to segment the vowel length pronunciation. The Vowel duration is naturally assumed to be very important for classifying vowel length pronunciations. In [16] it is argued that the consonant following a long vowel is shorter, or at least not longer, than a consonant following a shorter vowel. Thus the consonant can help to further contrast between a long and short vowel pronunciation.

The thought behind *Normalized Duration* is to normalize the Vowel duration based on the speaking rate. Non-native speakers have a tendency to speak at a slower rate than natives, thus normalization of the Vowel duration could be important. The Consonant duration is also used because the Consonant duration should be shorter after a long vowel. This implies that the Normalized Duration can help further contrast between vowel length pronunciations.

The *Vowel and Consonant Log Likelihood* are found by using HMMs to segment the vowel length pronunciation, then using the Viterbi decoding to find the LL for the Vowel and Consonant. The LL is then normalized based on the Vowel and Consonant duration. The normalized LL for the Vowel and Consonant is called Vowel and Consonant Log Likelihood.

The *Total Log Likelihood* is the LL for the whole vowel length pronunciation, divided by the duration of the pronunciation. In general high LL implies that the phoneme(s) is acoustically close to the HMM, while low LL implies that the phonemes are not acoustically close.

Calculation of *Vowel and Consonant Log Likelihood Ratio* is explained in section 2.6. The same idea as for LL holds for them just that they are normalized based on other phonemes as well as the duration.

The *Vowel and Consonant Distortion* are found by using DTW to segment the vowel length pronunciation, then the accumulated distortion is calculated for the Vowel and Consonant. The accumulated distortion is further normalized by the duration, which gives the Vowel and Consonant Distortion.

The *Total Distortion* is the normalized accumulated distortion for the whole vowel length pronunciation. In general low Distortion implies that the phoneme(s) is acoustically close to the reference, while high Distortion implies that it is a large difference.

[14] shows that people listening to vowel length pronunciations used, among other things, spectral information to identify the vowel length. Thus long vowels should be pronounced acoustically different from short vowels. The Vowel LL is assumed useful for classification of vowel length pronunciations because the LL for a long vowel should have a higher likelihood with a long vowel HMM than with a short vowel HMM, and vice versa for short vowel. Consonant LL is less likely to differentiate between long and short vowel length, because the spectral information does not change as much for different vowel length pronunciations. The same principle holds for Distortion, just that the distortion should be lower for a long phoneme compared with a long reference than with a short reference.

The *Vowel Uniform Energy* consist of 3 energy values. The energy values are found by splitting the Vowel duration into 3 equally sized parts - start,

middle and end - then calculate the energy for each of these parts. Another way of splitting the energy is by using state information, which gives the *Vowel State Energy*. It is assumed that the HMMs use 3 states per phoneme. Thus the Vowel duration can be divided into 3 parts by using the time the Viterbi decoding used the different states. For example start energy value can be calculated based on the time the HMM went into the first state until the time the HMM left the first state.

The formula for calculating a energy value is shown in equation 2.8 where it is assumed that S contains the part of the sound signal which shall be calculated. The *Consonant Uniform Energy* and *Consonant State Energy* are calculated in the same way as for the Vowel.

$$\mathrm{EV} = \frac{1}{T}\sum_{i=1}^{T}(\mathrm{LE}_i) \tag{2.8}$$

$$\mathrm{LE}_i = 10 * log_{10}(\mathrm{E}_i)) \tag{2.9}$$

$$\mathrm{E}_i = \frac{1}{128}\sum_{k=1+i*32}^{128+i*32}(S_k^2) \tag{2.10}$$

LE = Log Energy, E = Energy, EV = Energy Value, S = Sound Signal. The energy calculation is based on a 16 kHz sound signal, thus the window size is 8 ms (128 samples) and the window step size is 2 ms (32 samples).

Vowel length pronunciations with short vowel length sound like they contain more stress on the Consonant compared to a pronunciation with long vowel length. The Vowel energy might also be what humans use to differentiate between vowel length. Thus it also included as a classification feature. The reason for calculating 3 energy values is to capture dynamics in the energy.

2.5 Automatic classification: Fisher Linear Discriminant

The purpose of a classifier is to decide if a given utterance with unknown vowel length pronunciation should be classified as short or long vowel length pronunciation. A classifier takes in classification features and calculates a score. The score together with a threshold is used to classify a vowel length pronunciation. This section describes how a classifier can be trained.

Linear discriminant analysis (LDA) or Fisher Linear Discriminant (FLD) are automatic methods for training a classifier given some labeled training

data. FLD and LDA find the linear combination of features which best separate different classes. The terms LDA and FLD will be used interchangeably. The theory behind FLD and how FLD is used for classification will be presented in this section. A more detailed description of FLD is found in [4] and [19].

The idea with FLD is to find projection to a line so that samples from different classes are well separated. An example is shown in figure 2.13. Finding a projection line can be done in different approaches, but a good solution is to base it on the mean and (co)variance of the classes that should be classified.

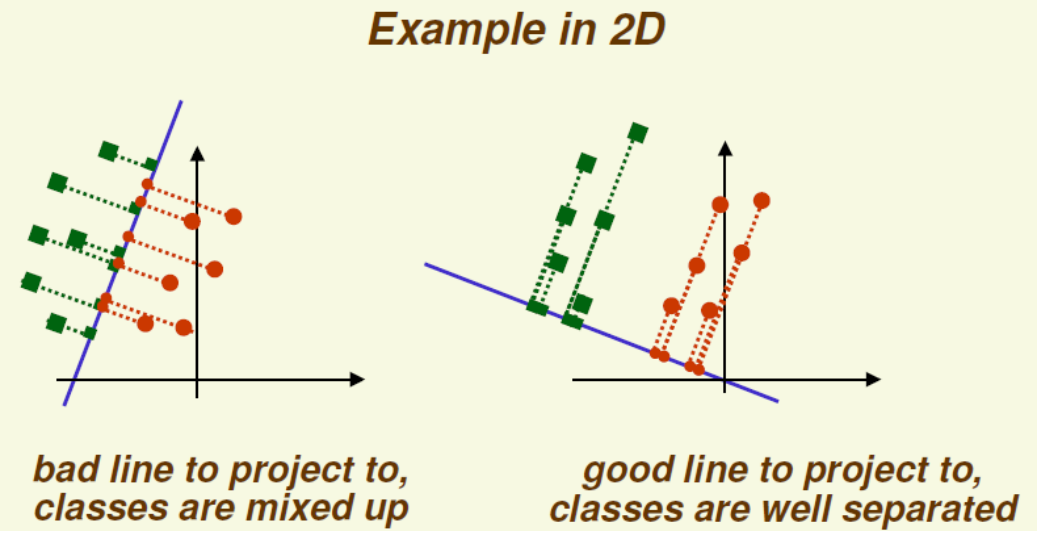


Figure 2.13: The idea is to find a projection to a line so that samples from different classes are well separated. The figure shows an example of a good and a bad line to project to. Figure taken from [4]

Assume that there are two classes with equal probability and that each of the classes has D dimensional samples $x_1, x_2, ..., x_n$. If v is a unit vector in the D dimensional space, then $y_k = v^t * x_k$ is the distance from the origin. Thus y_k is a projection of the sample x_k into a D-1 dimensional subspace.

The problem is to find the optimal line to project into. The optimal line (unit vector) can be found by using optimizing criterions that depends on the unit vector to maximize the separation after projection. Fisher Linear Discrimination uses two criterions which will be presented in the next paragraphs.

The higher difference in the means μ_i of the two classes the easier the separation can be performed. Thus the projected means $\hat{\mu}_i$ of the two classes

should be a good criterion for separating two classes if the two projected means are well separated. Therefore $|\hat{\mu}_1 - \hat{\mu}_2|$ should be large. How μ_i and $\hat{\mu}_i$ can be calculated is shown in equation (2.11) and (2.12).

$$\mu_i = \frac{1}{n} * \sum_{x_k \in \text{Class i}}^{n} x_k \tag{2.11}$$

$$\hat{\mu}_i = \frac{1}{n} * \sum_{x_k \in \text{Class i}}^{n} v^t * x_k \tag{2.12}$$

The variance of the two classes should preferably be small, because a small variance implies that the samples are clustered around the mean. Thus after projection the variance should be smaller. Instead of using the variance as a criterion, the 'scatter' is used. The scatter is basically the same as the variance, just no downscaling by n-1. The scatter and projected scatter can be calculated as shown in equation (2.13) and (2.14).

$$s_i^2 = \sum_{x_k \in \text{Class i}}^{n} (x_k - \mu_i)^2 \tag{2.13}$$

$$\hat{s}_i^2 = \sum_{y_k \in \text{Class i}}^{n} (y_k - \hat{\mu}_i)^2 \tag{2.14}$$

The Fisher Linear Discriminant chooses the line which maximizes the mean and minimizes the scatter by maximizing equation (2.15).

$$J(v) = \frac{(\hat{\mu}_1 - \hat{\mu}_2)^2}{\hat{s}_1^2 + \hat{s}_2^2} \tag{2.15}$$

The maximization is done by derivation of equation (2.15) with regards to the unit vector v and setting the resulting equation equal to zero. For a two class problem the optimal unit vector v is shown in equation (2.16). Further, an offset value b is calculated as shown in equation (2.20). The offset value

implies that the optimal line does not have to go through origo.

$$v = S_w^{-1}(\mu_1 - \mu_2) \tag{2.16}$$
$$S_w = S_1 + S_2 \tag{2.17}$$
$$S_1 = \sum_{x_k \in \text{Class 1}} (x_k - \mu_1)(x_k - \mu_1)^t \tag{2.18}$$
$$S_2 = \sum_{x_k \in \text{Class 2}} (x_k - \mu_2)(x_k - \mu_2)^t \tag{2.19}$$
$$b = -0.5 * (\hat{\mu_1} - \hat{\mu_2}) \tag{2.20}$$
$$\hat{\mu_1} = v^t * \mu_1 \tag{2.21}$$
$$\hat{\mu_2} = v^t * \mu_2 \tag{2.22}$$

Equation (2.16) assumes that the inverse exists for S_w. If the inverse does not exist the eigenvalues must be found, which will not be discussed here.

To classify a test sample x using FLD the projected value $y = v^t * x + b$ is calculated. If $\hat{\mu_1}$ was larger than $\hat{\mu_2}$ the test sample is assumed to belong to class 1 for a positive projected value, and vice versa for a negative projected value.

The unit vector v, together with b and the threshold that decides which class a test sample belongs to, creates a classifier. The projected value will be referred to as LDA score. The more data used to train the LDA model the better the model works for new data, assuming that the new data follows the same pattern as the training data.

As with any automatic method that needs training the number of training data versus dimensionality is important. In theory the more dimensions (features) used to classify data the better result is expected. However increasing the number of dimensions while keeping the training data low will result in specialized models for the training data. A model should be general in order to fit new data better. Thus it is not possible to increase the number of dimensions forever and expect to get better results without increasing the number of training data. This is called the 'curse of dimensionality' and is discussed further in [21].

2.6 Confidence measures

A confidence measure gives an indication of how confident we are that the unit to which it has been applied (e.g. a phrase, word, phone) is correct [1]. This section presents a confidence measure based on segmentation and LDA score.

The HMM and DTW segmentation of an utterance forms the basis of the classification features. Thus if the segmentation is unreliable the classification features are most likely unreliable. If forced alignment has been used to segment an utterance there is no information whether the phonemes in the utterance and the HMMs used to segment it correspond at all. The utterance might not contain the phonemes assumed to be in it. Pronunciations with wrong phonemes are likely to result in unreliable segmentation. It is also a point in itself to detect if a pronunciation is wrong for CAPT systems, because a L2 learner should be taught to pronounce words correctly.

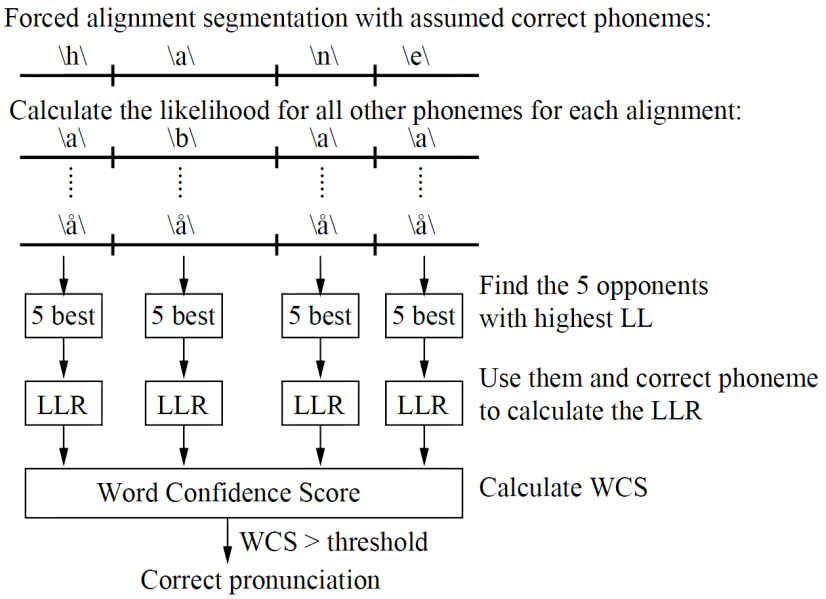


Figure 2.14: The figure illustrates how the Word Confidence Score is calculated by using the forced alignment segmentation. The LLR is calculated for each of the phonemes, then summed up to the Word Confidence Score.

The confidence measure for segmentation is based on finding out how likely the phonemes (and corresponding HMMs) used to segment a word are compared to other phonemes. To find out how much a phoneme can be trusted the log likelihood ratio (LLR) is calculated per phoneme. This is done by finding the likelihood of the assumed 'correct' phoneme versus the average likelihood of a subset of the rival phonemes[1]. If the LLR is low the assumed 'correct' phoneme is less likely to have been pronounced correctly or the segmentation has displaced the 'correct' phoneme where another phoneme

[1]A rival phoneme is any of the other phonemes except for the assumed correct phoneme.

is. The calculation of LLR is shown in equation (2.23).

To check whether the whole pronunciation contains errors the LLR is summed up for all of the phonemes in a way that emphasizes low log likelihood ratio scores. The summed up LLR is called the Word Confidence Score (WCS). If the WCS is below a set threshold then the pronunciation should ideally contain phoneme error(s), while above the threshold the pronunciation should be correct with regards to the assumed phonemes in the word. Equation (2.24) shows how the WCS is calculated. The confidence measure is illustrated in figure 2.14 where there are assumed to be 4 phonemes in the word and an average of 5 phoneme rivals are used to calculated the LLR.

$$\text{Ph.LLR} = \text{LL}(\text{Ph}_x) - \frac{1}{N} \sum_{k \in \text{subset}}^{N} \text{LL}(\text{Ph}_k) \tag{2.23}$$

where the subset is defined as being the N rival phonemes with highest likelihood. LL = log likelihood, Ph = phoneme, LLR = log likelihood ratio.

$$\text{WCS} = \log((\frac{\sum_{k=1}^{L} e^{(-\text{Ph.LLR})}}{L})^{-1}) \tag{2.24}$$

where L is the number of phonemes in the word. Silence/noise at he start and end of an utterance is not included because only phoneme errors are of interest.

Another confidence measure can be found by using the LDA score from the FLD trained classifier. A LDA score close to zero will logically be close to class 1 and class 2. By assigning an area around zero as 'unsure area' most of the classification errors are found while most of the correct classifications are outside the window. This is illustrated in figure 2.15.

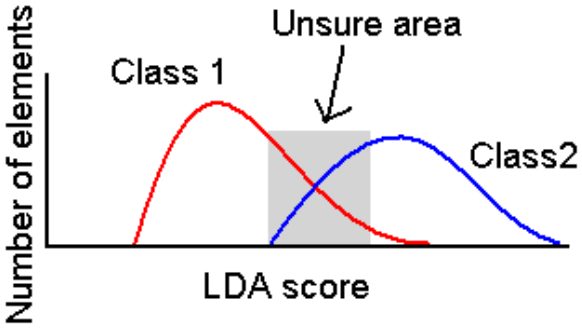


Figure 2.15: This figure shows an example of the score from a LDA. The score can be used as a confidence measure by assigning scores around 0 (the score between the two classes) as 'unsure'.

Chapter 3

Recording and preparing a database

This chapter describes how a database of both native and non-native speakers were recorded and prepared. Section 3.1 explains the reasoning behind the choice of the words which were recorded while section 3.2 presents how the database was recorded. Further, section 3.3 presents how the sound files in the database was prepared. Finally, section 3.4 explains how the labeling of the database was done.

3.1 Word selection

Before the recording of the database could be performed the words to be recorded had to be chosen. This section describes the reasoning behind the word selection.

It was chosen to record isolated speech[1]. Isolated speech reduces the time taken to record the words and the complexity in determining the phoneme boundaries compared to continuous speech. Further, all of the recorded words were recorded in minimal pairs with respect to vowel length. Minimal pairs make it easier for the L2 learner to grasp the pronunciation difference between short and long vowel length.

The vowel length pronunciation classification method involves segmentation of the phonemes in the utterances. Automatic segmentation is more reliable if the sound file with the utterance contains the expected phonemes[2]

[1]Isolated speech implies that the sounds files with the spoken words contain either one word or words with clear pauses between them.

[2]In a language, a phoneme is the smallest posited structural unit that distinguishes meaning[2]

pronunciation. The phoneme 'r' is difficult to pronounce for foreigners and difficult to segment using automatic systems due to large variations in pronunciation. Thus the phoneme was avoided in most of the minimal pairs. However some words still include 'r', because it was difficult to find alternative words.

18 regular words (9 minimal pairs) were selected. The words were selected to include all of the 9 vowels in the Norwegian language and based on the mentioned criteria.

In addition to regular words, a set of non-sense words were included where only the vowel changes while the consonants are always the same: k-vowel-t-e (long vowel word) and k-vowel-t-t-e (short vowel word). It was assumed that automatic segmentation methods can segment the non-sense words more reliably than many other words due to large difference between the phonemes in the words. It was also assumed that the consonant phonemes 'k' and 't' are easier to pronounce for non-natives than for example phonemes like 'r'. Therefore the pronunciations and segmentations of the non-sense words should be more reliable than for many other words.

The non-sense words are possible to pronounce in Norwegian and some of them are actual words. The list with all of the chosen words is shown in appendix A.2.

3.2 Recording a database

This section presents details about how the utterance database was recorded.

Before the start of the thesis a teacher voice was recorded. The teacher voice database contains all the different words clearly articulated by a professional speaker. The recording of the teacher voice was done in a professional studio with the same microphone and computer as the one used in the test database recording. The reason for recording a teacher voice is that in a CAPT system the user will most likely not know how to pronounce a word by just looking at the letters, neither does the non-native test speakers.

The teacher voice was also used as a reference for DTW segmentation. DTW needs a segmented reference to segment other words, as explained in section 2.2. The teacher voice recordings were segmented manually by a professional speech scientist.

As a part of this work a recording interface was developed. This was done to make the recording procedure as easy as possible for the speaker being recorded. The speaker was only expected to click on a button. The first click on the button played the word the speaker should pronounce and the second click started the recording. The recording lasted for 3 seconds, then a

new word was presented and the process was repeated until every word was recorded. The user interface is shown in figure 3.1.

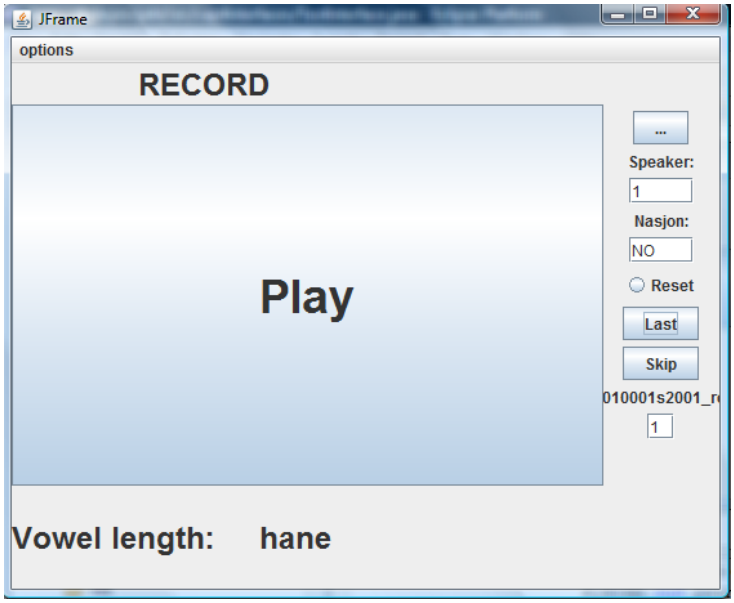


Figure 3.1: The recording interface used in this work. The user is only supposed to click on the button marked 'Play' which changes into 'Record', then back to 'Play' again, and so on. The rest of the buttons are for the recording supervisor.

The recorded speakers were chosen based on different factors. One such factor was the speaker's background. The background can be important for the realization of pronunciations. Thus to minimize variations the speakers were gathered based on their mother tongue and on how many speakers could be obtained for a specific mother tongue. Another factor, in which the speakers was chosen by, was that the mother tongue should not be closely related to Norwegian (for example Swedish and Danish).

The recording of the database with natives and non-natives was done in a way to reduce variations and noise. All of the speakers used the same equipment, were recorded in the same room, and the speakers received the same knowledge about the motivation for the recordings. The two first pages of the document shown in appendix A.4 was sent by e-mail to the speakers before start. The pages explained the motivation and what they were going to do. When the speakers arrived at the recording room, the two last pages of the document were presented. The pages showed the words they were going to pronounce and the meaning in English. They also got a short briefing on what was expected of them.

For each speaker a test recording was always performed. The test recording was done to check that the recording level did not exceed maximum and to let the speaker get used to the equipment. A recording supervisor was always present in the room to explain what the speakers should do and to make sure the recordings were acceptable. The recording was performed in the following manner:

- Repeat the list with tone words three times (not used in this work)
- Repeat the list with vowel length words three times
- Repeat the list with non-sense vowel length words three times

In total there were recorded 1404 utterances with vowel length pronunciations. For more information about the recording conditions and how many speakers were recorded see appendix A.

All of the non-natives recorded in this work had been living in Norway for some time. Length of residency and other factors like their mother tongue can affect the pronunciation of the words and thus the result from automatic classification. Appendix A.3 shows what was assumed to be important information about the speakers. In case the number of classification errors per speaker differed much, the information could be helpful to determine why.

3.3 Preparing the database

This section explains how the utterance database was prepared by processing the sound files.

The recording of the database was done with a sampling rate of 48 kHz. However the microphone used in the recording was of limited quality, therefore sound above a frequency of 8 kHz was noisy. The recorded sound files included a lot of silence due to the 3 second recording window. Silence and noise are unwanted parts of the sound files, because they include no information about the pronounced words. Figure 3.2 shows how the database was processed to remove some of the silence and noise.

The following list explains the steps shown in figure 3.2 and why they were performed:

- Amplitude scaling to a maximum amplitude of 0.6: If the absolute amplitude is close to 1, a filter operation (which is performed when downsampling) can cause saturation in the sound files. A solution is to scale down the amplitude before filtering. Amplitude scaling also normalizes the energy across speakers.

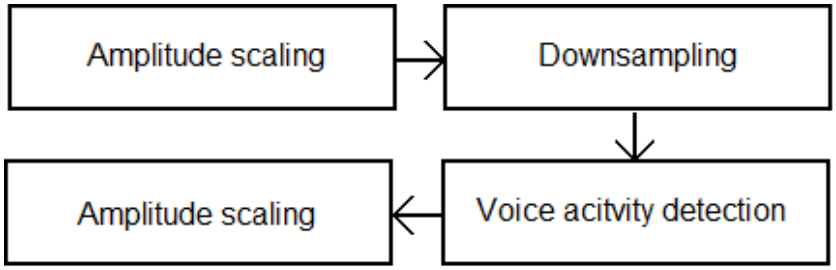


Figure 3.2: This figure shows the order in which the database was prepared by removing noise and silence.

- Downsampling (and filtering): Downsampling was performed to remove noisy frequency parts of the sound signal while keeping the parts that are useful. The noisy frequency parts largely stem from the microphone, which is not accurate at high frequencies. The sound files were downsampled to a sampling rate of 16 kHz. The reason for keeping a rather high sampling rate was because of the detailed acoustic analysis that was going to be performed.

- Voice activity detection (VAD): VAD was done with an algorithm from [15] to remove silence at the start and end of the sound files. One reason for including a VAD algorithm is that silence has no relevant information and may only confuse the segmentation algorithms.

- Amplitude scaling to a maximum amplitude of 0.6: The preceding operations changed the amplitude, thus another amplitude scaling was performed to ensure that the maximum absolute amplitude was the same for all files.

3.4 Pronunciation evaluation by a judge

The utterance database had been recorded and prepared, but there were no information what vowel length the non-native speakers had pronounced. The part of the database with non-native speakers had to be labeled. The labels were going to tell whether the vowel length pronunciations were short vowel or long vowel length, and whether the utterances contained pronunciation errors.

To ensure that the judge would make unbiased labels, a decision interface was made which played the recorded words. When using the interface the judge would only know which minimal pair he/she was listening to and

nothing else. The decisions of the judge were used to label the utterances. Based on what the judge perceived, a decision was made between four possible labels:

- Sure short vowel
- Unsure short vowel
- Unsure long vowel
- Sure long vowel

In addition, the judge could label whether there were any pronunciation errors in the utterance. The judge was instructed to label an utterance as pronunciation error if a Norwegian would have problems understanding the pronunciation, where the error in the pronunciation was not related to the vowel length.

The utterances in the database were presented after what minimal pair they belonged to. Decisions were first made on word pair 1, then word pair 2 et cetera. All of the utterances within a minimal pair were randomized, giving the judge no clue as to which vowel length the non-natives had tried to pronounce. The decision interface is shown in 3.3

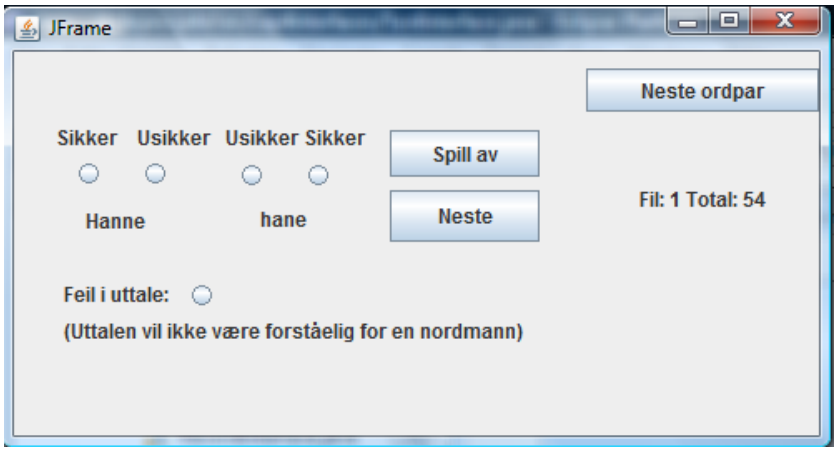


Figure 3.3: The decision interface used in this work. The judge must play the word at least one time and choose one of the 4 label options before clicking to get another word.

After the decision interface was implemented a native Norwegian labeled the non-native database. The judge labeled each utterance 2 times. The minimal pairs switched places in the interface for the second label. With two

labels per utterance it was possible to inspect how consistent the judge was labeling the utterances.

The conflicts and non-conflicts are shown in table 3.1. A conflict is defined as an utterance with two different labels while a non-conflict is defined as an utterance with the same label.

-	Sure SV	Unsure SV	Unsure LV	Sure LV
Sure SV	357	34	2	6
Unsure SV	-	8	16	7
Unsure LV	-	-	9	24
Sure LV	-	-	-	509

Table 3.1: The labels made by the judge on the non-native database. If the 2 labels were consistent for an utterance it is shown in the diagonal, while the rest of the table shows the conflicting labels. SV = Short Vowel, LV = Long Vowel

As can be seen from table 3.1 there are $2 + 6 + 16 + 7 = 31$ labels that change from unsure or sure short vowel to unsure or sure long vowel. With 972 non-native utterances, the conflicting labels result in a long to short vowel conflict rate of 3.1%. The conflict rate should be low because any automatic classification test of the database using the judge as solution is in theory lower bounded by the conflict rate. The classifier cannot be more correct than the solution.

The number of conflicting labels on pronunciation errors was 58. The first decision labeled 149 pronunciation errors, which results in 15.3% of the utterances containing pronunciation errors.

The decisions were used to label the recorded utterances and used instead of what the recorded speakers were asked to pronounce. Where there were conflicts between the labels the first decision was used. The number of utterances belonging to the different labels is shown below:

- Sure short vowel length: 374 (38.4% of total)
- Unsure short vowel length: 44 (4.5% of total)
- Unsure long vowel length: 25 (2.6% of total)
- Sure long vowel length: 529 (54.4% of total)
- Total number utterances: 972

If only looking at the vowel length the distribution is:

- Short vowel length: 418 (43% of total utterances)
- Long vowel length: 554 (57% of total utterances)
- Number of utterances with short vowel length and pronunciation error: 48
- Number of utterances with long vowel length and pronunciation error: 101

It is important to remember that the labels are one person's perception of the utterance and that the labels would most likely be different for another judge. The labeling results even showed that some of the labels were different for the same judge. [20] discusses human labeling consistency and shows that there is a significant variability among different judges.

Ideally many different judges should have labeled the utterances. Then the average decision on each utterance could have been used instead of one person's labels. Using an average of many people would have given more 'objective' labels.

Chapter 4

Method for classification of vowel length

Figure 4.1 shows how the classification of vowel length pronunciations was done. This chapter explains the specific choices made at each step and how the proposed method was evaluated.

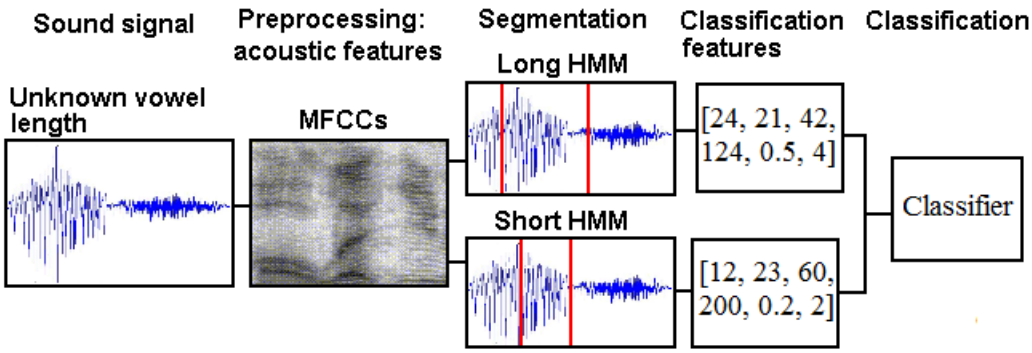


Figure 4.1: Classification of vowel length pronunciations

Section 4.1 presents the chosen MFCC parameterization, while section 4.2 presents the chosen way of doing segmentation. Section 4.3 explains how the classification features and classifier were used. Section 4.3 also presents two different methods of training the classifier. Finally, section 4.4 explains how the confidence measures were used, calculated and evaluated.

4.1 Preprocessing: acoustic features

This section explains how the preprocessing was done in this work.

The acoustic feature (MFCCs) calculation was done with the Hidden Markov Model Toolkit (HTK). For a detailed description of the toolkit see [5]. For each sound file, or equivalently for each utterance, there were calculated 39 MFCCs (including energy, 1st-order delta and 2nd-order delta coefficients) per window. The 39 MFCCs will be referred to as Dynamic MFCCs. The chosen window size was 15 ms and the step size was 5 ms. For more details about the parameterization with HTK see appendix B.3.

DTW segmentation was also done with another MFCCs parameterization than using 39 MFCCs. It was also tested using a parameterization derived in [15]. This parameterization uses only 13 non-delta coefficients which were normalized and weighed. The normalization and weighing emphasized lower frequency components and the energy. For more about the normalization see appendix B.1. The normalized and weighed MFCCs will be referred to as Normalized MFCCs.

Thus the different MFCCs options were:

1) (HMM and DTW) Dynamic MFCCs: 39 coefficients which includes energy and delta coefficients

2) (DTW) Normalized MFCCs: 13 coefficients (including energy) which were normalized and weighed

4.2 Segmentation

This section explains how the HMMs and DTW were used in this work to segment utterances. This section also presents some information about the training of the HMMs.

The HMMs had to be trained before they were used. Before the start of the thesis HMMs were trained based on 2 different databases: a database with isolated speech[1] and a database with continuous speech. The HMMs trained on isolated speech will be referred to as *isolated HMMs* while the HMMs trained on continuous speech will be referred to as *continuous HMMs*. The training databases used to train the HMMs were not recorded in this work, but taken from another project. The trained HMMs include one model for each phoneme in the Norwegian language. Key information about the HMMs used in this work is listed in appendix B.2.

When doing segmentation it is assumed that the CAPT system knows which minimum pair the user of the system is exercising on, but not which

[1]Isolated speech is sound files with either one word or phoneme in them, or words with clear pauses between them.

vowel length the user has pronounced. With that assumption all of the phonemes in the utterance were known except for the phoneme representing the Vowel.

Because the vowel length was unknown two segmentations were performed per utterance: the first called short segmentation and the second called long segmentation. When doing segmentation with HMMs, a short HMM segmentation implies that a HMM representing a short vowel length was used, while a long HMM segmentation implies that a HMM representing a long vowel length was used. For DTW, a reference pronouncing a long or short vowel length was used instead of HMMs, but the same principle holds.

Segmentations were done with both HMMs and DTW to test which segmentation method results in the lowest number of classification errors. Forced alignment was the chosen method for HMM segmentations while the Restricted DTW was used to do the DTW segmentations. The HTK was used to do segmentation with HMMs.

4.3 Classification features and classification

This section explains how the segmented utterances were used to make classification features and how the classification features were used to classify vowel length pronunciations.

The short and long segmentations were used to make classification feature vectors. For each of the two segmentations a classification feature vector was calculated, which were combined before classification as shown below:

$$\begin{aligned}
\textbf{MainClassi.Vec.} &= (\text{Short Seg. Classi. Vec.}, \text{Long Seg. Classi. Vec.}) \\
\textbf{ShortSeg.Classi.Vec.} &= (\text{feature1}_x, \text{feature2}_x, \ldots, \text{featureN}_x) \\
\textbf{LongSeg.Classi.Vec.} &= (\text{feature1}_y, \text{feature2}_y, \ldots, \text{featureN}_y)
\end{aligned}$$

All of the classification features were presented in section 2.4. In total 21 HMM classification features were calculated per segmentation. This implies 42 classification features were used in the classifier when segmenting with HMMs. For DTW there were 12 classification features per segmentation, thus the classifier used in total 24 classification features when segmenting with DTW.

To do classification of vowel length pronunciations a classifier was trained to classify vowel length pronunciations into either long or short vowel length. The classifier was trained using the Fisher Linear Discriminant principle. The words were labeled into 4 vowel length categories by a native as described

in section 3.4: Sure short vowel, unsure short vowel, unsure long vowel, sure long vowel. When training and testing the classifier, the sure and unsure short vowel length classes were defined as short vowel length, while the sure and unsure long vowel length classes were defined as long vowel length.

The classifier thresholds were:

LDA score < 0 == short vowel length

LDA score > 0 == long vowel length

Two different methods of training the classifier were used:

1) Norwegian trained classifier

2) Rotation trained classifier

The first classification method was to train the classifier on all of the Norwegian speakers and use the classifier to classify non-native vowel length pronunciations. The other classification method was to train the classifier on a selected part of the non-native speakers, which is called rotation training. Rotation training implies that the classifier is trained on a set of speakers where one speaker is removed before training. The trained classifier is then used to classify the vowel length pronunciations belonging to the speaker left out. The speaker being left out changes until every speaker's vowel length pronunciations has been classified. The expected number of classification errors, given a classifier trained on all of the speakers, can then be calculated by finding the average number of classification errors with the rotation classification method. The rotation training method is useful when the number of training data is low.

Training of the classifier with FLD assumes that there is an equal probability of each class. As shown in section 3.4, there were 43% short vowel pronunciations and 57% long vowel pronunciations recorded for the non-native speakers. This made the equal probability assumption false when training with non-native speakers. The small difference was assumed to have little impact on the results and was therefore not inspected further.

Given a classifier and a test database the percent chance of doing classification errors, called error rate, can be calculated to evaluate the classifier. A classification error is defined as when a vowel length is classified as a long vowel length pronunciation by the classifier, while the judge has labeled the utterance as a short vowel pronunciation, and vice versa. The total number of classification errors was found by classifying all of the vowel length pronunciations in the selected part of the database that is of interest. The error rate was calculated by equation (4.2).

$$\text{P(classification error)} = \frac{\text{Number of classification errors}}{\text{Number of test utterances}} \tag{4.1}$$

$$\text{ErrorRate} = \text{P(classification error)} * 100 \tag{4.2}$$

4.4 Confidence

The confidence measures presented in section 2.6 were in this thesis used to detect utterances with high potential of being classified wrong and therefore discard them. What to do with the discarded utterances in a real CAPT system is educational psychology and was not considered in this report.

The discarded utterances were split into correctly and wrongly discarded utterances. Correctly discarded utterances were utterances with wrongly classified vowel length, pronunciation errors or utterances labeled as unsure vowel length by the judge. Wrongly discarded utterances are the rest of the utterances. Pronunciation errors should be detected and the L2 learner given feedback that the pronunciation is wrong. Thus discarding them before vowel length classification is reasonable. The same is true for unsure vowel length words, because if a native judge is not sure which word was pronounced then something in the vowel length pronunciation is assumed to be wrong.

To calculate the LLR discussed in section 2.6 the 5 best opponent phonemes was used. All of the recorded words in the database had 4 phonemes, thus the WCS was calculated based on the corresponding 4 LLRs.

To evaluate the different confidence measures a score was calculated called Confidence Evaluation Score (CES). CES is the percent chance of correctly discarding utterances. A score close to 100% means that the confidence measure discards nearly only utterances which should be discarded. The equation for calculating CSE is shown in equation (4.3).

$$\text{CES} = \frac{\text{Number of correctly discarded utterances}}{\text{Number of discarded utterances}} \tag{4.3}$$

Chapter 5

Results and discussion

This chapter presents the results and a discussion of them. The classification tests were done on the non-native speakers unless otherwise specified. A test utterance is defined as an utterance which was used to test the classifier.

The chapter is structured as follows:

- Section 5.1: Compares the HMM segmentations with the manual segmentation of the teacher voice. This section also shows the manually segmented Vowel and Consonant duration.
- Section 5.2: Presents the classification results using HMM and DTW segmentation. This section also finds the segmentation option that result in least amount of classification errors.
- Section 5.3: Inspects the classification errors with regard to speaker and word type. This section also discusses the reasons for the classification errors.
- Section 5.4: Inspects the classification errors with regard to the features and finds a subset of classification features that result in least amount of classification errors.
- Section 5.5: The confidence measures are used to discard words which are thought to be problematic for the classifier and by that improve the classification result.

5.1 Segmentation results: teacher voice

This section compares the automatic segmentation results using HMMs with the teacher voice and discusses the teacher voice's manually segmented Vowel and Consonant duration.

The teacher voice was the only manually segmented speech in the recorded databases. The speech was used to see how close the isolated and continuous HMMs segmented the speech compared to the manual segmentation of the teacher voice. Table 5.1 shows the comparison. In the table correct HMMs implies that long vowel HMMs were used for utterances with long vowel length and short vowel HMMs were used for utterances with short vowel length. Long/short vowel (LV/SV) HMMs implies that only the specified HMM was used for all of the utterances.

Time (ms)	<= 5	<= 10	<= 15	<= 20	<= 40	<= 80
Cont. Correct HMMs	12%	27%	38%	56%	78%	97%
Cont. LV HMMs	12%	25%	38%	55%	78%	97%
Cont. SV HMMs	12%	24%	36%	53%	76%	95%
Isol. Correct HMMs	19%	48%	61%	71%	90%	97%
Isol. LV HMMs	20%	50%	62%	75%	91%	97%
Isol. SV HMMs	18%	48%	58%	66%	89%	97%

Table 5.1: The table shows how many boundaries, in percent, the HMMs segment below X ms in absolute distance from the manual segmentation of the teacher voice. Total number of boundaries was 180. Cont. = continuous, LV = long vowel, SV = short vowel. A higher percent implies that the segmentation is closer to the manual segmentation.

Table 5.1 shows that segmentation with isolated HMMs results in a lower absolute distance from the manual segmentation compared with the continuous models. This is consistent with the theory about HMMs because the teacher voice was spoken as isolated speech, thus there were a match between trained HMMs and the speech.

The more surprising result in table 5.1 is that using only segmentation with long vowel HMMs results in a consistently better segmentation than only using segmentation with short vowel HMMs. Better segmentation implies that the segmentation is closer to the manual segmentation. When segmenting with the isolated HMMs, the long HMMs result in better segmentation than only using correct HMMs. This indicates that the long vowel HMMs are better trained than the short vowel HMMs or that the teacher voice is acoustically closer to the long HMMs. However, the low number of boundaries and small difference between correct and long vowel HMMs implies that the result is not significant.

Figure 5.1 shows the Vowel and Consonant duration of the teacher voice using manual segmentation. As can be seen by the figure, the Vowel duration clearly contrast between short and long vowel length pronunciations, while

the Consonant duration is distributed nearly randomly.

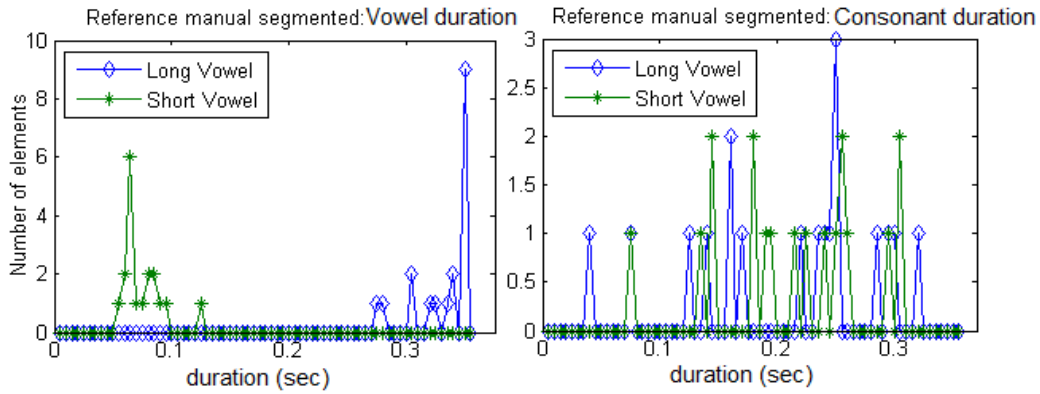


Figure 5.1: Manually segmented teacher voice: Vowel and Consonant duration.

The teacher voice was clearly articulated, thus it comes as no surprise that the Vowel duration contrast between the vowel length. According to [16], the Consonant duration should be shorter after a long vowel length and vice versa. As can be seen in figure 5.1 the teacher voice's Consonant duration does not seem to hold any information at all about which vowel length was pronounced.

One explanation for the more or less random Consonant duration could be that the Consonant duration is not that important for vowel length pronunciations after all. Because the Consonant duration does not contrast between vowel length with manual segmentation, it cannot be expected that the Consonant duration contrast with automatic segmentation either. However, there are few Consonant durations in the figure, thus in average on a larger test database the Consonant duration might still contrast between vowel length pronunciations.

5.2 Classification results

This section presents and discusses the classification results using different segmentation and classifier training methods.

Table 5.2 presents the classification results with non-native speakers, using different segmentation options. The table shows that the lowest classification error rate was 7.7%, which was calculated based on segmentation with continuous HMMs. The second lowest error rate was with Restricted DTW using Normalized MFCCs, which resulted in an error rate of 8.6%.

Segmentation method	Number of errors	Error rate
Continuous HMM	75	7.7%
Isolated HMM	119	12.2%
Rest. DTW (Dyna. MFCCs)	119	12.2%
Rest. DTW (Norm. MFCCs)	84	8.6%

Table 5.2: Number of classification errors made with different segmentation methods on non-native speakers. The total number of test utterances was 972 and the classifier was trained on the Norwegian speakers. Rest. = Restricted, Norm. = Normalized, Dyna. = Dynamic

The results in section 5.1 indicate that segmentation with isolated HMMs is better than with continuous HMMs. Based on those results it was expected that segmentation with isolated HMMs would result in a lower classification error rate than with continuous HMMs.

Table 5.2 shows that segmentation with continuous HMMs results in a lower error rate than using isolated HMMs, which is contrary to the expectation. One reason the results did not follow the expectancy might be because the test speakers were non-native while the HMMs were trained on Norwegian speakers. Some of the problems with speech recognition and non-native speech are described in [13]. The paper explains that HMMs trained on native speech cannot be used reliably for speech recognition on non-native speakers because, among other things, they speak with a non-native accent. The accent results in a mismatch between the trained HMMs and the speech.

In this thesis the HMMs were trained on native speakers and used to segment non-native speakers, and thus also has this mismatch. That the isolated HMMs segmented native speech (the teacher voice) better than the continuous HMMs does not necessarily imply that the isolated HMMs segment non-native speech better. A low classification error rate does not need to correspond with a segmentation that is close to a manual segmentation either. The purpose of the segmentation is to get classification features which contrast between long and short vowel length pronunciations. Thus an automatic segmentation can in theory be better than a manual segmentation for vowel length classification. However, this does not explain why the classification results varied that much between the two HMM segmentation options and no good explanation was found.

Table 5.2 also shows that the classification error rate is lower when using Restricted DTW with Normalized MFCCs than Restricted DTW with

Dynamic MFCCs to segment the test utterances. This is consistent with the results from [15] where native speech was classified using similar MFCC options. The reason for the lower classification error rate with Normalized MFCCs is most likely because emphasizing the first MFCCs increases the coefficients that are estimated better or are more important for distinguishing between the phonemes. This may have resulted in segmentations with a more consistent difference between short and long vowel lengths when using DTW with Normalized MFCCs to segment.

Finally, table 5.2 shows that the classification error rate with segmentation based on Restricted DTW using Normalized MFCCs is very close to the error rate with continuous HMMs. The difference is only 0.8 percentage points.

In general HMMs can handle greater speaker variations than DTW due to the nature of general models which are trained on many speakers versus just using one reference speaker (the teacher voice). Thus using HMMs to segment should result in a more reliable segmentation. However, all of the speakers listened to the teacher voice before they were recorded. Thus they were influenced to speak in the same manner as the teacher voice. Since the teacher voice was used as a DTW reference, this may have been an advantage. As explained earlier, the purpose is only to get a segmentation with a clear contrast between short and long vowel length pronunciations. This may have made the advantages with HMMs less apparent than they would have been if only the segmentation was compared.

Table 5.3 shows the classification error rate when using rotation versus Norwegian trained classifier. The HMM and DTW options with lowest classification error rate in table 5.2 were used to do segmentation, which as discussed above were continuous HMMs and normalized DTW. The table also shows how many classification errors were made when classifying native (Norwegian) speakers.

Table 5.3 shows that using rotation classifier increases the classification error rate slightly for HMM. Using a Norwegian trained classifier to classify non-native speakers results in a mismatch between the classifier and test data. Using the non-natives as training data results in a match between classifier and test speakers, but a mismatch between the HMMs and the non-natives. Training of the rotation classifier with non-native speakers included vowel length pronunciations which were judged as unsure in section 3.4. Many of the non-native utterances were labeled as pronunciation errors. Thus the training data for the non-native rotation classifier included a lot of problematic utterances which were likely to result in inconsistent classification features.

The inconsistent features are probably the reason for the lower error

classifier	HMM		DTW		Test database
	Number of errors	Error rate	Number of errors	Error rate	
Norwegian	75	7.7%	84	8.6%	Non-natives
Rotation	78	8.0%	84	8.6%	Non-natives
Rotation	4	0.9%	7	1.6%	Natives

Table 5.3: This table shows the classification results when using Norwegian trained classifier and classification result using rotation trained classifier. The rotation training and testing was done with regard to the speaker's mother tongue. The number of test utterances for non-native speakers were 972. For natives there were 432 test utterances.

rate when using rotation trained classifier. However, the percentage point difference between the two classifier training methods is insignificant.

An advantage with the Norwegian classifier is that the classification result calculated based on the classifier can be expected to be improved easily. A classification feature can be inconsistent between natives and non-natives, because non-natives might use other clues in the pronunciation to pronounce vowel length. An inconsistent feature should result in more classification errors than if it was removed. This implies that for the Norwegian trained classifier removing a classification feature can improve the classification result.

Table 5.3 also shows the classification error rate when natives are classified. The error rate is significantly lower than for the non-native speakers. When classifying the natives there is a match between the HMMs used for segmentation and the speech. There are no pronunciation errors and no non-native accent to make the segmentation difficult with automatic methods. The classification error rate on native speech can therefore intuitively be seen as the lower bound for the classifier.

In the rest of the thesis only the segmentation and classifier with lowest classification error rate was used. The segmentation method with lowest error rate was segmentation with continuous HMMs and the classifier with lowest error rate was the Norwegian trained classifier.

5.3 Inspection of classification errors

This section inspects the classification errors with regard to the speaker and word spoken. This section also discusses the reasons for the classification errors.

The segmentation is a crucial part in the classification method, everything after depends on the segmentation. For that reason, a manual inspection was done of the utterances to check if it was likely that the segmentations were the main cause for the classification errors.

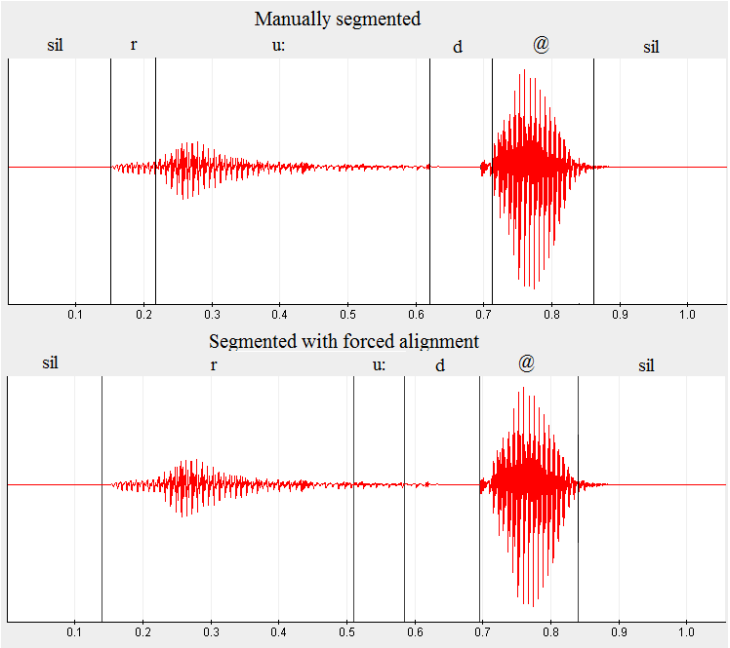


Figure 5.2: Example of one of the utterances with significant segmentation error when doing forced alignment segmentation. The figure also shows the manually segmented utterance.

Due to time restrictions the manual inspection of the utterances was only done on the vowel length pronunciations which were classified wrong. The manual inspection counted how many of the utterances were subject to significant segmentation errors. Figure 5.2 illustrates an example of a significant segmentation error and the manual segmentation of the same utterance. The large difference between the manual segmentation and automatic segmentation indicates a significant segmentation error.

The manual inspection of the utterances with regard to segmentation found that approximately half of the utterances were subject to significant segmentation errors. Based on this inspection it can be assumed that segmentation errors are the reason for most of the classification errors, but that some of the classification errors also are due to other reasons. One of the reasons for non-segmentation errors can be that the classification features do not capture all of the necessary information about the vowel

length pronunciation to classify them correctly. Another reason can be that the linear classifier is too simple. Another more elaborate classifier might decrease the classification error rate.

There are many reasons for the segmentation errors. As discussed earlier, the non-native accent is likely to be the reason for some of the segmentation errors. In addition, the pronunciation errors combined with forced alignment will most likely result in significant segmentation error. Even a HMM segmentation of native speech is not perfect. This was shown in section 5.1 where 3% of the boundaries were more than 80 ms in absolute distance from a manual segmentation. Without a manual segmentation of the database it is impossible to determine for certain the impact of the segmentation errors.

Table 5.4 shows the classification error rate per speaker. The table shows a small subset of the speakers caused most of the classification errors. $15 + 22 = 37$ of 75 classification errors originate from 2 of the speakers.

Speaker (Mother tongue)	Number of errors	Error rate	Residency	Level
4 (Chinese)	10	9.3%	9	1
5 (Chinese)	15	13.9%	5	1
9 (Chinese)	6	5.6%	6	0
12 (Chinese)	2	1.8%	6	1
13 (Chinese)	9	6.3%	42	2
14 (Chinese)	22	20.4%	7	0
6 (Persian)	1	0.9%	5	1
7 (Persian)	3	2.8%	27	2
8 (Persian)	7	6.5%	18	2
All	75	7.7%	-	-

Table 5.4: This table shows the classification error rate per speaker. There were a total of 108 utterances per speaker. The error rate, in this table, was calculated based on the total number of utterances per speaker. Section A.3 shows more information about the speakers. Residency = length in months of residence in Norway. Level = level of Norwegian course.

That two of the speakers are the reason for approximately half of the classification errors indicates that there are some people the automatic methods do not work reliably on. The speech might be too acoustically different from the speech used to train the HMMs, which causes segmentation errors. Another reason for the high error rate may be that the speakers use different clues in the pronunciation to pronounce the vowel length than the other speakers. If the classification features do not capture the necessary

information or are inconsistent, then the classifier will fail.

There does not seem to be a clear connection between the length of residency and the number of classification errors. The level of Norwegian course seems to be a factor which reduces the number of classification errors. The speakers with level 2 course got in general a lower amount of classification errors than the rest of the speakers, but there are too few test utterances to conclude that it is significant. However, it would seem reasonable that speakers with a higher level of Norwegian course are better at speaking Norwegian.

Table 5.4 clearly shows a correlation between the speaker's mother tongue and the number of classification errors. The speakers with Persian as mother tongue are classified with a lower error rate than the speakers with Chinese as mother tongue.

Persian and Norwegian are Indo-European languages, while Chinese is not. The fact that Norwegian and Persian is closer related to each other than Chinese and Norwegian might help the speakers because the language is more familiar. '... one must expect that a non-native will significantly mispronounce sounds that are not in his native auditory collection, by projecting the pronunciation onto his native articulatory and acoustic space which might result in the elimination of certain vital clues' [13]. Thus the reason for a lower error rate for Persian speakers might be that they pronounce sounds closer to native speech.

The fact that the Persians had a high average Norwegian course level and length of residency can be also a factor in the lower error rate. The low number of speakers implies that the result is not statistically significant, but that the speaker's mother tongue is likely to be important. This is as expected, hence the decision to exclude people whose mother tongue is too close to Norwegian in section 3.2. Thus the assumption was reasonable.

In table 5.5 the error rate of the non-sense and regular words is shown independently. The error rate with the non-sense words is 11.5 percentage points lower than for the regular words.

Word type	Number of errors	Error rate	Number of test data
Non-sense words	12	2.5%	486
Regular words	63	13%	486
All words	75	7.7%	972

Table 5.5: Number of classification errors on non-sense words versus regular words. The error rate was calculated based on the number of test utterances for that word type.

As explained in section 3.1 the non-sense words were assumed to be easier to pronounce and segment compared to many other words. The non-sense words were specifically chosen for that very reason. The results clearly indicate that the assumption is true, because most of the classification errors are caused by the regular words.

However, there is another factor which could have made the non-sense words easier to classify. 50% of the database were utterances with non-sense words, which are structured the same way (k - vowel - t - e), while the rest of the utterances were regular words. Thus all the non-sense words have a structure that bears a greater overall resemblance to more of the words that the classifier is trained on, than the structure of a given regular word will. The fact that the classifier is trained on many utterances with the same structure could be a part of the reason the vowel length in non-sense words are classified with a lower error rate than the regular words.

Table 5.6 presents the 7 words pairs with highest number of classification errors. The table shows that the words 'rodde/rode' and 'mulle/mule' resulted in 26 of 75 classification errors. There were 18 different word pairs, thus this is a significant amount.

Word pair	Number of errors	Error rate	Pronunciation errors
Rodde/rode	15	27.8%	13
Mulle/mule	11	20.4%	2
Monne/måne	8	14.8%	2
Lesse/lese	8	14.8%	0
Lynne/lyne	7	13.0%	0
Verre/været	6	11.1%	3
Lisse/lise	5	9.3%	0

Table 5.6: This table shows the 7 word pairs with the highest number of classification errors. Total number of utterances per minimal pair was 54. The error rate was calculated based on the total number of utterances for the minimal pairs. The (perceived) pronunciation errors were assigned by the judge in section 3.4.

Nearly all of the classification errors in the word pair 'rodde/rode' were utterances with pronunciation errors. Inspection of the utterances showed that the non-native speakers pronounced the phoneme 'r' as 'l'. As explained earlier, forced alignment combined with pronunciation error will most likely lead to significant segmentation errors. Thus the classification errors from the utterances with the word pair 'rodde/rode' most likely originate from the segmentation.

For the word pair 'mulle/mule' there was another problem than pronunciation errors. The phoneme /l/ is a semivowel [21]. The pronunciation of /l/ is acoustically close to a vowel. A small difference between the pronounced consonant and vowel combined with a non-native accent could result in unreliable segmentation, thus explaining the high number of classification errors.

5.4 Evaluation of classification features

This section evaluates the different features and finds the best selected feature combination.

Table 5.7 shows how many errors were made with classification features calculated only with long or short HMM segmentation[1]. The classification error rate using only short HMM segmentation is 3.6 percentage points higher than with only long HMM segmentation.

Segmentation type	Number of errors	Error rate
Long HMM segmentation	76	7.8%
Short HMM segmentation	111	11.4%
Both segmentations	75	7.7%

Table 5.7: This table shows how many classification errors are made with classification features calculated only based on segmentation with long or short HMMs.

The result from table 5.7 indicates a large difference in the segmentation with Short HMMs and Long HMMs. Figure 5.3 shows that there are many long vowel length pronunciations having a segmented Vowel duration of 15-25 milliseconds. It is unlikely that a pronounced long vowel got a duration of 15 milliseconds. Thus the most likely explanation for so short Vowel duration is significant segmentation errors. There are many more Vowels with low duration for segmentation with short HMMs than with long HMMs. Based on this, the short HMM segmentation seems to be more unreliable than the long HMM segmentation.

In general using both long and short HMM segmentation should improve the classification result, because they represent the long and short vowels better than only one HMM can. The reason the short HMMs resulted in more unreliable segmentation than long HMMs was not found. One explanation

[1]Long HMM segmentation is a segmentation where a long vowel HMM represents the Vowel. A short HMM segmentation is a segmentation where a short vowel HMM represents the Vowel.

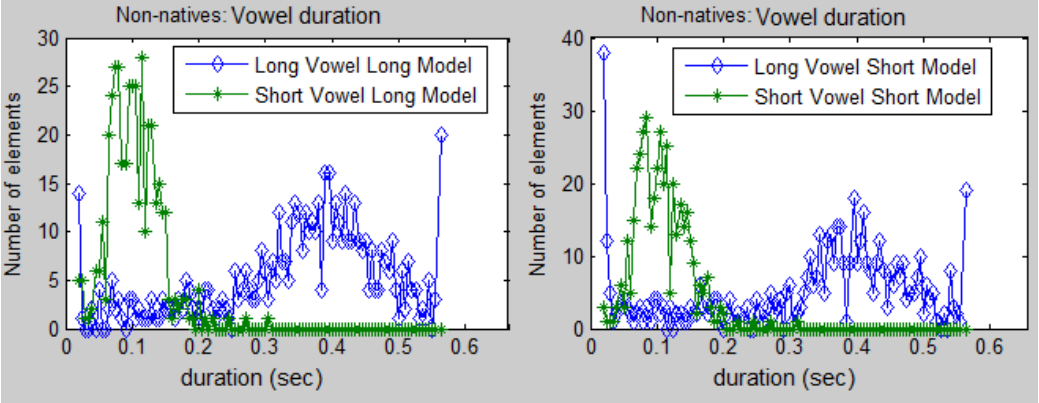


Figure 5.3: HMM segmentation with short and long HMMs representing the Vowel. (Short Vowel Long Model = Short vowel length, long HMM)

for difference is that there might have been more training data for the long vowel HMMs or that the training of long vowel HMMs had better initial conditions. Using an extra segmentation option did not increase the number of classification errors. Thus there is no reason to remove the classification features based on the short HMM segmentation, except if computationally performance is a factor.

It was not expected that many of the long vowel length pronunciations got segmented to a low Vowel duration. The reason for the short Vowel durations is, as will be shown in section 5.5, most likely because of pronunciation errors which causes the forced alignment segmentation to do segmentation errors. That more long vowels were segmented with a low Vowel duration is explained by the number of pronunciation errors for long and short vowel length pronunciations, which is shown in section 3.4. There were about twice the amount of utterances with long vowel length and pronunciation errors, than with short vowel length and pronunciation errors.

Table 5.8 presents the classification error rate produced by either training the classifier on a single feature or using all features except for one feature. The table shows that the Vowel duration is the feature which affects the number of classification errors most, but that even without the Vowel duration the number of errors does not change drastically.

Most of the feature error rates follow what was expected from the theory. The Vowel LL can in average be used to discriminate between vowel length pronunciations, while the Consonant LL is useless. The LLR does not improve the classification result compared to the LL. This is not that surprising because the LLR indicates mostly how probable a phoneme is compared to other phonemes and not which vowel length pronunciation is

Feature X	Use only feature X		Remove feature X	
	Number of errors	Error rate	Number of errors	Error rate
Vowel duration	85	8.7%	88	9.0%
Consonant duration	333	34.3%	71	7.3%
Normalized Duration	151	15.5%	75	7.7%
Vowel LL	184	18.9%	75	7.7%
Consonant LL	432	44.4%	76	7.8%
Total LL	199	20.5%	74	7.6%
Vowel LLR	272	28.2%	75	7.7%
Consonant LLR	462	47.5%	76	7.8%
Total LLR	305	31.4%	76	7.8%
Vowel Uniform Energy	196	20.2%	74	7.6%
Consonant Uniform Energy	401	41.3%	69	7.1%
Vowel State Energy	421	43.3%	77	7.9%
Consonant State Energy	298	30.7%	75	7.7%
All features	75	7.7%	-	-

Table 5.8: Classification result with using only classification feature X and classification result by removing feature X from the feature vector. For the case where feature X is removed, the feature is more important if the error rate increases. Total 972 test utterances.

most likely.

The results from all of the energy features show that the features have some potential for classification of vowel length pronunciations. Used alone the energy can achieve an error rate of 20%. The energy results also show that the way of partitioning the energy might be important. This can be seen by noting the large classification result difference between state and uniform energy.

The Consonant duration is worse than expected compared with theory, though it does in average contrast between different vowel length pronunciations. It is particularly strange that the Normalized Duration contrast worse than using just the Vowel duration because a normalized feature should compensate for different speaking rates. Since the Normalized Duration contrast worse than using just Vowel duration the Consonant duration is too unreliable to be used to normalize the speaking rate. The reason the Consonant duration is unreliable might originate from the automatic segmentation, but the result from 5.1 implies that even manually segmented Consonant duration is not reliable.

As discussed earlier and shown in table 5.8, the classification result can be improved by removing features. For example, the error rate without 'Consonant Energy: Uniform' is 7.1% compared to an error rate of 7.7% with all of the features. By removing the feature that decreases the error rate the most, until no feature further decrease the error rate if removed, the subset of the features which results in least number of classification errors can be found.

The resulting selected classification feature vector consisted of: Vowel LLR, Consonant LLR, Vowel LL, Total LL, Vowel duration, Vowel State Energy, State Consonant Energy. The error rate using the selected feature vector was 6.7% and the number of classification errors was 65. This subset of classification features was used in the rest of the thesis.

5.5 Classification results with confidence

This section presents the confidence measures used to discard test utterances which were detected to be problematic. The thresholds used to discard utterances must be manually chosen. The thresholds were, in this work, chosen based on keeping as many correctly classified vowel length pronunciations while discarding as many classification errors as possible. Another choice could have been to find thresholds which discard an equal number of correctly and wrongly discarded utterances.

The most important results from this section is summed up and evaluated in the end of this section. The end of this section also presents the case where all of the confidence measures are used at the same time. The classification was done using the selected features from section 5.4 with a Norwegian trained classifier.

The number of classification errors without using any confidence measure was:

- 65 errors out of 972 test utterances
- 6.7% error rate

A confidence measure presented in section 2.6 was detection of pronunciation errors based on either the LLR or the WCS. Figure 5.4 indicates a correlation between the Vowel LLR and classification errors. The figure shows that many of the test utterances with low LLR are classified wrong.

That many of the words with low vowel LLR get classified wrong follows the theory. As discussed in section 2.6, low LLR suggests that something has

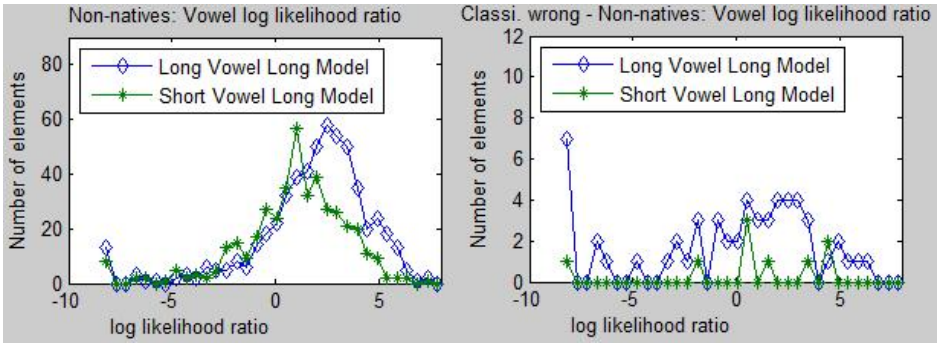


Figure 5.4: Figure to the left: The Vowel LLR for all non-native speakers. Figure to the right: The Vowel LLR for utterances that were classified wrong.

gone wrong in the segmentation, which can be caused by pronunciation of wrong vowel or some other mispronunciation that has resulted in a significant segmentation error. This implies that setting a threshold which discard utterances with Vowel LLR below the threshold can be effective at discarding utterances which are classified wrong. Table 5.9 shows how many of the utterances that were correctly or wrongly discarded for a given threshold.

Threshold (LLR)	Wrongly discarded	Class. error	Unsure vowel length	Pronun. error	Correctly discarded
-8.5	2	10	1	15	17
-7.5	3	10	1	16	18
-6.5	4	12	1	20	23
-5.5	5	12	1	21	24
-4.5	12	13	2	27	30
-3.5	19	13	2	31	34
-2.5	38	16	4	35	40

Table 5.9: This table shows how many of the utterances were correctly or wrongly discarded for a given threshold. As explained in section 4.4, correctly discarded implies that the vowel length was either classified wrong, or that the test utterance was labeled as unsure vowel length or pronunciation error. Wrongly discarded implies that the test utterance was classified correctly and not labeled unsure vowel length or pronunciation error. Total 972 test utterances. Class. = classification, Pronun. = pronunciation

Depending on what should be detected different thresholds can be chosen. An equal number of correctly and wrongly discarded utterances is at -2.5 Vowel LLR, where most of the discarded utterances were labeled as

pronunciation errors. The chosen threshold for best tradeoff between number of discarded classification errors and wrongly discarded utterances is a threshold at -6.5 Vowel LLR. Setting the threshold at -6.5 LLR results in 4 wrongly discarded utterances and 12 discarded classification errors.

The same correlation between classification and pronunciation errors were also found for Consonant LLR and Word Confidence Score, but it was not as strong as for Vowel LLR. They were therefore not used in this work. To see how much the pronunciation errors affected the result all of the pronunciation errors perceived by the judge were removed, which resulted in:

- 19 discarded classification errors
- A total of 149 discarded utterances
- By discarding 15.3% of the test utterances the error rate was reduced with 29.2%.

The results indicate a significant correlation between pronunciation and classification errors. However discarding all pronunciation errors would not improve the classification result much compared to just using the Vowel LLR. Thus by using the Vowel LLR most of the pronunciation errors that affect the classification result were found. This is logical because a high Vowel LLR implies that the Vowel duration is correctly segmented. The Vowel duration is, as shown in section 5.4, the most important classification feature. Thus if it is segmented correctly then the test utterance should be classified correctly.

Based on inspection of the classification features, a feature was found that was added as a confidence measure. Figure 5.3 in section 5.4 shows that many of the utterances with long vowel length were segmented with a Vowel duration of 15-25 ms, while there were few short vowel length utterances in that region. The Vowel duration was found to be the most important factor in the classification feature vector, thus test utterances with such a low Vowel duration should result in classification errors. A table where several Vowel duration discarding thresholds are tested for long and short HMM segmentation is in respectively table C.1 and table C.2. What was thought to be the most important result is discussed in this section.

The results indicate that the low Vowel durations were strongly correlated with pronunciation errors. This is logical because pronunciation errors most likely leads to segmentation errors. An approximately equal number of correctly discarded and wrongly discarded pronunciations was found with a Vowel duration threshold of 55 ms, which resulted in 25 correctly discarded and 28 wrongly discarded utterances. The chosen Vowel duration threshold was 15 ms with long HMMs because that duration was deemed to have the

best tradeoff between number of discarded classification errors and wrongly discarded test utterances. By discarding all utterances with a threshold at 15 ms, the following results were found:

- 3 wrongly discarded utterances
- 11 classification errors
- 1 unsure vowel length pronunciations
- 12 pronunciation errors
- 16 correctly discarded utterances

Table C.1 and table C.2 shows that discarding test utterances based on the short HMM segmentation results in more wrongly discarded than correctly discarded utterances compared to the long segmentation. This most likely comes from the fact that the long HMM segmentation was more reliable than the short HMM segmentation, which is discussed in section 5.4.

Another confidence measure was using an unsure window, which is described in section 2.6. The LDA score distribution for long, unsure long, short and unsure short vowel length pronunciations is shown in figure 5.5.

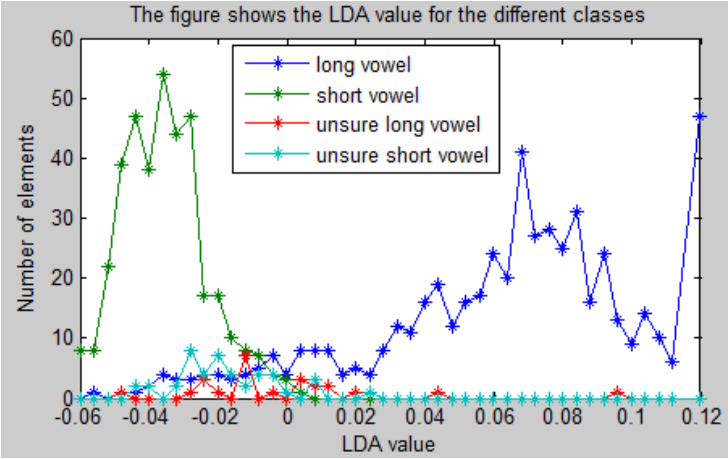


Figure 5.5: This figure shows the distribution of the LDA score calculated when classifying the non-native speakers

As can be seen from figure 5.5, many of the long vowel length pronunciations have a LDA score below zero. A vowel length pronunciation below zero is classified as short vowel length, thus they are classified wrong. The figure also shows that there are many unsure vowel length pronunciations

just below zero LDA score. The classification errors and unsure vowels can be discarded by using an unsure window as described in section 2.6. Table 5.10 shows the number of correctly and wrongly discarded utterances for a given window.

Unsure window (LDA score)	Wrongly discarded	Class. error	Unsure vowel length	Pronun. error	Correctly discarded
[0, 0.004]	2	3	1	2	6
[0, 0.008]	6	4	4	6	14
[0, 0.012]	11	4	9	11	22
[0, 0.016]	19	4	11	12	24
[0, 0.02]	22	4	11	13	25
[0, 0.024]	25	4	12	16	28
[0, 0.028]	29	4	14	16	30
[0, 0.032]	36	4	14	17	31
[0, -0.004]	3	7	5	3	13
[0, -0.008]	9	12	9	6	23
[0, -0.012]	16	16	18	13	37
[0, -0.016]	26	19	22	15	44
[0, -0.02]	42	23	30	20	57
[0, -0.024]	54	27	37	27	73
[0, -0.028]	95	30	46	34	91

Table 5.10: This table shows how many of the utterances were correctly or wrongly discarded for a given unsure window. As explained in section 4.4, correctly discarded implies that the vowel length was either classified wrong, or that the test utterance was labeled as unsure vowel length or pronunciation error. Wrongly discarded implies that the test utterance was classified correctly and not labeled unsure vowel length or pronunciation error. Total 972 test utterances. Class. = classification, Pronun. = pronunciation

The results from table 5.10 indicates a strong correlation between low LDA score and pronunciation errors, classification errors and unsure vowel length pronunciations. An approximately equal number of correctly and wrongly discarded utterances can be found with an unsure window starting at -0.028 and ending at 0.028 LDA score. The equal number of discarded window results in approximately 129 discarded utterances for each class. The chosen window was from 0 to -0.012 LDA because that window had a reasonable tradeoff between number of classification errors and wrongly discarded utterances. With the chosen window 37 utterances are correctly discarded while 16 utterances are wrongly discarded.

Table 5.11 shows all of the results from this section together with the Confidence Evaluation Score. The Confidence Evaluation Score is, as explained in section 4.4, the percent chance of correctly discarding utterances.

Specification	Error rate	Number of classi. errors	Percent test utt. discarded	Number of test utt. discarded	CES
Base result	6.7%	65	-	-	-
LDA WS [-0.012, 0]	4.5%	39	5.8%	53	70%
Disc. Vowel LLR below -6.5	5.7%	11	3.0%	27	89%
Disc. vowels under 15 ms *	5.8%	10	2.0%	19	84%

Table 5.11: This table shows the error rate by using confidence measures with the chosen thresholds. The error rate was calculated based on the number of test utterances that was not discarded. * = Discarded all words with under 15 ms vowel length(segmented with long HMM). Perc = percieved, pron. = pronunciation, disc. = discarded, WS = window size, utt. = utterance, CES = Confidence Evaluation Score, class. = classification.

Table 5.11 shows that the 3 chosen confidence measures have a CES higher than 50%. This implies that they discard more classification errors, pronunciation errors and unsure vowels than correctly classified and pronounced words. The high CES indicates that the confidence measures discard the problematic utterances.

Some of the detected utterances with the confidence measures will overlap. The 3 confidence measures were combined to see the effect of using the confidence measures together. The result was:

- 32 classification errors in total, 33 less classification errors compared to the base result without these confidence measures
- 81 utterances discarded out of 972 test words, 8.3% of the test utterances were discarded
- 3.6% error rate (based on the remaining 891 test utterances)
- 33 of the discarded utterances were perceived by the judge as pronunciation errors
- Discarded 19 out of 69 utterances with unsure vowel length pronunciations
- 44 of the discarded test utterances were either unsure vowels or pronunciation errors

- 4.5% of the discarded test utterances were either unsure vowels or pronunciation errors
- 74% CES

As the results show, the error rate was approximately halved compared to the error rate without confidence measures. To do this, 8.3% of the test utterances were discarded. The CES of 74% implies that most of the utterances were correctly discarded. Thus the confidence measures were successful at pruning out utterances with high potential of being classified wrong.

As can be seen by the results in this section many more pronunciation errors, unsure vowel length pronunciations and classification errors could have been discarded with other thresholds. Which thresholds to chose depends on what is important: keeping many correctly pronounced utterances and correctly classified vowel length pronunciations or discarding wrongly pronounced utterances and classification errors.

Chapter 6

Conclusion

This thesis has proposed and tested several methods for evaluation of vowel length pronunciations for non-native speech. A two-stage system is developed. In the first stage classification features are derived based on forced alignment segmentation using phone HMMs trained on native continuous speech. For comparison corresponding classification features are derived based on templates and DTW. The second stage consists of a simple binary linear classifier deciding on short or long vowels. For a case of 6 Chinese and 3 Persian people, error rates of 7.7% and 8.7% are achieved using respectively HMM and DTW.

The difference is so small that for a CAPT system where there are no trained HMMs the DTW is a good option. A DTW based system requires recordings (reference templates of native language) for every word which are to be used. In contrast, phone based HMM models can model any word using a dictionary. Another advantage with HMMs is that they model natural pronunciation variations while DTW requires the user to mimic the teacher voice for a good score. Thus HMM based system should be used when possible.

Classification of vowel length pronunciations for native speakers resulted in an error rate of 0.9%. This low error rate can be seen as a lower bound for the non-native speech error rate; i.e. a goal to achieve for the non-native speaker. Manual inspection of the non-native classification errors showed that approximately half of the utterances were subject to significant segmentation errors. Non-native accent and pronunciation errors are the most likely reasons for these errors. The non-native accent results in a mismatch between the trained HMMs and the speech. Pronunciation errors combined with forced alignment will inevitably lead to segmentation errors and thus classification errors. The classification features are calculated based on the segmentation. Thus using methods which can compensate for the

accent and detect pronunciations error should improve the vowel length classification error rate.

The classification result was inspected with regards to the word type. The inspection showed that few of the words resulted in most of the classification errors. This indicates that the classifier is unreliable on certain phonemes or phoneme combinations. This implies that the classifier can be improved significantly by focusing on improving the classifier on these phonemes.

Further, the classification features which were consistent between native and non-native speakers was found by removing classification features which resulted in an increase in the classification error rate. The selected classification features resulted in an error rate of 6.7%.

Confidence measures were used to detect problematic utterances which resulted in most of the classification errors. The confidence measures were able to discard about half of the classification errors resulting in a classification error rate of 3.6%. The downside with the confidence measures was that 8.3% of the test utterances had to be discarded to achieve this result. However, 74% of the discarded utterances was either classification errors, pronunciation errors or labeled as unsure by the judge. Thus the confidence measures were effective at discarding problematic utterances. The low classification error rate when using the confidence measures implies that the approach for classifying vowel length pronunciations is a promising method.

The confidence measures and the selected classification features were found based on using the recorded database to select features and confidence measures that improved the classification result. Thus the confidence measures and features were 'trained' on the recorded database. This implies that it is somewhat uncertain whether the error rate would increase with a new database.

Bibliography

[1] Definition of confidence measure. http://videolectures.net/mlmi04ch_cox_cmsr/. Retrieved 04.06.2009.

[2] Definition of phoneme. The American Heritage Dictionary of the English Language, Fourth Edition. http://dictionary.reference.com/browse/phoneme. Retrieved 10.06.2009.

[3] Definition of utterance. http://dictionary.reference.com/browse/utterance. Retrieved 10.06.2009.

[4] Fisher linear discrimination by olga veksler. www.csd.uwo.ca/~olga/Courses//CS434a_541a//Lecture8.pdf. Retrieved 24.03.2009.

[5] Hidden markov model toolkit. http://htk.eng.cam.ac.uk/. Retrieved 27.2.2009.

[6] Hidden markov model toolkit book. http://htk.eng.cam.ac.uk/prot-docs/htk_book.shtml. Retrieved 12.10.2008.

[7] Sox. http://sox.sourceforge.net/. Retrieved 22.02.2009.

[8] Speech courses. http://languagelanguage.com/no2/index.php?cPath=28&language=no. Retrieved 28.10.2008.

[9] Tell me more. http://www.tellmemore.com/content/view/full/128. Retrieved 05.11.2008.

[10] Viterbi illustration. http://upload.wikimedia.org/wikipedia/commons/7/71/Hmm-Viterbi-algorithm-med.png. Retrieved 5.05.2009.

[11] A. Neri, C. Cucchiarini, H. Strik. "ASR-based corrective feedback on pronounciation: does it really work?". Centre for Language and Speech Technology (CLST), Radboud University Nijmegen, The Netherlands.

[12] A. Neri, C. Cucchiarini, H. Strik. "Automatic Speech Recognition for second language learning: How and why it actually works". Department of Language and Speech, University of Nijmegen, The Netherlands.

[13] D. Van Compernolle. "Recognizing speech of goats, wolves, sheep and ... non-natives". Speech Communication, vol. 35, pp 71-79, 2001.

[14] Dawn M. Behne, Peter E. Czigler, Kirk P. H. Sullivan. "Percieved Swedish vowel quantity: British versus native listeners". Phonum, vol. 6, UmeåUniversity, 1998.

[15] Eivind Versvik. "Computer Assisted Pronunciation Training". Department of Electronics and Telecommunications, Norwegian University of Science and Technology, 2008.

[16] Gjert Kristoffersen. "Kvantitet i norsk". Norsk Lingvistisk Tidsskrift, 1992.

[17] Lawrence Rabiner, Biing-Hwang Juang. "Fundamentals of speech recognition". Prentice Hall, Inc., Englewood Cliffs, New Jersey 07632, 1993.

[18] Min Xu et al. "HMM-based audio keyword generation". Kiyoharu Aizawa, Yuichi Nakamura, Shin'ichi Satoh. Advances in Multimedia Information Processing: 5th Pacific Rim Conference on Multimedia. Springer., 2004.

[19] S. Balakrishnama, A. Ganapathiraju. "Linear discriminant analysis - A brief tutorial". Institute for Signal and Information Processing, Department of Electrical and Computer Engineering, Mississippi State University.

[20] S.M. Witt, S.J. Young. "Phone-level pronunciation scoring and assessment for interactive language learning". Engineering Department, Cambridge University, Trumpington Street, Cambridge, UK, 2000.

[21] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon. "Spoken Language Processing". Prentice Hall, Inc., Upper Saddle River, New Jersey 07458, 2001.

Appendices

Appendix A

Information about the databases

A.1 General information

General information	
Room:	Sound proof
Headset with microphone	Sennheisser PC-156
Computer OS	Windows XP
Soundcard	SoundMax (using standard settings)
Recording software	Developed java software
Recorded at sampling rate	48 kHz
Downsampled using SoX [7] (high quality filter)	16 kHz
Estimated time used pr speaker	30-40 min

Table A.1: General information about the recording conditions and equipment

A.2 List of recorded words and number of recorded words

The number of unique recorded words were 36. With 3 repetitions of each word per speaker results in 108 recorded words per speaker.

Country	Number of speakers	Vowel words
Norwegian	4	432
Persian	3	324
Chinese	6	648
All non natives	9	972
All utterances	13	1404

Table A.2: What country the speakers were from

Word list	
Vowel length words	Vowel length nonsense words
hane	kate
Hanne	katte
lese	kete
lesse	kette
Lise	kite
lisse	kitte
rode	kote
rodde	kotte
mule	kute
mulle	kutte
lyne	kyte
lynne	kytte
været	kæte
verre	kætte
dømet	køte
dømme	køtte
måne	kåte
monne	kåtte

Table A.3: List of the selected words

A.3 Information about the speakers

Language names using: ISO 693-1 CODE. Japan = JA, Norwegian = NO, English = EN, Chinese (mandarin) = ZH, Persian = FA, Arabic = AR, French = FR.

Explanation of the abbreviations used in the table.

- ID = Speaker ID
- MT = Mother tongue
- Lang. = Knoweldge of other languages (than norwegian)
- Residency = Length of residence in Norway (in months)
- Level = Level of Norwegian course (0 = no course)
- Use = Use of Norwegian (1 = rarely, 4 = often)
- Sex = Male (M) or female (F)
- Age = Age of the speaker
- Edu. = Level of education

ID	MT	Lang.	Residency	Level	Use	Sex	Age	Edu.
2	NO	-	-	-	-	M	24	-
3	NO	-	-	-	-	M	31	-
10	NO	-	-	-	-	M	23	-
15	NO	-	-	-	-	M	30	-
04	ZH	EN	9	1	1	M	30	Master student
05	ZH	EN	5	1	1	M	28	PhD student
09	ZH	EN	6	0	1	M	24	Master student
12	ZH	EN, JA	6	1	1	F	29	PhD student
13	ZH	EN	42	2	1	M	26	PhD student
14	ZH	EN	7	0	1	F	27	PhD student
06	FA	EN	5	1	1	M	30	PhD student
07	FA	EN, AR	27	2	1.5	M	27.5	PhD student
08	FA	EN, FR	18	2	1	M	30	PhD student

Table A.4: Information about the speakers

A.4 Information given to the speaker before recording

The speakers in the test database were given some information about the recording procedure and why they were being recorded. The information given to them is presented here.

(The tone words were not used in this project)

Recording of speech material for master project on computer assisted pronunciation training. Spring 2009

Background

The project objective is to investigate methods for automatically detecting pronunciation errors in speech from non-natives learning Norwegian. In order to learn any foreign language a lot of practice is needed. A computer-assisted pronunciation training (CAPT) systems is always available and can be used at home or from any other convenient place and may therefore help non-natives learn Norwegian faster and better. CAPT systems have been shown to be particularly effective for beginning students or students with a low proficiency level. Thus, CAPT and traditional classroom teaching complement each other.

Most computer based language learning systems will present a recorded teacher voice the student can imitate. It may be difficult for the student to hear the errors he/she makes himself/herself and automatic pronunciation error detection would greatly improve the usability of a language learning system. There exits some systems including CAPT, but not for learning Norwegian.

In order to do research on automatically detecting pronunciation errors in speech from non-natives learning Norwegian we need recordings of non-natives.
We hope you can help us!

The recordings will take place in room C031 in the Electro building.
It should take about 15-20 minutes.

Please ask if you have any questions!
Best regards,
Ingunn Amdal and Eivind Versvik
Department of Electronics and Telecommunications, NTNU
Email: ingunn.amdal@iet.ntnu.no
Mobile: 957 66 821

Recording procedure

You will be presented with isolated words both in writing and by a Norwegian teacher voice. We ask you to try to repeat the word with the same pronunciation as the teacher, but using your own natural voice quality.

A recording supervisor (Eivind) will be present to assure the recording is OK and you can repeat a word if there is a problem. We don't want too many repetitions as there will be a learning effect. Avoid hesitations and creaky voice.
You will decide the tempo in playing the teacher voice and recording your own voice.
Don't hesitate to ask Eivind if anything is unclear!

We are focusing on two aspects of Norwegian that may be especially difficult:

1. Vowel length
2. Word tone

Vowel length

In Norwegian the vowels may be long or short. The words "hat", with a long vowel, and "hatt", with a short vowel, means different things (hate and hat/had).
We have chosen 9 word pairs where the only difference is the vowel length.

Read the list once first for practice and to get used to the recording equipment. This recording will not be used. Then the list will be repeated three times to get three versions of your pronunciation of the word.

The consonants before and after the vowel may affect the pronunciation. We therefore also want recordings of long and short vowels using the same consonants: k-vowel-t-e.
All these nonsense words are perfectly possible to pronounce in Norwegian (some of them are actual words). There are again 9 "word" pairs and the list will be repeated three times.

Word tone

In Norwegian we have two different word tones. The pattern may vary from dialect to dialect and we have chosen the South-East variant (which is fairly close to Trøndersk as well).

Word lists for vowel length:

Real Norwegian words with translation.
The first word in a pair is with a long vowel, the second with a short vowel.

1.	hane	(a) rooster/cock
	Hanne	girl's name
2.	lese	(to) read
	lesse	(to) load
3.	Lise	girl's name
	lisse	(a) lace/string
4.	rode	(an) area (not area in general)
	rodde	rowed (verb)
5.	mule	(a) muzzle/mule
	mulle	name of a fish
6.	lyne	(to) lighten/flash
	lynne	(a) temperament/disposition
7.	været	the weather
	verre	worse (adjective)
8.	dømet	the example
	dømme	(to) convict/judge
9.	måne	(a) moon
	monne	(to) help/avail (not help in general)

The nonsense words are also presented in pairs:

1. kate
   katte
2. kete
   kette
   ...

Word list for word tone:

Real Norwegian words with translation.
The first word in a pair is with word tone 1, the second with word tone 2.

1.	bønder	farmers
	bønner	beans
2.	året	the year
	åre	(an) oar
3.	bygget	the building
	bygge	(to) build
4.	fjæra	the feather
	fjæra	the beach
5.	loven	the law
	låven	the barn
6.	smilet	the smile
	smile	(to) smile
7.	svaret	the answer
	svare	(to) answer
8.	tanken	the tank (container)
	tanken	the idea/thought
9.	Verdi	composer's name
	verdig	dignified

As you may notice it is quite common in Norwegian that corresponding verb and noun differ only by word tone.

Appendix B

HMM and DTW information

B.1 DTW normalization

Normalization of the 13 MFCCs when doing DTW segmentation was done as in an earlier project [15]. The reason for normalization and how the normalization was found is explained there, but the main purpose is to emphasize important MFCCs, which is the lower frequency and energy MFCCs.

The normalization was done as follows. The mean is removed for all of the MFCCs. The standard deviation is set to 1, then the standard deviaion for lower frequency components is emphasized. The list below shows the standard deviation for each component in the MFCC vector:

- MFCC 1: Standard deviation = 4
- MFCC 2: Standard deviation = 3
- MFCC 3-5: Standard deviation = 2
- MFCC 6-12: Standard Deviation = 1
- MFCC 13 (energy): Standard deviation = 4

The first MFCCs represent the lower frequency components of the sound signal.

B.2 Hidden Markov Model training data

Information about both the continuous and isolated speech database:

- 3 states per phoneme/silence
- 8 mixture components per state
- 39 dimensional MFCC vector (includes energy and delta delta coefficients)
- Trained with a MFCC step size of 5 ms and window size of 15 ms
- Separate states for long and short vowel phonemes
- About 20 hours of speech
- Around 900 different speakers from all of Norway

Information about the continuous speech database:

- The sound files contain sentences
- 10.000 utterances

Information about the isolated speech database:

- The sound files contain word(s) and spelling
- 30.000 utterances

Both of the databases are trained using context indepedent models. All of models are trained using 3 states per phoneme/silence. The silence model allows skipping of states, while the phoneme models only allow transition to the next or same state.

B.3 HTK MFCC parameterization

The HTK parameterization config file is shown below. For more information about HTK config files and what the different terms means, see the HTK book [6].

```
# Acoustic parameterization for segmentation
# Use normalized log energy and CMS for better cross-corpus performance

# Input waveform file format (16 kHz 16 bit wav format)
 SOURCEKIND     = WAVEFORM
 SOURCEFORMAT   = WAV
 SOURCERATE     = 625
 ZMEANSOURCE    = TRUE  # To avoid DC offset

# Parameterization (5 msec frame shift, 15 msec frame window)
 TARGETKIND     = MFCC_D_A_E_Z
 TARGETFORMAT   = HTK
 TARGETRATE     = 50000
 WINDOWSIZE     = 150000.0
 USEHAMMING     = TRUE
 PREEMCOEF      = 0.97
 USEPOWER       = FALSE
 NUMCHANS       = 26
 LPCORDER       = 12
 CEPLIFTER      = 22
 NUMCEPS        = 12
 ENORMALISE     = TRUE

 SAVECOMPRESSED = FALSE
 SAVEWITHCRC    = FALSE
```

Appendix C

Results

Threshold (ms)	Wrongly discarded	Class. error	Unsure vowel length	Pronun. error	Correctly discarded
15	3	11	1	12	16
20	5	13	1	14	19
25	7	13	1	14	19
30	7	13	2	15	21
35	8	14	2	16	22
40	10	14	2	17	23
45	13	15	2	18	25
50	19	15	2	18	25
55	28	15	2	18	25
60	44	16	2	20	28
65	58	18	2	22	31

Table C.1: This table shows how many of the utterances were correctly or wrongly discarded for a given Vowel duration threshold with long HMM segmentation. As explained in section 4.4 correctly discarded implies that the vowel length was either classified wrong, or that the test utterance was labeled as unsure vowel length or pronunciation error. Wrongly discarded means that the test utterance was classified correctly and not unsure vowel length or pronunciation error. Total 972 test utterances. Class. = classification, Pronun = pronunciation

Threshold (ms)	Wrongly discarded	Class. error	Unsure vowel length	Pronun. error	Correctly discarded
15	20	9	2	13	18
20	27	11	2	16	22
25	33	15	2	19	28
30	34	15	3	20	29
35	37	16	4	20	31
40	41	16	4	22	33
45	44	17	4	23	35
50	50	17	4	24	36
55	62	18	4	26	39
60	73	18	4	27	40
65	85	18	4	29	42

Table C.2: This table shows how many of the utterances were correctly or wrongly discarded for a given Vowel duration threshold with short HMM segmentation. As explained in section 4.4 correctly discarded implies that the vowel length was either classified wrong, or that the test utterance was labeled as unsure vowel length or pronunciation error. Wrongly discarded means that the test utterance was classified correctly and not unsure vowel length or pronunciation error. Total 972 test utterances. Class. = classification, Pronun = pronunciation

Appendix D

CAPT-demo

The CAPT-demo's graphical user interface is shown in D.1. In addition to training on vowel length the CAPT-demo includes an option to train on word tone, but that module was not changed in this thesis.

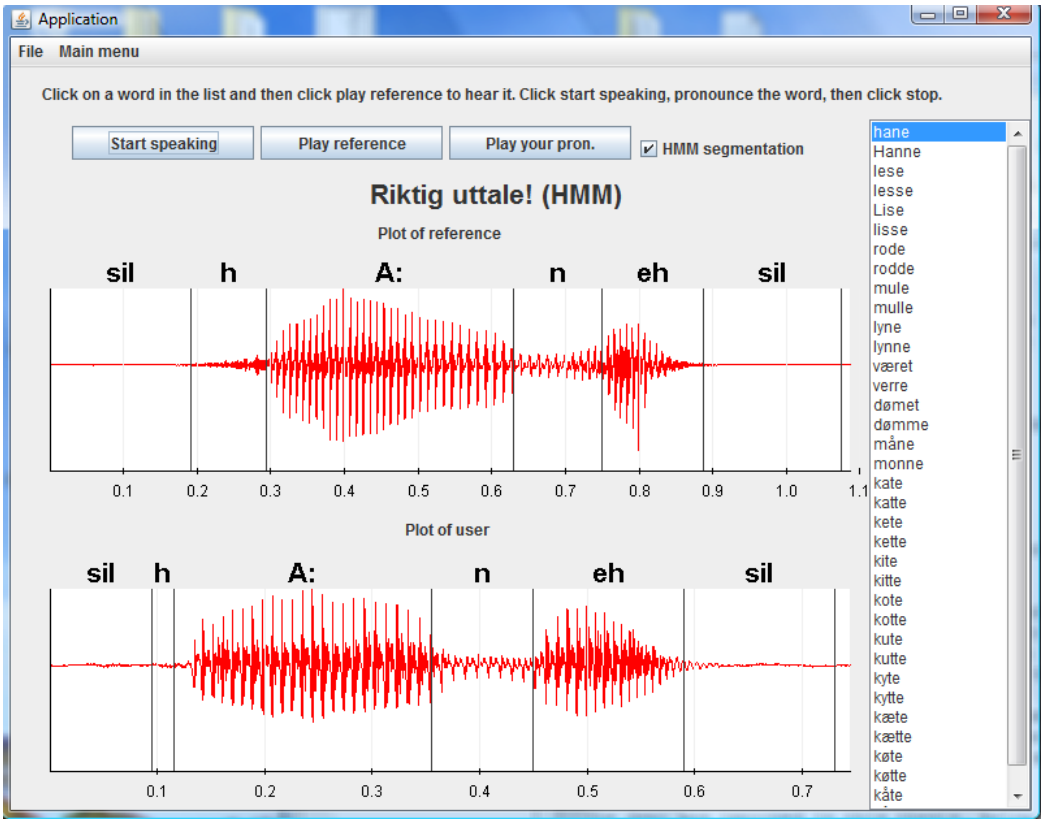


Figure D.1: The graphical user interface

The demo was first developed in figure [15]. Some information about the demo:

- Based on Java. Used the Swing library to make the GUI.
- Can play a teacher voice (reference).
- Gives simple text feedback based on the user's pronunciation.
- User driven system. The user needs to click on 'Start speaking' before pronouncing a word. Then 'Stop' (same button) when the user is finished talking.
- A voice activity detection algorithm removes most of the silence before and after the word.
- The sound signal of the reference and user is plotted. The plotted sound signal is segmented for the reference and user. The reference has a manually segmentation, while the user is automatically segmented.

The demo was improved in this thesis. Below is a list of the improvements:

- New reference sound files.
- Seperate play reference button.
- A button to play the user's last pronunciation
- List of words instead of a list with sound files.
- The program use HMMs to do segmentation.
- Option for DTW segmentation (deselect the 'HMM segmentation' checkbox).
- The program calculates all classification features except for the ones based on LLR and use them to classify.
- The program calculates which vowel length was pronounced. Based on that the program uses the forced alignment segmentation with long vowel HMM if the pronunciation was long vowel length and vice versa for short vowel length.